ORACLE

Oracle® Thesaurus Management System

User's Guide

Release 5.1

E53658-02

March 2015

ORACLE®

Oracle Thesaurus Management System User's Guide, Release 5.1

E53658-02

Contents

Part I   Setting Up and Administering TMS

3   Administration

5 Integrating TMS with Other Systems

6 Defining and Loading Dictionaries

7 Defining Other TMS Elements

8   Upgrading to a New Dictionary Version

9   Allocating Tasks to TMS Users

Part II   Managing Classifications

10   Omission Management

11   VTA and VTI Maintenance

12   Repository Maintenance

Part III   Browsing the Data Repository

13   Browsing the Repository in TMS

14 Using the HTML Browser

A Sample Dictionary Definitions

Glossary

Index

Preface

This manual describes the use and administration of Oracle Thesaurus Management System (TMS).

This preface contains:

- Audience on page xxi
- Documentation Accessibility on page xxi
- Finding Information and Patches on My Oracle Support on page xxi
- Finding Oracle Documentation on page xxiii
- Related Documents on page xxiv
- Conventions on page xxiv

Audience

This document is intended for all TMS users and administrators.

- Administrators must be able to read SQL*Plus scripts and edit and run them, in order to create TMS users and grant them roles. Administrative tasks are covered in Chapter 3, "Administration".
- Information technology expertise is not required for other users.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Finding Information and Patches on My Oracle Support

Your source for the latest information about Oracle Thesaurus Management System is Oracle Support's self-service website My Oracle Support (formerly MetaLink).

Before you install and use Oracle Thesaurus Management System, always visit the My Oracle Support website for the latest information, including alerts, White Papers, installation verification (smoke) tests, bulletins, and patches.

Creating a My Oracle Support Account

You must register at My Oracle Support to obtain a user name and password account before you can enter the website.

To register for My Oracle Support:

1. Open a Web browser to https://support.oracle.com.

2. Click the **Register** link to create a My Oracle Support account. The registration page opens.

3. Follow the instructions on the registration page.

Signing In to My Oracle Support

To sign in to My Oracle Support:

1. Open a Web browser to https://support.oracle.com.

2. Click **Sign In.**

3. Enter your user name and password.

4. Click **Go** to open the My Oracle Support home page.

Finding Information on My Oracle Support

There are many ways to find information on My Oracle Support.

Searching by Article ID

The fastest way to search for information, including alerts, White Papers, installation verification (smoke) tests, and bulletins is by the article ID number, if you know it.

To search by article ID:

1. Sign in to My Oracle Support at https://support.oracle.com.

2. Locate the Search box in the upper right corner of the My Oracle Support page.

3. Click the sources icon to the left of the search box, and then select **Article ID** from the list.

4. Enter the article ID number in the text box.

5. Click the magnifying glass icon to the right of the search box (or press the Enter key) to execute your search.

    The Knowledge page displays the results of your search. If the article is found, click the link to view the abstract, text, attachments, and related products.

Searching by Product and Topic

You can use the following My Oracle Support tools to browse and search the knowledge base:

- Product Focus — On the Knowledge page under Select Product, type part of the product name and the system immediately filters the product list by the letters you have typed. (You do not need to type "Oracle.") Select the product you want from the filtered list and then use other search or browse tools to find the information you need.

- Advanced Search — You can specify one or more search criteria, such as source, exact phrase, and related product, to find information. This option is available from the **Advanced** link on almost all pages.

Finding Patches on My Oracle Support

Be sure to check My Oracle Support for the latest patches, if any, for your product. You can search for patches by patch ID or number, or by product or family.

To locate and download a patch:

1. Sign in to My Oracle Support at https://support.oracle.com.

2. Click the **Patches & Updates** tab. The Patches & Updates page opens and displays the Patch Search region. You have the following options:

   - In the **Patch ID or Number** field, enter the number of the patch you want. (This number is the same as the primary bug number fixed by the patch.) This option is useful if you already know the patch number.

   - To find a patch by product name, release, and platform, click the **Product or Family** link to enter one or more search criteria.

3. Click **Search** to execute your query. The Patch Search Results page opens.

4. Click the patch ID number. The system displays details about the patch. In addition, you can view the Read Me file before downloading the patch.

5. Click **Download.** Follow the instructions on the screen to download, save, and install the patch files.

Finding Oracle Documentation

The Oracle website contains links to all Oracle user and reference documentation. You can view or download a single document or an entire product library.

Finding Oracle Health Sciences Documentation

To get user documentation for Oracle Health Sciences applications, including an HTML version of this guide, go to the Oracle Health Sciences documentation page at:

http://www.oracle.com/technetwork/documentation/hsgbu-154445.html

> **Note:** Always check the Oracle Health Sciences Documentation page to ensure you have the latest updates to the documentation.

Finding Other Oracle Documentation

To get user documentation for other Oracle products:

1. Go to the following Web page:

   http://www.oracle.com/technology/documentation/index.html

   Alternatively, you can go to http://www.oracle.com, point to the Support tab, and then click **Documentation.**

2. Scroll to the product you need and click the link.

3. Click the link for the documentation you need.

Related Documents

This section lists the documents in the Oracle Thesaurus Management System documentation set. The most recent version of each guide is posted on the Oracle website; see "Finding Oracle Health Sciences Documentation" on page xxiii.

- *Oracle Thesaurus Management System Installation Guide*

- *Oracle Thesaurus Management System User's Guide*

In addition, Oracle Thesaurus Management System customers can request a copy of the *Oracle Thesaurus Management System Technical Reference Manual* from Oracle Support.

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

What's New

This section describes the new features and enhancements included in recent releases of Oracle Thesaurus Management System (TMS). All changes are included in the current release.

- New Features in Release 5.1
- New Features in Release 5.0
- New Features in Release 4.6.2
- New Features in Release 4.6.1
- New Features in Release 4.6
- New Features in Release 4.5.2
- New Features in Release 4.5.1

New Features in Release 5.1

Release 5.1 includes the following new features.

Support for WHO-Drug Format C

In **WHO-Drug Format C**, many drug names occur more than once in the coding level. They are distinguished from one another by auxiliary information such as the country where they are distributed, pharmaceutical form, and strength. Verbatim terms need to be classified to the dictionary term whose auxiliary information matches their own. To support this, TMS 5.1 includes the following enhancements:

- You can allow nonunique terms in the coding and VT dictionary levels.

- You can define sublevels of the classification level for each type of auxiliary information.

- If you are using Oracle Clinical, you can set up question sets with additional input questions to collect auxiliary information for each source term.

- A new type of classification called a **verbatim term individual (VTI)** is created when you classify a source term to a nonunique dictionary term. TMS handles VTIs similarly to VTAs, but with some differences. VTIs are always created as approved and are always specific to a domain—never global. Both VTAs and VTIs can be designated either misspelled or accepted. Both VTAs and VTIs are supported in the same dictionary.

- If a dictionary is defined to allow autoclassification with auxiliary information, autoclassification classifies all occurrences of the same verbatim term with the same auxiliary information to the same dictionary term.

Coding Identical Source Terms to Different Dictionary Terms

MedDRA supports classifying identical source terms to different dictionary terms without auxiliary information. For example, the source term "Migraine Headache" might sometimes be classified as "Headache" and other times as "Migraine."

TMS supports this by letting you allow nonunique terms in the VT level.

Autoclassification cannot handle coding source terms to different dictionary terms without auxiliary information.

Classifying High-Level Terms Based on Indication

Different instances of the same dictionary coding-level term may be classified to different high level terms—for example, Anatomical - Therapeutic - Chemical (ATC) terms in WHO-Drug Format C—based on the indication for which the drug was taken by a patient. In this case, multiple high-level terms are related to a single lower-level dictionary term with no primary path defined. TMS cannot derive the high-level term and creates an *indication omission*. After a user manually classifies the indication omission, TMS derives all current and future occurrences of the same term for the same verbatim term/indication combination.

To collect indication information for source terms:

- If you are using Oracle Clinical you can create a question set variable of type Indication to collect the indication.

- If you are using a different external source data system you must import "Indication" and the indication value as a value pair.

> **Note:** Indication omissions are created only after a verbatim term is classified. Users cannot classify an indication omission without an indication, and after a verbatim term is classified, there is no way to send an action message to the source system requesting information. So you must set up indication collection before classifying verbatim terms.
>
> Indication omissions are not replicated. If TMS runs in a distributed environment, indication assignments may vary across instances. Users can run the Indication Assignments Report on each instance and compare the results.

New and Modified Reports

Existing reports that include VTAs now include VTIs too, when applicable:

- Dictionary Impacts
- Predict VTA/VTI
- VTA/VTI Creation
- Classification Statistics
- VT Domain Differences
- Classification to a New Domain
- VT Modifications

The following are new reports:

- VT Auxiliary Classifications

- Unclassifiable Indication Omissions

- Indication Assignments

Other Changes to Support 5.1 Enhancements

- A new dictionary term filter supports searching for dictionary terms using specific auxiliary information values.

- Repository Maintenance APIs include VTI functionality.

- The Maintain Repository Data window includes VTIs.

- The Repository Authoring window includes VTIs.

- The Copy Domain operation includes auxiliary information and uses mapping information provided by the user.

- The Autoprocess VTAs job includes VTIs.

- The Maintain Predictionary window includes VTIs.

New Features in Release 5.0

Release 5.0 includes upgrades to the technology stack, leveraging more up-to-date capabilities and tools and alleviating end-of-support windows for legacy components. The technology stack is the same as for Oracle Clinical 5.0, facilitating integration between the two systems.

In addition, because of the technology stack upgrade, the HTML Browser user interface has been rewritten using the Oracle Application Developer Framework. The functionality remains the same.

Release 5.0 supports integration with Oracle Health Sciences Data Management Workbench (Oracle DMW), a data transformation and discrepancy management system that can be integrated with Oracle Health Sciences InForm. Documentation is included in the Oracle DMW *Installation Guide* and *User's Guide*.

New Features in Release 4.6.2

Release 4.6.2 includes upgrades to the technology stack, leveraging more up-to-date capabilities and tools and alleviating end-of-support windows for legacy components. The technology stack is the same as for Oracle Clinical 4.6.2, facilitating integration between the two systems.

New Features in Release 4.6.1

Release 4.6.1 includes the following new features:

- Classification to Nonapproved Dictionary Terms Can Be Allowed on page xxvii

- Filtering and Multirecord Selection for Task Allocation by Term on page xxviii

- New Informative Note Attribute for Use in RDC Special Listings on page xxviii

Classification to Nonapproved Dictionary Terms Can Be Allowed

You can allow classifying terms to nonapproved dictionary terms in selected domains. For example, in a domain used for an active study you can prevent classification to nonapproved terms, but in a domain used for post-marketing surveillance, you can allow classification to nonapproved terms to prevent unnecessary reclassification after a dictionary upgrade. Global VTAs cannot be linked to nonapproved dictionary terms.

This change affects many features and is documented in each relevant section:

- [Creating Domains and Assigning Dictionaries to Domains](#) on page 6-37
- [Setting TMS Preferences with Reference Codelist Values](#) on page 3-25
- [Defining Search Objects](#) on page 7-10

> **Note:** If you are already using custom search objects and now start allowing classification to nonapproved dictionary terms in some domain/dictionary combinations, you may need to modify your algorithm. The sample Domain Match search object now contains code you can use as a model; see ["Sample Domain Match Search Object"](#) on page 7-11.

- [Running the Autoprocess VTAs/VTIs Job](#) on page 7-10
- [Dictionary Impacts Report](#) on page 8-3
- [Maintaining Predictory Terms and Relationships](#) on page 8-8
- [Classifying Terms Manually](#) on page 10-5
- [Promoting and Demoting VTAs](#) on page 11-2
- [Reclassifying and Declassifying Verbatim Terms](#) on page 11-6
- [Modifying Repository Data in the Maintain Repository Data Window](#) on page 12-9

Filtering and Multirecord Selection for Task Allocation by Term

Task Allocation now includes a Filter button and standard TMS filter functionality to limit the terms displayed. In addition, you can select multiple terms at the same time using either Shift+Click or Ctrl+Click to assign the terms. See ["Using Manual Allocation by Term"](#) on page 9-7.

New Informative Note Attribute for Use in RDC Special Listings

The Oracle Clinical Remote Data Capture (RDC) Option Release 4.5.3 and above has a Special Listings feature that displays concomitant medications and adverse events for patients. This feature is available only when TMS is integrated with Oracle Clinical and displays parent questions mapped to TMS dictionaries with the question responses to all other questions in the parent question's question group that are defined as nonderived and displayed.

By default, the system displays the TMS dictionary name in the action drop-down list on the RDC Home page, Casebooks page, and elsewhere. You can change this to the text of your choice by defining a new type of Informative Note called RDC Action Text for each dictionary. The system substitutes the text you supply in the Value field of the Informative Note for the dictionary name; see ["RDC Action Informative Notes"](#) on page 7-27.

See also ["Planning Question Groups and DCMs for RDC Special Listings"](#) on page 4-15.

New Features in Release 4.6

Release 4.6 includes the following new features described in the following sections:

- [Internal Actions](#) on page xxix

In addition, see ["Documentation Changes"](#) on page xxxiv for more information.

Internal Actions

Until Release 4.6, all Actions were *external*. They were sent to an external source data system to convey information and request an Action on the part of an external system user. Release 4.6 introduces Internal Actions that are used inside TMS. They serve two purposes:

- Internal Actions allow approving Action assignments before they are sent to the external system. You can now require approval of Action assignments for a domain and dictionary combination the same way you can require the approval of VTAs.

- Internal Actions allow users to indicate that they are working on a term so that other users do not duplicate their efforts.

In addition, external Actions have been renamed: Conditional Actions are now called Answerable Actions, Never Expire Actions are now called Unanswerable Actions, and Single Term Actions are now called Discrepancy Comments.

The new windows that support this feature are Approve Action Assignments (in the Omission Management menu) and Maintain Action Assignments (in the VTA Maintenance menu).

For more information see ["Defining and Using Actions"](#) on page 7-6 and ["Approving Action Assignments"](#) on page 10-14.

Task Allocation and Activity Lists

Beginning in Release 4.6, you can explicitly allocate tasks to specific users, including classifying omissions and approving unapproved VTAs and Actions. You can use the following methods:

- Direct allocation: The system allocates tasks automatically as soon as they are created.

- Automatic pool allocation: The system allocates multiple tasks to one or more users at a time according to your specifications.

- Manual pool allocation: You manually assign tasks to users.

Using the Filter pop-up, you can create *activity lists* of the omissions, unapproved VTAs, or unapproved Actions that are assigned to you and use these lists as your daily "To Do" list. If you have the necessary privileges, you can create activity lists for other users to track their progress.

Task allocation is optional. You can use activity lists to save lists of tasks that satisfy filter criteria you set, even if you are not using task allocation.

There is a new menu item called Task Allocation that includes four new windows and a new report, respectively: Maintain User Task Allocation Pools, Task Allocation by Term, Task Allocation by Quantity, Maintain Direct Allocation Error Logs, and Unfulfillable Assignments. Two new windows under Omission Management support activity lists: Activity List and Maintain Activities.

See Chapter 9, "Allocating Tasks to TMS Users" and "Setting Filters and Creating Activity Lists" on page 10-24, "Using Activity Lists" on page 10-24, and "Maintaining Activity Lists" on page 10-28 for more information.

Term Status History

Release 4.6 introduces new term statuses and sub-statuses that track a term's process throughout its life cycle. You can see a term's current status and sub-status in the following windows:

- Classify Omissions
- Reclassify Verbatim Terms
- Approve VTAs
- VT History
- Activity List
- VT Simple Search (HTML Browser)
- VT Advanced Search (HTML Browser)
- VT Details (HTML Browser)

You can bring up a display of a term's complete status history by clicking the Status/Notes button from most of the windows listed above. In the non-browser windows you can add a note when you change a term's status. (See Chapter 10, "Omission Management" and the sections on each of the windows above.)

Data Access Groups

You can create Data Access Groups (DAGs), which allow you to selectively restrict the data users can see. As before, database roles determine which windows a user can access. DAG assignments dictate the data users can see in those windows and whether they can make changes to the data.

You can also designate a user as a superuser. Superusers can see and operate on all data and cannot belong to a DAG (the pre-4.6 behavior).

When you define a DAG you can make the following specifications:

- You can specify the dictionary/domain combinations for which users in the group can see data.
- You can specify whether users in the group can make changes or only read data.
- You can specify whether users in the group can see data that originated in one or more specific external systems, and, for example, from a particular project or study.
- You can specify whether users in the group can see data that originated in one or more specific external system databases.

- You can specify whether a user can see only his or her own task assignments and/or those of which specific other users.

You define a DAG and then assign users to it. A single user can be assigned to multiple DAGs. The user then has access to the sum of data allowed by all his or her DAG assignments.

There is a new menu item called Security that supports DAGs. It includes the original, now modified, Define Users window as well as new windows Define Security Columns and Maintain DAGs. It also includes a new report, DAG and Setting Inconsistencies and a new job, Create/Drop EXT_VALUE Indexes.

See "Using Data Access Groups (DAGs) for Security" on page 3-9.

Advanced Term History and Dictionary Release Labeling

Release 4.6 introduces dictionary Release Labeling and a new type of named relation called Release Label (RL). When you define a dictionary, you must now specify a label prefix for the dictionary. Each time you run Activation, TMS applies a Release Label to the successfully activated terms and relations in the Activation Group. The label consists of the prefix you define plus a build number that represents the number of times Activation has been run on the dictionary. For example, if you defined a label prefix of `3.0.` for WHO-Drug 3.0, the system would label the first Activation Group `3.0.1`; the system would label the second Activation Group `3.0.2`.

When you upgrade to Release 4.6 from an earlier version of TMS, the upgrade process looks for a dictionary version-type Informative Note for each dictionary you have already defined and, if there is one, populates the label prefix with that value. For each existing virtual dictionary, the upgrade process creates a Release Label that consists of the prefix (if any) and an appended number: either 1 or, if there are more than one virtual dictionaries with no dictionary version Informative Note value (or the same value) a unique number among those virtual dictionaries. The upgrade process inserts a period/full stop (.) between the prefix and the appended number if the prefix does not already include one.

You can use named relations of type Release Label to record the relationship between a term in one dictionary or dictionary version to another, for complex changes such as term merging or term splitting that TMS cannot maintain automatically. For example, if a new release of MedDRA splits Term A into Term A and Term B, you can create a named relation of type Release Label between Term A in the older dictionary version and Term B in the newer dictionary version.

You can view term history, including Release Label named relations, in the HTML Browser. Do a search in the Terminology tab and then click the term to see the Term Details window with a Related Release Label section.

One new window supports this feature: Release Label Authoring under the Repository Maintenance menu.

See "Defining a Dictionary" on page 6-17 for information on defining a label prefix. See "Defining Named Relations" on page 7-16 for information on defining named relations of type Release Label. See "Using the Release Label Authoring Window" on page 12-30 for information on creating a Release Label named relation between two terms.

Dictionary Upgrade Analysis

In Release 4.6 you can analyze the impact that upgrading to a new version of a dictionary will have on your existing VTAs and omissions before activating it. A new job performs many of the changes you will need to make. Several reports allow you to see the changes, and a new form allows you to see changes and to make changes

manually to terms and relations, including VTAs, while they are still in the predictory tables to avoid problems during and after Activation.

There is a new menu item, Dictionary Upgrade, with a new window and several new reports and jobs.

See Chapter 8, "Upgrading to a New Dictionary Version" for more information.

Filter Dictionaries: Support for MedDRA SMQs

Release 4.6 supports MedDRA Standardized MedDRA Queries (SMQs) with a new dictionary type called a filter dictionary and a new Informative Note type called algorithm Informative Note. In the HTML Browser you can search for source terms using MedDRA SMQs and retrieve external system information about the source terms.

The HTML Browser includes several new screens to support this feature.

See "Filter Dictionaries" on page 6-4 and "Searching for Terms Using Filter Dictionaries" on page 14-11 for more information.

Disconnected System Integration (DSI) Enhancements

Release 4.6 introduces the following DSI enhancements. See "Using TMS with Disconnected Systems" on page 5-3 for more information.

- New and changed Informative Notes of type Workflow associated with verbatim terms are now included in both imports and exports for both source and sponsor sites, and are synchronized across instances.

- New jobs allow you to export and import data for a particular X Area from a single instance instead of all instances.

- You can now exchange data between two UTF-8 databases as well as two Western European (WE) character databases. Both the sponsor and source sites must support the same character set.

- DSI uses the codelist XAPPVTA at source as well as sponsor sites to control whether export files contain unapproved VTAs.

- Only superusers can run DSI jobs. (We made this change to accommodate Data Access Group (DAG) security.)

- Before you can set up DSI, you must initialize your instance as either a DSI source site or a DSI sponsor site.

- When you define a disconnected system integration on a source instance, TMS now displays a list of values for the X Area Name. It includes all X Areas defined on the external sponsor system and the database specified in the Definitions tab.

> **Note:** The XML schemas required for both sponsor and source sites have changed to support these enhancements. See Appendix B, "Disconnected System Integration XML Files."

For more information, see "Using TMS with Disconnected Systems" on page 5-3.

New Reports and New Appearance for All Reports

Release 4.6 uses Oracle XML Publisher to format reports. All reports have a new appearance. In addition, you can customize the template to change the appearance of reports; see "Customizing Report Templates" on page 7-4 for more information.

Release 4.6 includes the following new reports:

- DAG and Settings Inconsistencies Report on page 3-16
- Dictionary Version and Release Information Report on page 12-35
- Dictionary Impacts Report on page 8-3
- Predict VTA/VTI Report on page 8-5
- Site Impacts Report on page 8-6
- Omission Impacts Report on page 8-7
- Cross Dictionary Impacts Report on page 8-8
- Unfulfillable Assignments Report on page 9-9
- X Areas with Outstanding Changes Report on page 11-19
- Verbatim Term Modifications Report on page 11-18 (This report combines and replaces the three classifications reports.)

New HTML Browser Functionality

Release 4.6 expands HTML Browser functionality to complement several new features:

- Filter dictionaries
- Status history
- Status notes

See "Searching for Terms Using Filter Dictionaries" on page 14-11 and "Viewing VTA and Omission Status Information" on page 14-22 for more information.

In addition, it is now possible to search for omissions as well as verbatim terms in the HTML Browser. You perform this action from a new Verbatim Term Status tab. See "Performing a Simple Verbatim Term Status Search" on page 14-22 for more information.

Minor Enhancements

Release 4.6 includes the following minor enhancements.

Applying Informative Notes to Omissions

In the Classify VT Omissions window, you can now apply Informative Notes to unique occurrences of verbatim term omissions (VTOs) in a domain/dictionary combination. See "Using the Status/Notes Pop-up Window" on page 10-21 for more information.

Restricting Informative Note Attributes to Specific Dictionaries

When you define new Informative Note attributes you must now specify one or more dictionaries within which they can be used. Then when you add an Informative Note to a term, relation, or dictionary, TMS allows you to use only Informative Note attributes that have been specified for use with the appropriate dictionary.

When you upgrade to Release 4.6 from an earlier version, the upgrade process populates the new "Applies To" field for your existing Informative Note attributes with the dictionaries with which they are currently used. To use an Informative Note attribute with additional dictionaries, use the Define Dictionary Informative Note Attributes window. See "Defining Informative Note Attributes" on page 7-22 for more information.

Creating Users for Loading Dictionaries and for Administration Tasks

Due to changes required to implement Data Access Groups, you no longer create a user during dictionary loading for the purpose of loading the dictionary. You can now grant dictionary loading privileges to any user through the TMS Define User window by setting the new Load User check box to Yes. See "Defining Users" on page 3-5 for more information.

The tmsadduser.sql script now creates a user with OPA_ADMIN privileges. This user has access to the Define Users window in the UI and can create all other user accounts.

User Profile Setting for Requiring Approval for VTAs and Actions

A new user profile setting allows you to specify users who can create approved VTAs and Actions even in dictionary/domain combinations where approval is required. See "TMS_PROCESS_DICTIONARY" on page 3-21 for more information.

Declassifying Verbatim Terms in the Approve VTAs Window

Users with Reclassify or Classify privileges can now declassify verbatim terms in the Approve VTAs window. As in the Classify VT Omissions window, users with Classify privileges have a configurable period of time in which they can change their own classifications. Users with Reclassify privileges have no time limit. See "Understanding the Approve VTAs Window" on page 10-11 for more information.

Displaying a Meaningful Value for X Areas

You can write a function to replace the X Area ID with a meaningful value for X Areas in reports and on screen. For Oracle Clinical, the X Area is a study and TMS now displays the study name instead of the X Area ID. If you prefer, you can write a function to display a different value for Oracle Clinical X Areas. See "Defining the External System in TMS" on page 5-30 for more information.

External Value Length Increased to 500 Characters

The user-defined external values (and those predefined for Oracle Clinical) have been increased in size so that you can store values of up to 500 characters. The previous limit was 30 characters.

Users with Browse-Only Privileges Do Not See the Source Data Icon in the HTML Browser

Users with browse-only privileges now do not see the Source Data icon.

Documentation Changes

We documented the new features and made the following changes to the *Oracle Thesaurus Management System User's Guide*:

- In the chapters in the *Oracle Thesaurus Management System User's Guide* that correspond to menu items, reports are grouped at the end of chapters in sections with the name of each report as the section heading.

- Some new windows do not have field-level online help. However, you can click the More button to see the online help for the window as a whole, including a description of each field.

New Features in Release 4.5.2

The enhancements included in Release 4.5.2 relate primarily to integration with Oracle Clinical:

-

-

-

In addition, the Define Users window has been moved to a new location: the newly created Security menu. Therefore the TMS_DEFINE_PRIV privilege is no longer required to access the Define Users window; only OPA_ADMIN is required. The documentation in [Chapter 3, "Administration"](#) has been updated accordingly.

Deriving Data from Two Dictionaries

To allow deriving data from two dictionaries for a single term collected in Oracle Clinical, Release 4.5.2 makes two changes:

- Question set parent questions can be defined as derived. Their value can be populated by an Oracle Clinical derivation procedure as necessary, including reading the value of a derived child question set question and writing it to the value of a derived parent question.

- If there are derived parent question values after Oracle Clinical procedures have been executed during Batch Validation, the TMS portion of Batch Validation is run again just on those values, and TMS information is derived for their child questions during the same Batch Validation job.

This functionality may also be useful in other circumstances. See ["Using a Derived Question as a Parent Question"](#) on page 4-13.

Parameterizing Batch Validation Failure

Until Release 4.6, if Oracle Clinical Batch Validation encountered a serious problem during the TMS processing portion, the whole Batch Validation process failed. You can now choose whether or not you want this to happen by setting a new reference codelist value in Oracle Clinical. Also, if the errors encountered during TMS processing are not serious, Batch Validation will not fail, but will display a warning.

For more information, see ["OCL_STATE"](#) on page 4-22.

Running Only the TMS Portion of Batch Validation

A new job is available in Oracle Clinical for running only the TMS portion of Batch Validation. See ["Running Only the TMS Portion of Batch Validation"](#) on page 4-5.

New Installer

The installation and migration of Release 4.5.2 requires a new Installer, which handles a fresh installation as well as migration from previous versions 4.5.x and above.

New Features in Release 4.5.1

Release 4.5.1 included the following new features:

-

-

-

-

Disconnected System Integration

The Disconnected System Integration (DSI) feature provides a customized XML-based batch data load process that enables you to use TMS as a central dictionary classification tool for systems located remotely and operated by different businesses or research organizations.

DSI is designed specifically to allow a pharmaceutical company sponsoring a clinical trial that is being conducted by multiple contract research organizations (CROs), to use TMS as a central classification tool for terminology quality control. CROs send their source data to the sponsor, with contextual information (such as patient ID and document number) and preliminary classifications. The import process at the sponsor site compares the CRO's classifications to the sponsor's and enters data in the sponsor database and/or creates warnings or errors associated with faulty data. When the sponsor sends the data back to the CRO, the CRO's import process overwrites CRO classifications with those of the sponsor, if there are conflicts, and passes on any Actions created at the sponsor site for particular terms.

The sponsor instance stores all CRO source terms and contextual information, and runs TMS Synchronization on CRO data. However, the sponsor instance does not have a UI drill-down capability for CRO data, and, if the sponsor is using Oracle Clinical, OC Batch Validation does not run on CRO data.

The sponsor must use TMS. The CRO must use either TMS or another dictionary coding system that is capable of processing XML files in the required DSI structure. The external source data system feeding data into TMS can be Oracle Clinical, the Oracle Adverse Events Reporting System, or a third-party system.

For more information, see "Using TMS with Disconnected Systems" on page 5-3.

> **Note:** The Disconnected System Integration feature does not support the UTF8 character set.

Fixing Classification Mistakes

Until Release 4.5.1, when a TMS user made a mistake classifying a verbatim term, he or she required reclassification privileges to fix the problem. In Release 4.5.1, the user has a certain number of hours in which to fix the mistake, even without reclassification privileges.

There is a new short value in the TMS_CONFIGURATION reference codelist called MTRECCLASSHOUR. The long value sets the number of hours after the initial classification during which users can change the classification.

During that time, the user has access to the Reclassify Verbatim Terms window, but the window displays only VTAs created by that user. The user will be able to change only those classifications that he or she created within the time limit set in the reference codelist.

This information is included in "Look for a Good Match" on page 10-5 and in "Reclassifying and Declassifying Verbatim Terms" on page 11-6.

Deleting Domain Elements

Until Release 4.5.1, you could not delete a TMS domain element in Oracle Clinical if any classifications or omissions existed for that domain/dictionary combination in any study. Now you can delete a project or study's domain element if no classifications or omissions exist for that particular project or study.

This information is included in "Linking an Oracle Clinical Study to TMS" on page 4-17.

New Symmetric Replication Process

Release 4.5.1 includes a new model for symmetric replication. In the original multi-master approach, multiple instances replicated information to each other. In the new approach, which uses materialized views, local instances send new data to a single master instance first, and then data on the master is replicated to all sites. The new approach prevents the No Data Found errors that sometimes occurred in the old model.

This change requires extra steps during the installation of Release 4.5.1, but thereafter the difference in approach is transparent to the user.

The new symmetric replication model was introduced in Release 4.0.6.12 and was patched in 4.0.6.14. Therefore the installation process is different if you are upgrading from either of those versions of TMS than if you are upgrading from any other version (Release 4.5 did not include the new symmetric replication model). The installation steps for each type of upgrade are included in the release notes for Release 4.5.1.

1

Overview

This introduction toOracle Thesaurus Management System (TMS) includes the following concepts:

- Purpose on page 1-1
- Usage Models on page 1-2
- Dictionary Definition and Customization on page 1-3
- The API on page 1-4

Purpose

Pharmaceutical companies that develop new drugs are required to produce reports that document the effects of the drug. However, the data used to produce these reports, including information on health conditions and medications, can be hard to analyze because the terminology is inconsistent. Even when a company uses a dictionary of appropriate terms provided by an organization such as the World Health Organization, data collected during the study may be inconsistent with its terminology for a variety of reasons:

- Medical personnel may use terminology different from that of the dictionary.
- Companies may have developed new drugs since the dictionary's creation.
- A study may require more precise terminology for a health condition than is available in the dictionary.
- Medical personnel may misspell drug names or health conditions.
- Data entry personnel may misinterpret or mistype the data.

TMS can transform this data into a consistent terminology that statisticians can analyze more easily.

TMS refers to an item of data initially entered or loaded into the system—drug or health condition information or other data item—as a **verbatim term** (**VT**). TMS maps verbatim terms to dictionary terms in a process called **classification**. TMS automatically classifies verbatim terms that directly match a dictionary term, and enables you to classify variant or misspelled terms manually. You can choose to have future occurrences of the same verbatim term automatically classified the same way, or you can choose to require manual classification, with the previous classification presented. You can also add terms to vendor-supplied dictionaries and modify your own terms at any time.

Usage Models

TMS serves as the single terminology repository for clinical studies, and can be available to all applications used in studies. Particular applications, or **external system**s, are more or less compatible with TMS. Depending on the degree of compatibility, TMS can be installed in different degrees of integration with the external system(s).

The more fully TMS integrates with your clinical study application, the better it serves you. However, you can still benefit from TMS if your other software permits only partial integration, or even no integration.

Full Integration

Full integration requires the installation of TMS objects in the external system, a level of integration that is recommended if the external system is running in a stable global Oracle environment. In full integration, TMS maintains external data in both the tms_source_terms table and the tms_vt_omissions table.

A fully integrated system benefits from the full range of TMS functionality. It feeds source terms to TMS with contextual data you specify (such as Patient and Document Number) so that if you reclassify or declassify a term in TMS, TMS can send information about each affected source term back to the source data system.

In TMS, you run Autoclassification, manually classify remaining terms (omissions), assign Actions, and reclassify or declassify as necessary. You specify the information you want to derive from TMS for each source term, and TMS sends that data to the source data system associated with each source term.

See "Full Integration" on page 5-2 for more information.

Partial Integration

If full integration is not desirable—for example, if space is an issue—you may want to set up TMS with only partial integration to your external source data system. In partial integration, TMS maintains external data in the tms_vt_omissions table only. It does not store any data in the tms_source_terms table.

The external system feeds data to TMS, and TMS returns derived data and omissions. You can classify terms manually and perform all other functions within TMS. However, because the tms_source_terms table is not used, TMS has no external system information (such as patient ID or document number) associated with source terms and therefore cannot associate derived terms with source terms after the initial classification.

See "Partial Integration" on page 5-3 for more information.

No Integration

Even if TMS has no knowledge of your external system, you can use TMS as a dictionary repository. You can use TMS to access, add, and modify dictionary terms. You can take advantage of TMS's flexible hierarchical dictionary structure and data validation. However, TMS functionality in this scenario is limited in the following ways:

- TMS cannot receive terms directly from the external system, but you can load them or enter them manually in TMS.

- TMS cannot handle omissions.

- The external system must handle the impact of reclassification and de-classification on derived data.

- High-level classification is not available.

Following is one scenario for using TMS with no integration to the external system:

1. Check for matches between the external system's source data and the TMS repository.

2. Load or create VTAs in TMS for verbatim terms in the external system that did not map to a term in the TMS repository.

The no-integration approach is recommended if the external system does not run on an Oracle database.

Dictionary Definition and Customization

TMS supports the inclusion of any number of dictionaries in its repository. You can use dictionaries supplied by a vendor such as WHO (an **external dictionary**) and/or convert a legacy company dictionary. You can customize external dictionaries by adding company terms. You can customize dictionaries in multiple ways, all available simultaneously, by using TMS domains.

Flexible Definition

You can create a dictionary in which terms inherit the relations of the levels to which they belong, or one in which you store all the terms on a single level and use the relation itself to define the connection between the terms. For dictionaries with multiple levels, you can define any number of levels in a given dictionary; you can group levels, and you can create a hierarchy of any shape or degree of complexity. You define both ends of the relation between each pair of related levels, specifying:

- whether the relation is optional in each direction, and in some cases, whether the relation is mandatory in each direction

- whether a single term in one level can map to more than one term in the other level, in each direction

- whether the parent-level end of the relation can have multiple links, whether a term in the child level must have a Primary Link to a single term in the parent level

- the currency of viewable data. Base dictionaries contain all dictionary information that is currently not expired in the TMS repository, while virtual dictionaries can represent the state of a dictionary at a particular point in time.

See Chapter 6, "Defining and Loading Dictionaries" for more information.

TMS automatically performs validation to ensure that links between terms do not violate the level relations you have defined.

Customization

You can add company terms to external dictionaries if, for example, you need to use the name of a new drug that is not included in the external dictionary.

If you need several more specific terms instead of one more general term provided by the dictionary, you can mark the external dictionary term as Unapproved to ensure that it is not used for classification, forcing the use of the more specific company terms instead.

Different studies or projects may have different terminology needs. TMS addresses this issue with the concept of user-defined TMS **domain**s. You associate external dictionaries with TMS domains and customize them differently in each domain, creating TMS domain-specific terms and relations. See Chapter 6, "Defining and Loading Dictionaries" for more information on defining and using domains in TMS.

The API

TMS includes PL/SQL packages to accomplish all TMS tasks, providing the ability to interface with many external applications. Your external system calls these packages instead of manipulating the database directly, thus avoiding database corruption. The API is documented in the *Oracle Thesaurus Management System Technical Reference Manual*. Contact Oracle Support to receive a copy.

The following table shows a subset of the API packages, and the purpose of each one.

Table 1–1   Sample API Packages and Their Purpose

Package	Purpose
tms_user_classification	Creates accepted or misspelled verbatim term assignments (VTAs), which map the corresponding correct or misspelled verbatim terms to dictionary terms.
tms_user_action	Enables you to specify comments for unclassified VTs, which can be propagated to the external system's discrepancy management system.
tms_user_reclassification	Reclassifies VT to another dictionary term or declassifies the VT.
tms_user_fullautocode	In fully integrated external systems, this package matches a new or updated verbatim term with a dictionary term, and returns values.
tms_user_autocode	In partially integrated external systems, this package matches a new or updated verbatim term with a dictionary term.

2

Common Functions

This section introduces the basics of launching and using Oracle Thesaurus Management System (TMS), including the tasks that you perform in more than one TMS window. These topics include:

Launching the Application

TMS runs in a Java-based window that you initiate from the launch page. To start a session:

1. Open a browser window and connect to the following URL:

   `http://server.domain/path_variable/launch.htm`

   where *server.domain* is the TMS application server and *path_variable* is the configuration variable that you set during the TMS front end installation; for example, `opa50`. (See the *Oracle Thesaurus Management System Installation Guide*.)

   Your browser window loads the Oracle Health Sciences Applications launch page.

2. Click **Login**. The browser launches two windows: an Oracle Developer Forms Runtime window, and a small browser window containing the warning not to close the window.

3. In the Forms Runtime window, enter your user name, password, and the TMS database to which you want to connect, then click **OK**. If you have an account on the selected database and have entered the correct password, the Forms Runtime window displays the TMS Navigator window.

   > **Note for Internet Explorer Users:** Do not open another Thesaurus Management System session from this session. To start another Thesaurus Management System session, open it from the Windows Start menu or from the Internet Explorer desktop icon. If you instead start a new session by selecting New from the File menu or by using the shortcut combination Ctrl+N, the new session may be unstable and unpredictable.

Changing Your Database Password

This section describes how to change your own password in the TMS database. You can only change your password for the database to which you are connected; you must repeat these steps for every database in which you want to update your password.

There are three methods of changing your password, depending on your privileges and on whether Oracle Clinical is integrated with TMS:

- If You Have Administrator Privileges in This Database

- If Oracle Clinical Is Not Installed in This Database

- If Your Database is Integrated With Oracle Clinical

If You Have Administrator Privileges in This Database
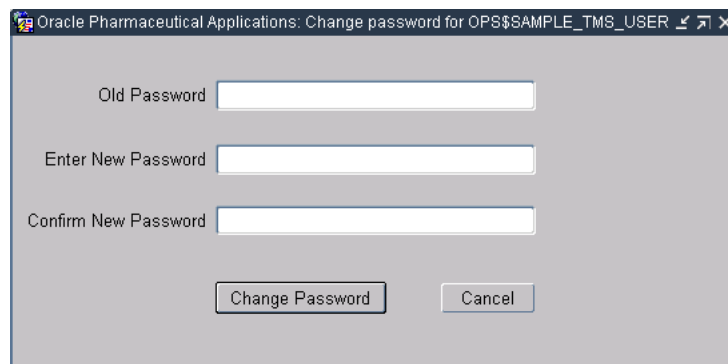
If you have privileges equivalent to the SYSTEM user in this database, you can change your own password (and update the passwords of other users in the database) from a SQL*Plus prompt. This process is an administrator-level activity, so see "Changing a User's Password" on page 3-8.

If Oracle Clinical Is Not Installed in This Database

If you are running TMS in a standalone configuration, follow the procedure in this section to update your database password.

1. From the TMS main window, select **File**, then choose **Change DB Password**. The Change Password for *User_Name* window opens (Figure 2–1).

Figure 2–1   Change Password Window

2. Enter your old password, then enter and confirm your new password.

3. Click **Change Password**. The system validates your old and new password entries. The system returns "Password changed" if all entries are correct, or a descriptive error message if the old password is incorrect or the two new password entries do not match.

   If you click **Cancel**, the system retains your old database password and closes the Change Password window.

If Your Database is Integrated With Oracle Clinical

If you are connected to a database that is integrated with Oracle Clinical, you can change your own password either by selecting **Options**, then **Change DB Password** from the Navigator window or from the Change Database Password window in

Oracle Clinical (From Oracle Clinical's **Admin** menu, select **Users**, then **Database Password**). However, once you launch the Change Database Password window, the system deactivates the **Change DB Password** selection under the **File** menu for the rest of your session.

> **Note:** When you use the Oracle Clinical Change Database Password window to change your password in an integrated database, the system disables TMS windows for the rest of your session. You must exit and reenter TMS before you can open any windows.

Using TMS Windows

All TMS windows, except those used for creating reports and other batch jobs, provide the same set of icons, keystrokes, and menu options to perform common functions. Windows used for reports and batch jobs provide another set. Appendix C, "GUI Quick Reference" contains a complete set of icons, keystrokes, and menu options for functions in both types of windows.

This section contains additional information on the following tasks in the GUI.

- "Understanding Color Indicators" on page 2-3
- "Changing the Display" on page 2-3
- "Using the Tree Structure" on page 2-4
- "Options Menu" on page 2-8
- "Showing the Environment" on page 2-8
- "Selecting Multiple Records" on page 2-9
- "Changing the Database Connection" on page 2-9
- "Printing the Contents of a TMS Window" on page 2-9
- "Querying in Windows" on page 2-9
- "Using the Favorites Menu" on page 2-11

Understanding Color Indicators

TMS conveys some information using color cues.

Table 2–1   Color Indicators in the User Interface

Function	Color Displayed	What is Colored
Current record	Blue	Area to left of record
Query mode	Light blue	Field background
Mandatory field (batch jobs and reports)	Red	Text in field
Drill-down information available	Blue	Text in field
Read-only information	Gray	Field background

Changing the Display

You can change the way data is displayed by selecting single- or multi-record display. You can change which data is displayed by using the tree structure and querying. See

Single/Multi-record Display Option

In general, the single-record display is more useful when you are modifying data because it allows you to see all the information for a record at once, without scrolling. The multi-record display is useful when you are querying; you can scroll up and down to quickly see all the records retrieved, and scroll across to see the complete information for each record.

Sort Order

You can sort records on any active column label (column labels that appear three-dimensional, with the column heading in boldface). Click once to sort on a column in ascending order. Click again to sort in descending order. Click a third time to return to the default unsorted state. A sort icon appears in the label after you use it to sort. The system sorts numbers by the first digits, no matter how many digits the number contains, so that 12345 comes before 543 in ascending order.

Using the Tree Structure

The tree structure that appears on the left of many windows serves different purposes depending on the window, but its behavior is consistent throughout TMS:

- Nodes, or branches, of the tree structure expand and collapse when you click the boxed + or - icons in the tree structure, use the +, -, ++, or -- icons in the toolbar, or use the menu options in the **Navigation** menu.

- The system populates the right side of the window according to what you highlight in the tree structure.

- The display of the tree structure changes according to changes you make on the right side of the window.

- In some windows you highlight a level in the tree structure and execute a query in the current-level block (usually the middle text block on the right side of the screen) to display data.

- When a tree structure node is the active field, dotted white lines appear along the top and bottom of the blue background, highlighting it. When no dotted white lines are displayed in the highlighted node, the active field is on the right side of the window.

The system uses the tree structure for two purposes: the menu and dictionary hierarchy.

The Navigator

When you first open TMS, the Navigator window opens. This window displays the main menu tree and, if you have it installed, Oracle Clinical as well. Each item under the TMS node represents a TMS subsystem; for example, from the **Repository Maintenance** menu you can open windows for updating term and relation data in the TMS repository. See Figure 2–2 for an overview of the menus that appear in the Navigator.

In some cases the objects you create also appear in the navigator tree: dictionaries, levels, and Activation Groups that you have already defined appear in the tree structure; if you click them, the system displays the definition you created. To create a new object, highlight the menu option above it in the tree structure and select **Insert Record**.

To open a window or run a batch job or report, click either the icon or the text.

Figure 2–2   The TMS Navigator
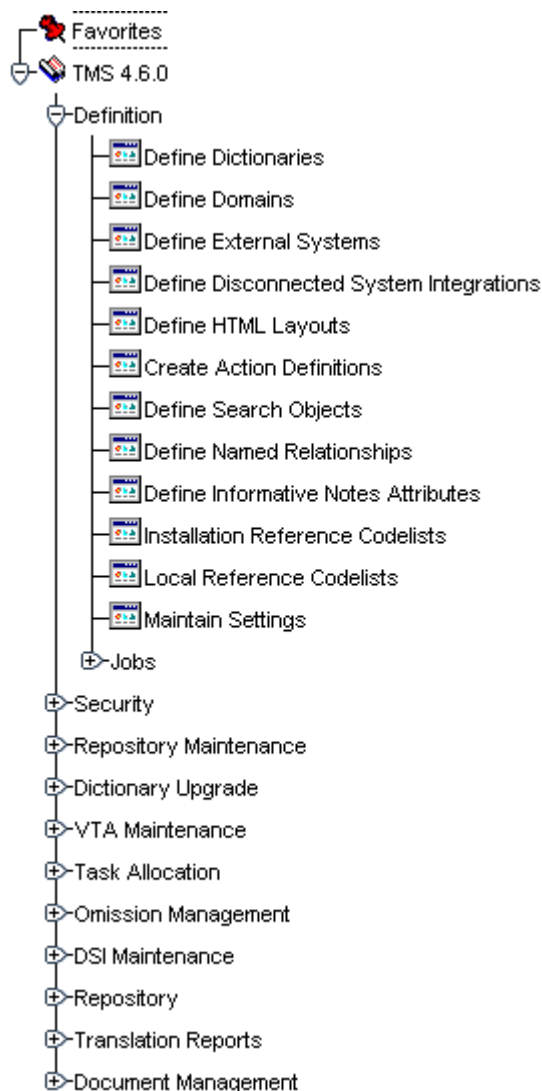
The menu items in the navigator are covered in TMS documentation as follows:

- **Favorites**. Save shortcuts for commonly used windows in the **Favorites** menu. See "Using the Favorites Menu" on page 2-11.

- **Oracle Clinical**. If you integrate TMS with Oracle Clinical, the OCL menu options appear above the TMS node. See Chapter 4 for TMS/Oracle Clinical integration information.

- **Define Dictionaries and Domains**. Define base and virtual dictionaries from the Define Dictionaries window, and link dictionaries together. The Define Domains window allows you to create domains and assign dictionaries to them. See Chapter 6.

- **Defining External System Integration**. Chapter 4 covers integration with Oracle Clinical. Chapter 5 covers integrating with external systems other than Oracle Clinical, including Disconnected System Integration.

- **Other Definitional Tasks**. Chapter 7 explains the other Definition windows, where you define elements for use in assigning VTs, attributes of the external systems with which TMS exchanges data, and reference codelists.

- **Security and Administration**. Chapter 3 covers reference codelists, other settings, and security.

- **Repository Maintenance**. Chapter 12 describes how you can modify data in the TMS repository within a single dictionary or between multiple dictionaries, and how to view dictionary loading error logs.

- **Dictionary Upgrade**. Chapter 8 describes how to use TMS features to control the impact of upgrading to a new version of a vendor-supplied dictionary on your repository.

- **Verbatim Term Assignment Maintenance**. The windows and jobs under **VTA Maintenance** enable you to promote and demote VTAs, maintain Actions, reclassify verbatim terms, perform High-Level Reclassification, and copy domains. You can also run reports about VTA data, such as the Nonapproved VTAs Report, the Classification to a New Domain Report, and the three Classification Changes reports. See Chapter 11.

- **Task Allocation**. Chapter 9 describes how to set up task allocation and how to allocate tasks—approving actions and VTAs and classifying omissions—to users.

- **Omission Management**. Chapter 10 describes how to classify VT Omissions, approve VTAs, and run Omission-related reports.

- **Disconnected System Integration Maintenance** (DSI). Chapter 5 describes how to use DSI to exchange information with disconnected systems.

- **Repository**. Users without data modification privileges may still view repository data using the windows under the **Repository** node. These selections are covered in Chapter 13.

- **Translation Reports** help you identify which term and relation data are not consistent between two dictionaries in a Translation Derivation Link. For background on translation derivation, see "Defining a Translation Derivation Link" on page 6-36. For specific information on the three translation reports, see "Translation Reports to Identify Inconsistent Data" on page 12-36.

HTML Browser. Chapter 14 describes how to use the HTML Browser, which has a separate login and user interface.

Dictionary Hierarchy Tree Structure

Dictionary levels, and the relations between them, are displayed in many windows through the tree structure (Figure 2–3, "Hierarchical Structure in the Maintain Repository Data Window").

Figure 2–3   Hierarchical Structure in the Maintain Repository Data Window
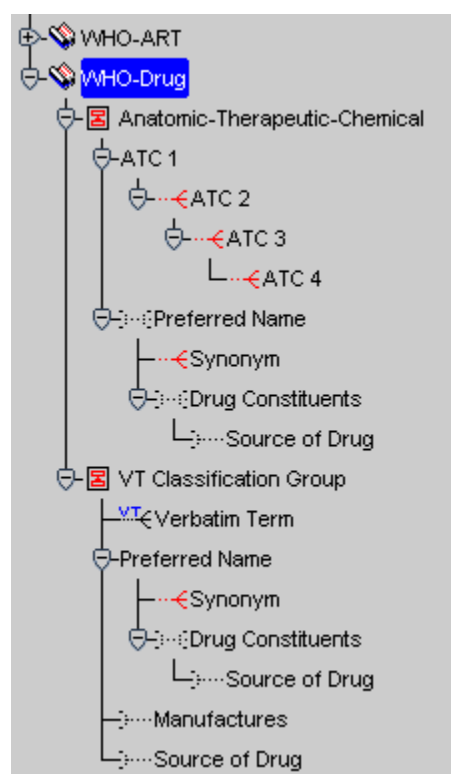
Using the Tree Structure to See Data The Maintain Repository Data window includes a dictionary's tree structure on the left side and three blocks on the right. To view data in the blocks of fields, highlight a level in the tree structure and then execute a query on the right. TMS displays terms that you queried for in the level you highlighted. After TMS populates the text block(s), you can highlight a different level in the tree structure and TMS will make that level the current level, with only terms related to the former current term displayed.

Understanding Multiple Displays of One Level In order to show all relations between levels, sometimes TMS displays a level more than once. For example, levels within a group are shown once within the group level, with the relation between the group level and the level above it displayed, and once with the relation between the top sublevel and the level above it displayed. Similarly, if a level has more than one parent level, it is displayed once for each parent. All its child, grandchild, and other levels also appear below it each time.

In the sample WHO-Drug dictionary shown in Figure 2–3, "Hierarchical Structure in the Maintain Repository Data Window", the **Source of Drug** level is shown twice in the **Verbatim Term Classification Group** level. TMS does this because **Source of Drug** has two parent levels: the **Drug Constituents** level and the **Preferred Name** level, and because it is also linked to the ATC group.

In the tree structure, select the level display that you prefer. Some levels may be displayed twice or more because they have a relation to more than one parent level. To see terms in a parent level as well as the current level, you choose the display of the current level that is linked to the parent level you want to see.

Understanding Tree Structure Icons TMS uses small icons to convey information about the hierarchy:

Table 2–2   Tree Structure Icons

Icon	Information
VT	Indicates that TMS has created this level to hold verbatim terms and their VTAs.
p	Indicates that a Primary Link is required in a parent level where many cardinality is allowed; that is, more than one term in the parent level can be linked to a single term in the child level, and one of them must be designated as the Primary Link.
pp	Indicates that a Primary Path Link is required in a parent level where many cardinality is allowed.
	Indicates that this level is a group level. Levels within the group (except for the topmost level) are displayed with their end of the relation shown in red. They appear on the same branch in the tree structure.
__	Mandatory relation; terms in the related level must have a relation to a term in the level with a solid relation line.
- -	Optional relation; terms in the related level may or may not have a relation to a term in the level with a dotted relation line.
	Many cardinality relation; terms in the related level may be related to more than one term in the level with a branched (crow's foot) relation line.
__	Single cardinality relation; terms in the related level may not be related to more than one term in the level with a single relation line.

To refresh the tree structure display, select **Navigate**, then choose **Refresh Tree**.

Changing the Sort Order

Many windows enable you to sort records by clicking one of the column headings. Click an active column head—one that looks like a button, and displays the column heading in boldface—once to indicate that you want to sort on it (alphabetically for character fields, numerically for numeric fields). Click again to sort in the opposite order.

> **Note:** TMS sorts numbers by the first digits, no matter how many digits the number contains, so that 12345 comes before 543 in ascending order.

Options Menu

The options included in the **Options** menu change for each window. Common selections under the **Options** menu include **Change Domain**, **Filter**, and **Informative Notes**. Any tasks you can perform using a button are also available in the **Options** menu and keystroke combinations. See Appendix C for a complete list of keyboard shortcuts.

Showing the Environment

TMS displays the current TMS version number and the current database in the title bar. Select **Help**, then **Show Environment** to display the following information:

- User and Instance
- Current Window Name, Module, Block, and Field

Selecting Multiple Records

Several windows in TMS enable you to select more than one record using the standard Windows behavior of Shift+Click for adjacent records and Ctrl+Click for separated records. The windows are High-Level Reclassification, Demote VTAs, and the Assign/Revoke Role subwindow of Define Users.

Changing the Database Connection

Under some conditions, you can change the database to which you are connected without logging out by selecting the **Connect** option from the **File** menu. If your database is integrated with Oracle Clinical, however, TMS disables this feature upon your first visit to an Oracle Clinical window.

Printing the Contents of a TMS Window

You can print a hard copy of any window in TMS by selecting **File**, then **Print** from the TMS menu bar. The generated printout displays the TMS window as it appears on your screen, showing the data that appears in the TMS interface for that window.

> **Note:** The other menu bar option, selecting **File** and then **Print Contents**, has no effect in TMS.

Querying in Windows

TMS follows standard Oracle querying procedures in every window of the user interface. In addition, in many TMS windows you can take advantage of the advanced text retrieval capabilities of Oracle's *inter*Media Text software (formerly known as the Context Server Cartridge), which is integrated into the RDBMS in an Oracle9*i* platform.

About Query Mode

You are in Query mode when you have entered but not yet successfully executed or canceled a query. When you are in Query mode:

- TMS changes the background color for all fields in the block to light blue.

- You cannot change the Query Type (Standard or Context).

- You cannot modify data in any window.

- You cannot select another level in the tree structure.

- You cannot change domains.

To end Query mode, either execute or cancel the query. If you execute the query and no records are retrieved, TMS stays in Query mode. If the query does retrieve records, TMS displays the records and exits Query mode.

Starting a Query

1. Do one of the following:

   - Press F7.

   - Choose **Query**, then select **Open Query**.

   - Click the **Open Query** icon (see Appendix C).

2. Enter the alphanumeric text you want TMS to search for. You can always enter a standard Oracle query. If there is a **Query** list in the TMS window where you are working, you can choose to enter either a Standard Oracle query or a Context query, which requires the *inter*Media Text software.

Entering a Standard Query

Standard Oracle queries recognize the following:

Wildcards   There are two wildcards that standard Oracle queries recognize:

The underscore (_) character is a single character wildcard, which can represent exactly one character in your search. Thus, the query no_e returns records with Nose and Note, but not Noise.

The percent sign (%) is a multi-character wildcard, which can match one or more characters in your search, or even none. A query on %bite% in the **Term** field would retrieve the terms Human Bite, Bite, and Bite on Nose.

SQL Advanced Queries   Enter :*letter* to define a variable in a queryable field, then press F8. In the pop-up box, write a SQL statement. For example, in the **ID** field, enter :a and press F8. In the pop-up box, enter:

:a in (*ID_number_1*, *ID_number_2*)

No Text   To retrieve all possible data, enter no parameters.

Entering a Context Query

Context queries use the *inter*Media Text software, which enables more flexible searches of TMS terms and relations. The Browse Repository Data window and the advanced search pages in the HTML Browser allow you to switch between standard and context queries; see Chapter 14, "Using the HTML Browser"

The Context Server Index is language-specific and configured for English language searches by default. You can change the language preference for the index by running the tmscincontextinx.sql script. See "Enabling Context Searches for Non-English Dictionaries" on page 3-41.

The following table lists four types of operators that context queries use to expand or change the scope of a search.

Table 2–3   Context Query Operators

Name	Operator	Function	Example
Fuzzy	?	Finds words with a similar form; checks for common spelling, keyboard, and OCR errors.	?aspirin finds aspirin, ascriptin, and aspellin
Stem	$	Finds words with the same linguistic stem (root).	$crush finds crush, crushed, and crushing
Soundex	!	Finds words that sound similar.	!hair finds hair and air
About	( )	Finds terms that contain concepts that are related to your query parameter.	(car) finds car, auto, and automobile

> **Note:** Two attributes are available to make fuzzy searches more
> precise. You can set fuzzy_numresults to *n* where *n* is the maximum
> number of results you want. You can also set fuzzy_score to a
> number between 1 and 80 to retrieve only terms that score *n* in
> nearness to the original term.

Because context queries are driven by the *inter*Media text software, you may require a
different syntax for these searches than in the rest of the TMS user interface. For
example, TMS forms and *inter*Media handle hyphenated terms differently. When you
enter a query for a term such as 'musculo-skeletal system' in a TMS form, TMS returns
all of the terms that contain that text. The *inter*Media text option, however, treats the
hyphen character ('-') as an operator between two terms in a search. To search for the
same term in the HTML Browser, you must precede the hyphen with a slash ('\'). The
search becomes:

```
musculo\-skeletal system
```

For more information on using special characters in context searches, see the *Oracle
interMedia User's Guide and Reference* (part number A88786).

Counting the Number of Records Your Query Retrieves

Do one of the following:

- Press Shift+F2.

- Select **Query**, then **Count Hits**.

- Click the **Count Hits** icon (see Appendix C, "GUI Quick Reference").

TMS displays the count at the bottom of the window.

Executing the Query

Do one of the following:

- Press F8.

- Select **Query**, then choose **Execute Query**.

- Click the **Execute Query** icon (see Appendix C, "GUI Quick Reference").

To execute the same query again, press F7 F7 F8, in sequence.

Cancelling the Query

Do one of the following:

- Press Ctrl+Q.

- Select **Query**, then choose **Cancel Query**.

- Click the **Cancel Query** icon (see Appendix C, "GUI Quick Reference").

Using the Favorites Menu

You can add shortcuts to the TMS Navigator window for quicker access to commonly
used forms and jobs. TMS displays these shortcuts under the **Favorites** node, at the top
of the **Navigator** menu tree.

Favorites are private and database-specific: your choices appear only when you are
logged into this TMS database.

Adding Shortcuts

To add a shortcut to the **Favorites** menu:

1. Select **File**, then **Save as Favorite**. TMS prompts you to select a form or batch job.

2. Click **OK**, and click the form or job you want to add to the Favorites. The Save as Favorite window appears.
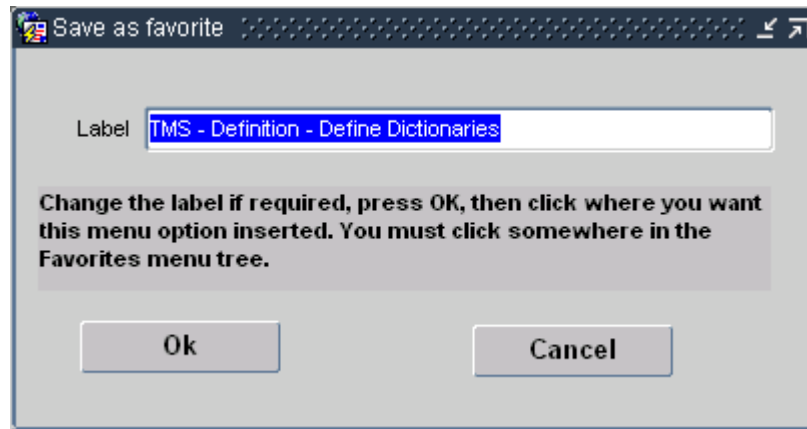
3. Enter a name for the shortcut, and click **OK**.

4. In the **Favorites** menu, click the selection under which you want the new shortcut to appear. To save the new shortcut at the top of the Favorites, or if this is the first shortcut for this system, click the **Favorites** node itself.

    TMS saves the new shortcut under the **Favorites** node in the navigator tree.

Removing Shortcuts

To remove a shortcut from the **Favorites** menu:

1. Select **File**, then choose **Delete Favorite**. TMS prompts you to select a form or batch job to remove from the **Favorites** menu.

2. Click **OK**, and click the **Favorites** shortcut you want to remove. TMS prompts you to confirm this deletion.

3. Click **OK** to delete the shortcut. TMS removes this Favorite from the navigator tree.

Using Online Help

TMS contains online help for every field and window except batch jobs.

To launch online help:

1. Place your cursor in the field for which you want help; if you wish to read online help for the entire window, the cursor can be placed in any field.

> **Note:** The cursor must be active in a field; you must actually click in a field or enter text. It is not enough to have the window open; you will see help for the last window in which you were active.

2. Click the question mark icon at the right of the toolbar, press F1, or select **Help**, and then choose **Help** from the menu. TMS displays the field help pop-up box for the field your cursor is in. Figure 2–4 shows the help box for the **Short Name** field in the Define Dictionaries window.

Figure 2–4   Field Help for Short Name Field in Define Dictionaries
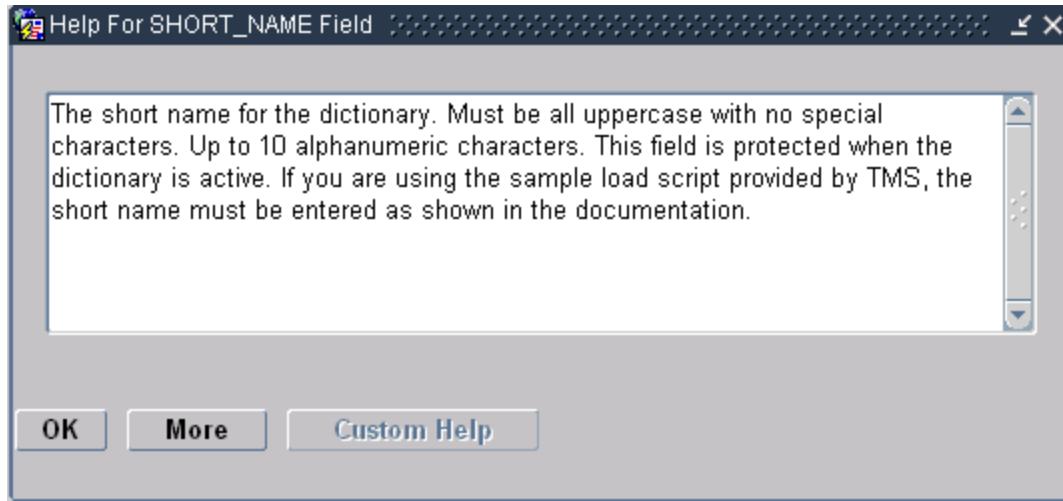
3. If this help is sufficient, click **OK** to close the window. If you want help for the window in general, click **More** in the field help pop-up box. A new browser window opens and loads the appropriate topic in a help file. You can scroll along the file or use hot links to navigate to other topics.

You can also write your own company-specific online help and access it through the **Custom Help** button in the field help window.

Setting Up and Running Reports and Other Batch Jobs

The parameters that appear for particular reports vary; this section lists all possible parameters in the General and Schedule sections of the Batch Job module.

The batch job submission and execution includes:

1. Setting Parameters. See also Using Parameter Sets to Populate Job Parameters

2. Running a Job

3. Checking Job Status

4. Stopping Job Execution, if necessary

Setting Parameters

TMS populates some parameter fields with default values. After the first time the report is executed, the parameters entered when you last ran the report become the default values. You can save your parameters for future use. See "Retrieving a Parameter Set" on page 2-15.

The column of boxes on the right indicates if you can enter a wildcard (%) to search for a valid value for the field. If a box is selected, you can use wildcards.

Fields displayed in red are mandatory; you must enter a value in them.

General Parameters

Many reports include, under General Parameters:

- **Output**: Select a format. Options include PDF, RTF, XML, HTML, and XLS.

- **Display Parameters**: Select **Yes** to display the values of the report-specific parameters in the report output.

Job-Specific Parameters

Many reports or batch jobs have parameters that are unique to the report, and that you can use to customize the report output to suit your needs more closely.

If there are job-specific parameters for your report or batch job, they appear under the Job Specific heading in the Report Submission window.

Context-sensitive online help is not available directly from Report or Batch Job Submission windows. However, you can get information about job-specific parameters as follows.

Online Help   To get job-specific information online, do the following:

1. Go to any window that is not a Report or Batch Job Submission window.

2. Click **Help**, choose **Help**, then select **More**. The TMS online help window opens, displaying information about the window you are on.

3. If you cannot see the **Contents**, **Index**, and **Search** buttons, click the navigation icon:

4. Use the Table of Contents, Index, or Search to find the report.

5. Click the report's hyperlink.

Report Template   is a job-specific parameter in many reports. If your company has created a custom template, it appears in the list of values. The Oracle Template is the default template. See "Customizing Report Templates" on page 7-4. for information on creating and setting up templates.

Scheduling Parameters

Some jobs have parameters that allow you to schedule the job to run in the future:

- **Frequency**. To schedule a single job execution, leave this field blank. To schedule execution on a regular basis, select one of the following from the list: **Hourly**, **Daily**, **Weekly**, **Monthly** or **Every**.

  If you choose **Every**, you can define the interval at which you want the job to run. Two additional fields appear:

  - **Frequency Length**. Enter the interval you want to have pass between job executions. For example, if you want the job to run every week, enter *1* here and enter *Week* in the **Unit** field below. A list of values is available; press F9.

  - **Unit**. From the list select one of the following: **Hour(s)**, **Day(s)**, **Week(s)**, or **Month(s)**. A list of values is available; press F9.

- **Clock**. From the list, select one of the following: **Midnight**, **Noon**, **Now**. Enter the time of day you want the job to run.

- **Date**. From the list, select one of the following: **Today**, **Tomorrow**, **Monday**…**Sunday**.

- **Report Server Name**. The system displays the name of the report server that last ran the report. You can use the same one or enter a different one. If you are running the report or other job for the first time, you may need to click in the line below to make the other parameters appear.

You must then select **Job**, then choose **Submit** or click the **Submit** icon.

Using Parameter Sets to Populate Job Parameters

To save time and ensure consistency in the batch jobs you submit, you can save the parameters you enter for a particular batch job and use them again at any time. You can save a parameter set so that it is only available for your use, or make it available for all TMS users of this database.

You need to save and retrieve parameter sets from the same window, Save/Retrieve Parameter Sets. This section discusses each of these operations. You cannot delete parameter sets from TMS.

Saving a Parameter Set

To save a parameter set for a batch job, start by selecting that batch job in the TMS navigator. Then follow the steps below:

1. Choose **Job**, then select **Save Parameter Set**, or click the **Save Parameter Set** button (see Appendix C). The Save/Retrieve Parameter Sets window opens. If you entered any settings in the batch job fields, TMS copies these into the fields in the Save/Retrieve Parameter Sets window as well.

2. Enter values for each field you want to include in this parameter set for this batch job.

3. Enter a name and description for the parameter set.

4. Select the box on the right if you want other TMS users to be able to use this parameter set. Clear the box to be the only person with access to this parameter set. Shared parameter sets are available to all users with access to this batch job at this TMS database instance.

5. Save. TMS saves the parameter set for this batch job.

Retrieving a Parameter Set

Retrieving a parameter set populates the selected batch job with a list of settings for each field in the batch job. You can retrieve a parameter set for a particular batch job once you have selected that batch job in the main TMS window. To retrieve a saved parameter set:

1. Choose **Job**, then select **Retrieve Parameter Set**, or click the **Retrieve Parameter Set** button (see Appendix C). The Save/Retrieve Parameter Sets window opens.

2. Query for the parameter set you want. You can query for the name or description of the parameter set itself, or for one of the specific parameters in it.

3. Click the **Retrieve** button. TMS populates the batch job window with the saved parameters. You can still change any of these parameters before executing the job.

Running a Job

To run the job immediately, select **Job**, then **Submit**, or click the **Submit** icon (see Appendix C for a listing of all icons and their purpose).

To schedule a job to run in the future, see "Scheduling Parameters" on page 2-14.

Checking Job Status

TMS provides information about the status of submitted batch jobs in two places: the main Submission window displays an icon that shows the status of the most recently submitted job, and the Job Status window provides more information about each batch job you have submitted from this TMS database.

Checking the Status of the Last Job Submitted

TMS enables you to determine the status of the current job by examining the **Gear** icon, which appears in the lower right corner of the Job Submission window once you submit the job. This icon changes with a job's status according to the selections in the table below.

Job Running

Displayed while the most recently submitted job is still running.

Job Successful

Displayed when the most recent job is completed successfully.

Job Failed

Displayed when the most recent job fails. Check the error messages in the Job Status window to see why a job failed.

Checking Job Status in the Job Status Window

The Job Status window displays more information about every batch job you have submitted using this data from this instance. Figure 2–5 shows a sample row from this window.

Figure 2–5   One Row in the Job Status Window

Choose **Job**, then **Status** or click the **Status** button (see Appendix C) to open the Job Status window. The window displays the most recently submitted jobs first, but you can query by any of the fields that appear in the window:

- **Label** – The name of the batch job as it appears in the TMS user interface. For example, Activate Preliminary Data.

- **Job Status** – ENTERED (while the job is still running), SUCCESS, or FAILED.

- **Module** – Module of the batch job.

- **Server Job ID** – The Reports service you used to run the job.

■ **Execution ID** – The unique identifier for this job execution in this database.

You can stop a batch job in process (that is, one with Job Status ENTERED) by selecting its row and clicking the **Stop Job** button.

Stopping Job Execution

To stop TMS from executing a job, either before a scheduled execution or during execution:

1. Select **Status** from the **Job** menu or the **Status** icon (see Appendix C) to open the Job Status window.

2. Highlight the job you want to stop.

3. Click **Stop Job**.

Part I

Setting Up and Administering TMS

This section describes what you need to do after installation to set up Oracle Thesaurus Management System (TMS) for use. It also includes information about ongoing administrative tasks.

Part I contains the following chapters:

3

Administration

This section describes the following administrative tasks in Oracle Thesaurus Management System (TMS):

- Security on page 3-1
- Defining Users on page 3-5
- Using Data Access Groups (DAGs) for Security on page 3-9
- Customizing Defaults in TMS Windows Using TMS Settings on page 3-17
- Setting TMS Preferences with Reference Codelist Values on page 3-25
- Managing Distributed Locations on page 3-32
- Autoclassification on page 3-35
- Synchronization on page 3-36
- Writing a Procedure to Change the Dictionary Display on page 3-39
- Controlling the Use of Materialized Views on page 3-39
- Enabling Context Searches for Non-English Dictionaries on page 3-41
- Refreshing the Context Server Index on page 3-42
- Analyzing Tables on page 3-42
- Purging Classified Omissions from the Omissions Table on page 3-42
- Running Scripts to Gather Schema Statistics for the 11g Optimizer on page 3-43
- Adding and Configuring WebLogic Server Data Sources on page 3-43
- TMS HTML Browser Administration on page 3-46
- Troubleshooting on page 3-51

Security

TMS enables data access to be controlled in multiple dimensions. First, database roles control what kind of activity each user is allowed to complete, such as classification, item definitions, item approval. Then, after a user has role assignments, your organization can choose to further specify the domains and/or dictionaries in which this user can perform these activities by assigning certain predefined Data Access Groups (DAGs) to that user. Database roles are created during installation; DAGs are defined in the UI.

This section includes:

Predefined Roles

This section illustrates the predefined roles, with the functions permitted and the menu option associated with each:

opa_admin
rxclin_read
tms_access
tms_allocate_priv
tms_approve_priv
tms_classify_priv
tms_define_priv
tms_dictupg_priv
tms_dsi_priv
tms_integrate_priv
tms_maintain_priv
tms_reclassify_priv
tms_research_priv

A description of each predefined role is included below.

In addition, you can see the grants that are predefined for each of these roles by logging into SQL*Plus as SYSTEM and entering the following:

```
select *from dba_tab_privs where grantee='TMS_ROLE';
```

opa_admin

This role grants access to the Security menu, which contains Define Users, Define Security Columns, and Maintain DAGS windows, as well as the DAG and Setting Inconsistencies report and the Create/Drop EXT_Value Indexes job. The three windows allow you to create new users, define their privileges, and migrate user information from an Oracle Clinical database. The DAG and Setting Inconsistencies report displays inconsistencies between users' data access via DAG and their profile settings. The Create/Drop EXT_Value Indexes job creates or drops indexes on external value columns for performance.

rxclin_read

All users are granted this role by default, which gives them access to items under the **Repository** menu. Thus, all users can browse repository data, VT classification data, and Informative Note data; and all users can run the Date Comparison Report, Dictionary Comparison Report, Dictionary Export Report, and view the HTML Browser.

tms_access

All users are granted this role automatically. This role enables you to view the **TMS** menu and the **Favorites** menu.

tms_allocate_priv

This role gives users access to the windows under the Task Allocation menu. In these windows, users can set up task allocation and allocate tasks—approving VTAs, approving Action assignments, and classifying omissions—to other users.

> **Note:** Users with this privilege can allocate/reallocate/deallocate any task in any domain/dictionary, regardless of their own DAG assignments. However, they can only allocate tasks to users whose DAG assignments and database roles give them access to the data required to perform the task.

> **Note:** Users with TMS_CLASSIFY_PRIV and TMS_APPROVE_PRIV have access to the Task Allocation by Term window, where they can deallocate terms assigned to themselves, and, depending on the value of the reference codelist setting DEALLOCONLY, reallocate those tasks to other users.

tms_approve_priv

This role gives users access to the Approve VTAs and Approve Action Assignments window under **Omission Management**. In this window, users can approve and unapprove VTAs, and create workflow Informative Notes for the VTA approval process.

tms_classify_priv

This role gives users access to all the items under **Omission Management** and also the Task Allocation By Term window under **Task Allocation**. However, users who do not also have tms_approve_priv cannot approve VTAs or Action assignments. Users with tms_classify_priv can classify verbatim terms, apply actions to omissions, and run the VTA Creation Report and VT Domain Differences Report.

tms_define_priv

This role give users access to all items under **Definition** and to the Reports tab in the HTML Browser. Users with tms_define_priv can define dictionaries, domains, external systems, Global Actions, search objects, TMS settings, named relations, and Informative Note Attributes. Users can also run synchronization, refresh the context server index, analyze tables, force rederivation from DSI, and purge classified omissions.

tms_dictupg_priv

This role gives users access to every item under **Dictionary Upgrade** where you can view and manipulate verbatim terms and their relations in the predictionary tables before Activation.

tms_dsi_priv

This role gives users access to the menu item **DSI Maintenance** including the Maintain DSI Files window and several jobs related to data exchange between TMS and disconnected systems.

tms_integrate_priv

This role gives users access to special integration features in the API, but does not unlock any menus or windows in the application.

tms_maintain_priv

This role gives users access to all items under the **Repository Maintenance** and **Translation Reports** menus. Repository Maintenance tasks include: creating and maintaining and deleting terms, relations, and Informative Notes; maintaining dictionary loading error logs; running the Preliminary Repository Report, and running Activation.

There are three reports under the **Translation Reports** menu: Inconsistent Dictionary Codes, Duplicate Dictionary Codes, and Inconsistent Dictionary Relations.

tms_reclassify_priv

This role give users access to all items under **VTA Maintenance**. Users with this role can promote or demote VTAs, maintain actions, reclassify and declassify verbatim terms, perform high-level reclassification, copy domains, maintain domain copying error logs, and run the Nonapproved VTs Report, Classification to a New Domain Report, and the VT Modifications Report.

tms_research_priv

This role enables Oracle Clinical and Oracle AERS users that do not have any other TMS privileges to use the HTML Browser for drilling down to external system data. You may want to grant this role to external system users who require access to source data searches in the HTML Browser, but do not need access to any TMS windows.

Enrolling an Administrator

In a new installation, you must use a script to create one user with the OPA_ADMIN role. That administrator can then use the Define Users window to create user accounts for all the other users in this database; see "Defining Users" on page 3-5 for instructions.

> **Note:**   If you are upgrading to TMS Release 5.1 from Release 4.5.2 you may need to use this script as well.

To create a new account for a TMS administrator:

1. Connect to SQL*Plus as SYSTEM and run the following script:

    opa_home\tms\install\tmsadduser.sql

2. Enter the user's first name.

3. Enter the user's last name.

4. Enter a password for this user. Your input is hidden.

    The script creates a user account for the user you specified and gives the user the OPA_ADMIN database role.

> **Note:**   You cannot use this script to create a nonadministrator user.

Privileges in Local Instances

Functions permitted at local (non-master) instances are limited to the following main menu and submenu options.

Definition

Local Reference Codelists and Maintain Settings windows; and all Jobs: Synchronize Dictionary Data, Refresh Context Server Index, Analyze Tables, Purge Classified Omissions, Create Source Terms Views, Refresh Source Term Mat. Views. Initialize Disconnected System Integrations.

Security

Define Users, Define Security Columns, and Maintain DAGs windows; DAG and Setting Inconsistencies report, Create/Drop EXT_VALUE indexes job.

VTA Maintenance

High-Level Reclassification window; Classification to a New Domain, VT Modifications, and X Areas with Outstanding Changes reports.

Omission Management

Classify VT Omissions and VT History windows; VTA Creation Report, Classification Statistics Report,and VT Domain Differences.

Repository

Browse Repository Data, Browse VT Classification Data, Browse VT History windows; Date Comparison, Dictionary Comparison, Dictionary Export reports.

Translation Reports

Inconsistent Dictionary Codes, Duplicate Dictionary Codes, and Inconsistent Dictionary Relations reports.

Defining Users

This section includes:

- Defining a New User on page 3-5

- Assigning and Revoking Roles on page 3-7

- Changing a User's Password on page 3-8

- Assigning Data Access Groups to a User on page 3-8

- Migrating Account Information from Oracle Clinical on page 3-9
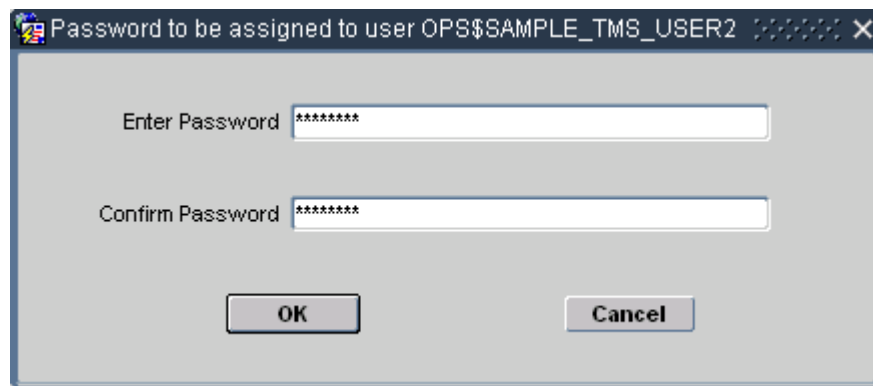
The OPA_ADMIN database role is required to access the Define Users window. See "Enrolling an Administrator" on page 3-4 for information on creating a user account with the OPA_ADMIN database role.

Defining a New User

When you create a new user in the database, you define the user's first and last name, user name in the database, and set a password. Additionally, you can add an email address and set superuser status.

To define a new user, from the Define Users window under the **Security** menu:

1. Define the **Account Name**, which is the user name a user specifies upon login to the application. The account name is often prefixed with OPS$ but this prefix is not required.

2. Enter the user's **First Name** and **Last Name**. The first name and last name appear in the menu bar when this user is connected to TMS.

3. Set the **Super User?** field to either **Yes** or **No**. If this person is a superuser then no Data Access Group information is stored (see "Assigning Data Access Groups to a User" on page 3-8). Superusers have access to all data. Only superusers can run the following jobs and reports:

   - all Disconnected System Integration (DSI) jobs, including importing and exporting data

   - Autoprocess VTAs

   - Predict VTA/VTI Report

   - Site Impacts Report

   - Omission Impacts Report

   - Cross Dictionary Impacts

   - X Areas with Outstanding Changes

   - Classification Statistics Report

   - Date Comparison

   - Dictionary Comparison

   - Dictionary Export

4. Set the **Load User?** field to either **Yes** or **No**. If set to Yes, this person can load dictionaries into the predictionary tables. Most TMS users do not need this ability.

5. (Optional) Enter a **Comment** about this user. Comment information can help administrators identify users, and thus maintain each user's privileges more effectively.

6. (Optional) Enter an **Email Address** for the user. This field is added to support future functionality.

7. Click **Create Schema**. The "Password to be Assigned to User *Account Name*" window opens.

8. Enter and confirm the password for this user, and click **OK**.

If the password entries match, TMS adds this user to the OPA_ACCOUNTS table and TMS_ACCOUNTS table, and grants the TMS_ACCESS role to the account. Proceed to "Assigning and Revoking Roles" on page 3-7 to assign additional roles for this user.

> **Note:** Users who work on replicated studies and who need to see source term-related values using external value filters must have an account on both the master and local databases. For example, if a user filters by an external value such as Study when querying for omissions, and the requested study has source terms entered on a database on which the user does not have an account, an invalid username/password error is displayed.

Assigning and Revoking Roles

TMS uses menu-based security, so roles give a user access to particular TMS menus or windows. The TMS role TMS_MAINTAIN_PRIV, for example, allows a TMS user to use all of the windows and batch jobs under the **Repository Maintenance** menu.

This section describes how to assign roles to, and revoke roles from, a TMS user.

Assigning Roles to a User

To assign roles to an account:

1. Select the user in the Define Users window, then click **Assign Roles**.

   The "Roles to be Assigned to User *Account Name*" window opens, displaying the TMS roles that this account does not currently have.
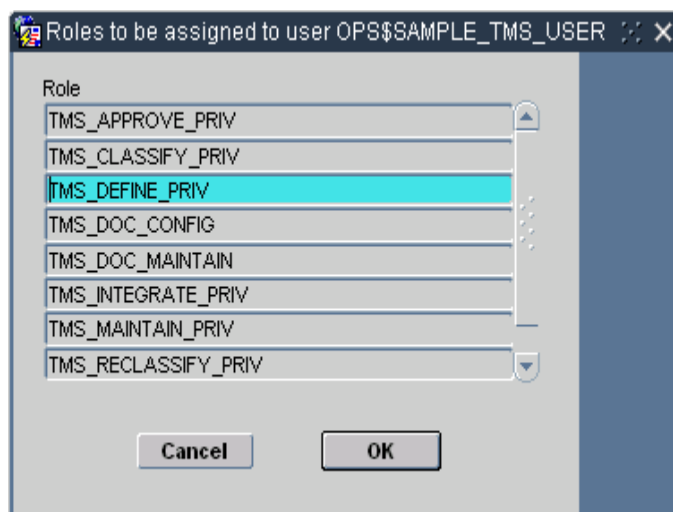
2. Highlight the roles you want to grant to this account. You can Shift-click to select multiple consecutive rows, or Ctrl-click to select multiple non-consecutive rows.

3. Click **OK**. TMS grants the role or roles you selected to this account.

Revoking Roles from a User

To revoke roles for an account:

1. Select the user in the Define Users window, then click **Revoke Roles**.

   The "Roles to be Revoked from User *Account Name*" window opens, displaying the roles currently granted to this account.
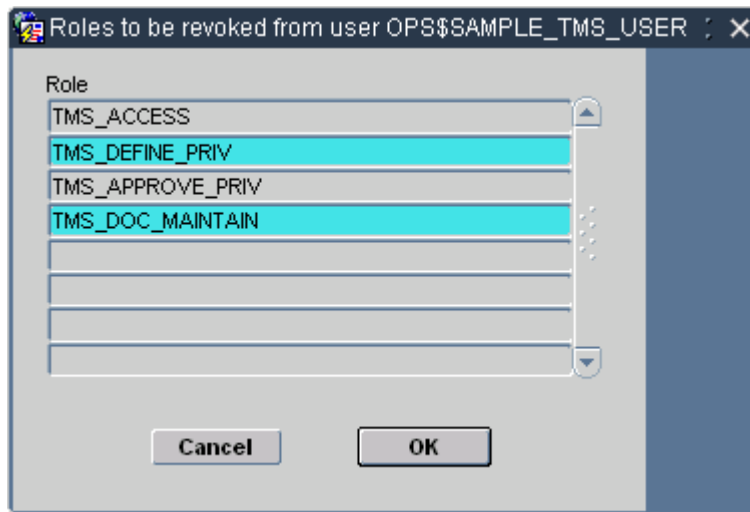
2. Highlight the roles you want to revoke from this account. You can Shift-click to select multiple consecutive rows, or Ctrl-click to select multiple non-consecutive rows.

3. Click **OK**. TMS revokes the role or roles you selected to this account.

Changing a User's Password

You can change any user's password in a database where you have administrator privileges.

To change a user's password:

1. Start a SQL*Plus session as SYSTEM.

2. Issue the following alter user command:

```
alter user username
identified by new_password
```

Assigning Data Access Groups to a User

Data Access Groups (DAGs) work in conjunction with roles to control a user's access to TMS data. You can use them to fine tune the data that a particular user or class of users can operate on. For example, the user with TMS_MAINTAIN_PRIV would still be able to access all of the windows and batch jobs under the **Repository Maintenance** menu, but if you also assign a DAG that only allows access to MedDRA, then he would see only items relevant to MedDRA.

If a person is designated as a superuser in the Define Users window, he or she belongs to all DAGs and has access to all data.

To assign a DAG:

1. Select the user in the Define Users window, then click **Assign DAGs**.

   The "Assign DAGs to user *Account Name*" window opens. If the user has been assigned any DAGs, they are listed.

2. Place the cursor in an empty field under Dag Name. From the LOV, select the DAG to assign.

3. Click **OK**. TMS assigns the DAG or DAGs you selected to this account.

Migrating Account Information from Oracle Clinical

This section applies only to TMS installations that are integrated with Oracle Clinical.

TMS stores general user account information in the OPA_ACCOUNTS table and TMS-specific user account information in the TMS_ACCOUNTS tables. See the *Oracle Thesaurus Management System Technical Reference Manual* for details.

If you have already set up user accounts in Oracle Clinical, you can migrate all account information from ORACLE_ACCOUNTS to OPA_ACCOUNTS, as follows:

1. In TMS, go to **Security**, then select **Define Users**.

2. Click **Get O*Clinical Account Data**.

   The system inserts all records from the ORACLE_ACCOUNTS table into the OPA_ ACCOUNTS table. The default comment text for each record is "Maintained in Oracle Clinical."

3. Set up TMS-specific security for each user:

   ▪ Either set their Superuser flag to **Yes** in the Define Users window or assign a Data Access Group; see "Assigning Data Access Groups to a User" on page 3-8.

   ▪ Assign roles as required; see "Assigning Roles to a User" on page 3-7.

   ▪ Give dictionary loading privileges if required; see "Defining a New User" on page 3-5.

   ▪ Set password; see "Defining a New User" on page 3-5.

You must continue to maintain these accounts in Oracle Clinical.

Using Data Access Groups (DAGs) for Security

This section includes:

▪ Defining Security Columns on page 3-10

-

-

-

-

-

In TMS, database roles dictate which windows in the user interface a user can access. For users assigned to one or more Data Access Groups, their DAG assignments dictate the data they see in TMS windows and in the HTML Browser, and whether or not they can operate on TMS data in specific dictionaries and/or domains.

When you define a DAG:

- You can specify whether users in the group can make changes or only read data.

- You can specify whether users in the group can see data that originated in one or more specific external systems, and from a particular project or study, for example.

- You can specify whether users in the group can see data that originated in one or more specific databases.

- You can specify whether a user can see only his or her own task assignments and/or those of which specific other users.

You define DAGs in the Maintain DAGs window and then assign users to the group either in the User Assignments tab of the Maintain DAGs window or as you create user accounts. A single user can be assigned to multiple DAGs. The user then has access to the sum of data allowed by all his or her DAG assignments. Alternatively, a user can be designated as a Superuser. Superusers can see all data and cannot be assigned to any DAGs.

The OPA_ADMIN database role is required to access the Maintain DAGs window.

Defining Security Columns

Before you define any DAGs, you must specify which TMS table columns—each of which stores a particular type of data—are available to include in DAGS; if a column is available to DAGs, a particular DAG can limit the data its users can see to one or more values stored in that column in TMS. If the column is not available to DAGs, data for all its values is always viewable to users with normal security access.

The columns available for use in DAG security are:

- **DEF_ DICTIONARY_ID**. This column contains the IDs of dictionaries defined in TMS. If you make this column available for use in DAGs, you can allow users assigned to DAGs to see data associated with only one or a few dictionaries.

- **DEF_DOMAIN_ID**. This column contains the IDs of domains defined in TMS. If you make this column available for use in DAGs, you can allow users assigned to DAGs to see data associated with only one or a few domains.

- **DEF_INSTANCE_ID**. This column contains the IDs of your TMS databases. If you make this column available for use in DAGs, you can allow users assigned to DAGs to see data collected on only one or a few databases.

- **DEF_INTEGRATION_KEY**. If you have integrated one or more external source data systems with TMS, this column holds the name of the system(s). If you make this column available for use in DAGs, you can allow users assigned to DAGs to see data associated with only one or a few external systems.

- **EXT_VALUE_1**. This column holds external system data associated with a term and stored in the EXT_VALUE_1 column in the tables TMS_SOURCE_TERMS and TMS_VT_OMISSIONS. For external systems other than Oracle Clinical, you specify the data stored in this column; see "Setting Up External System Columns and Details" on page 5-31. For Oracle Clinical, this column stores the name of the Project associated with a source term. If you make this column available for use in DAGs, you can allow users assigned to DAGs to see data associated with only one or a few Oracle Clinical projects, for example.

- **EXT_VALUE_2**. This column holds external system data associated with a term and stored in the EXT_VALUE_2 column in the tables TMS_SOURCE_TERMS and TMS_VT_OMISSIONS. For external systems other than Oracle Clinical, you specify the data stored in this column; see "Setting Up External System Columns and Details" on page 5-31. For Oracle Clinical, this column stores the name of the Study associated with a source term. If you make this column available for use in DAGs, you can allow users assigned to DAGs to see data associated with only one or a few Oracle Clinical studies, for example.

- **ASSIGNED**. This column holds the user name of the person to whom a term is assigned as a task allocation because the term requires work: approving a VTA, approving an Action assignment, or classifying an omission. If you make this column available for use in DAGs, you can allow users assigned to DAGs to see terms assigned only to themselves (LOGIN_USER) or to specific other users.

To make columns available for use in DAGs, do the following:

1. In the **Security** menu, select **Define Security Columns**. The **Define Security Columns** window opens, displaying each column in a different row.

2. For each column, select a value for the following fields:

    - **Used?**. If set to **Yes**, you can define DAGs to control whether users can see the corresponding field in TMS windows. If set to **No**, DAGs cannot control whether users see the field. The field is visible or enterable to anyone with access to the window.

        > **Note:** You can change the Used? setting from **No** to **Yes** at any time. After a column is included in a DAG definition you can no longer change the setting from **Yes** to **No**.

    - **Create Index?**. You can set this flag to Yes only for the EXT_VALUE columns. If set to Yes, when you run the Create/Drop EXT_VALUE Indexes job (under the Security menu) TMS creates an index on the values in this column to speed performance when these values are used in the context of security. If set to No, TMS drops the index when you run the same job.

3. Save. TMS enters values in the **Creation Time** and **Created By** fields. If you modify the settings for any column, TMS enters values in the **Modification Time** and **Modified By** columns.

Creating a DAG

The process of defining a Data Access Group includes three parts: creating the DAG, specifying the DAG rules, and assigning users to the DAG. There are three tabs in the Maintain DAGs window, one for each task.

To create a DAG:

1. In the **Security** menu, select **Maintain DAGs**. The **DAGs** tab is displayed.

2. On the **DAGs** tab, enter a **Name** and a **Short Name** for the DAG.

3. Select an option for the **Modify Flag?** option. If you want users assigned to this DAG to be able to modify data in the windows to which they have DAG access, select **Yes**. If not, select **No**.

4. Enter a **Status** of **Provisional**. You can set it to Active any time after the definition is completed, including defining the DAG rules. You can add users to a DAG whose status is either Provisional or Active. Only active DAGs are used to enforce data security.

5. Save. Click the **DAG Rules** tab and select settings as described below.

Setting DAG Rules

This section includes:

- About DAG Rules on page 3-12
- Defining DAG Rules on page 3-13
- Rules for External System Data on page 3-14
- Rules for Task Allocation on page 3-15

About DAG Rules

In the DAG Rules tab you specify the data that users assigned to the DAG can view. For one or more of the columns you selected in the Define Security Columns window, you can specify values to determine which data users in this group can see. For example, you can specify a particular dictionary or set of dictionaries. Users assigned to the DAG can then see data related to only those dictionaries (or other dictionaries to which they have access through another DAG).

You can also specify the type of operations that a user can perform in a dictionary or domain by requiring a DAG role for a particular dictionary or domain, and then specifying the role assigned to users in the DAG; see Example 3–1, "DAG Example 1".

DAG Roles   You can use DAG roles for the dictionary and domain columns to determine which operations users with normal security access to various TMS windows can perform there on which dictionary and/or domain data. The DAG roles are:

- **TMS_CLASSIFY** allows users assigned to the DAG to perform classification operations in the Classify VT Omissions window in a particular dictionary and/or domain, if they also have the database role TMS_CLASSIFY_PRIV.

- **TMS_APPROVE** allows users assigned to the DAG to perform approval operations in the Approve VTAs window in a particular dictionary and/or domain, if they also have the database role TMS_APPROVE_PRIV.

- **TMS_MAINTAIN** allows users to perform all operations in all windows in the Repository Maintenance and Translation Reports menus in a particular dictionary and/or domain, if they also have the database role TMS_MAINTAIN_PRIV.

- **TMS_DUPG** allows users to perform all operations in all windows in the Dictionary Upgrade menu in a particular dictionary and/or domain, if they also have the database role TMS_DICTUPG_PRIV.

- **TMS_RECLASSIFY** allows users assigned to the DAG to perform reclassification operations in the Reclassify Verbatim Terms window and in the High-Level

Reclassifications window in a particular dictionary and/or domain, if they also have the database role TMS_RECLASSIFY_PRIV.

Example 3–1   DAG Example 1

Data Access Group 12345 has the following DAG rule defined for the dictionary column (DEF_DICTIONARY_ID), for which a DAG Role is required:

- For the value MedDRA, the DAG roles specified are TMS_CLASSIFY and TMS_APPROVE.

- For the value WHO-Drug, only the DAG role TMS_CLASSIFY is specified.

If you assign a user who has all TMS database roles assigned to this DAG, the user can perform Classify and Approve activities on MedDRA. However, on WHO-Drug, the user can only perform Classify activities, and not, for example, Approve activities.

Example 3–2   DAG Example 2

User A belongs to only one DAG, and that DAG specifies MedDRA as the only dictionary he can access. Therefore, in any windows he can open, he can view only MedDRA data. User A has two database roles: TMS_CLASSIFY_PRIV and TMS_APPROVE.

User A can do the following:

- Open Classify VT Omissions window (because he has the database role TMS_CLASSIFY_PRIV) and open the Approve VTAs window (because he has the database role TMS_APPROVE_PRIV).

- See data from MedDRA in both windows (because MedDRA is specified as a value for the DEF_DICTIONARY_ID column in his DAG).

- Classify verbatim terms to MedDRA terms ((because his DAG has Roles Required for the column DEF_DICITONARY_ID, MedDRA defined as a value for that column, and the DAG role TMS_CLASSIFY is specified for that value).

However, he cannot approve VTAs in the Approve VTAs window because his DAG has Roles Required for DEF_DICTIONARY_ID, MedDRA defined as a value for that column, and TMS_APPROVE is not specified as a DAG role for MedDRA.

Defining DAG Rules

After you have created a DAG, define rules for it:

1. In the **Maintain DAG**s window, **DAGs** tab, select the DAG for which you want to define rules.

2. Click the **DAG Rules** tab. TMS displays the **DAG Rules** tab with the DAG name and short name displayed at the top of the tab.

3. Click in the first empty row in the **Columns** section, **Name** field.

4. Click the ellipsis (…) to invoke the list of values. TMS displays a list of the columns you defined as available to DAGs; see "Defining Security Columns" on page 3-10.

5. For each column you select, specify the following:

   - **Role Req'd**?. TMS allows setting this flag to **Yes** only for the DEF_DICTIONARY_ID (dictionary) and DEF_DOMAIN_ID (domain) columns.

     If set to **No**, users assigned to this DAG can perform any transactions on the data they can see in the windows they can open.

If set to **Yes**, users assigned to this DAG can perform only certain transactions (specified in the **Roles** section for each column value) on the data they can see (specified in the **Values** section) in the windows that they can open.

- **Values**. In the **Values** section, specify every value that applies for the currently selected column in the **Columns** section. For most columns, no value appears in the **External System** field. Click in the **Column Value** field and select a value from the list of values. To specify more than one value, select one in the next row.

  The list of values for the **Assigned** column includes the predefined value **[LOGIN_USER]**. If you select this predefined value, the user currently logged in to TMS can see the tasks assigned to himself or herself.

  For information on the external system-related columns, see "Rules for External System Data" on page 3-14.

- **Roles**. If you set **Role Req'd?** to Yes, you must specify at least one role for each column value. For example, if you specified Role Required for the DEF_DICTIONARY_ID column, and specified MedDRA as one of the values for the column, the Roles you specify for MedDRA determine what operations users in this user group will be able to perform on MedDRA terms. If you specify TMS_CLASSIFY, users with access to the Classify VT Omissions window will be able to classify verbatim terms to MedDRA terms. If you do not specify TMS_CLASSIFY, users with access to the window will be able to see MedDRA terms but will not be able to perform any classification operations on them.

6. Save. Repeat for each column you want to include in the DAG.

Rules for External System Data

To limit users' access to data from a particular external system, select the DEF_INTEGRATION_KEY as a column, then select the particular external system to which you want to allow access in the Column Value field.

To allow access to a particular level of information within an external system, use the EXT_VALUE_1 and EXT_VALUE_2 columns.

For Oracle Clinical, the EXT_VALUE_1 is predefined to hold project names and the EXT_VALUE_2 column is predefined to hold study names. For other external systems you define what data to store in these columns; see "Setting Up External System Columns and Details" on page 5-31.

For example, if you specify Oracle Clinical as an external system value and then specify Project A as the only EXT_VALUE_1 value and Study A as the only EXT_VALUE_2 value, users assigned to this DAG can only view source terms that originated in Oracle Clinical Project A, Study A. They cannot view source terms from Project A, Study B, or source terms that originated in another external source data system.

To specify external system data to be accessible to users in this DAG, do the following:

1. In the **Columns** section, **Name** field, select **EXT_VALUE_1** or **EXT_VALUE_2** from the list of values.

2. With **EXT_VALUE** column selected in the **Columns** section, click the **External System** field in the **Values** section.

3. From the list of values, select the external system whose data you want users to be able to see.

4. Click the **Column Value** field and invoke the list of values. TMS displays the values stored in the EXT_VALUE_1 (or EXT_VALUE_2) column for the external system you selected. For example, if you selected Oracle Clinical as the external system, TMS displays Oracle Clinical project names.

5. From the list of values, select a column value.

Users in this DAG will be able to see data associated with the external systems and column values you specify, and they will not be able to see data associated with external systems and column values you did not specify. Repeat this procedure to allow access to additional values for the same external system, and to allow access to data from additional external systems.

> **Note:** If you use the EXT_VALUE_1 or _2 columns to specify access to data from a particular external system, you must also specify the external system itself as a value in the DEF_INTEGRATION_KEY column.

> **Note:** For best performance, each time you add or delete a value from one of the EXT_VALUE columns, run the Create/Drop EXT_VALUE Indexes job.

Rules for Task Allocation

If your company is using the Task Allocation feature and you want to restrict users' ability to allocate tasks in certain dictionaries or domains, or restrict user's ability to see other users' assigned tasks, you can use DAGs to enforce those restrictions.

TMS allows users with task allocation privileges to allocate tasks to any user who is available for tasks associated with a particular dictionary. However, if the allocator is assigned to one or more DAGs, he or she can allocate tasks to other users only for the dictionaries he or she (the allocator) has access to through a DAG. The task-receiving user must have access to the same dictionary through a DAG as well (or as a superuser).

Example 3–3   DAG Definitions and Task Allocation

TMS user Ted has TMS_ALLOCATE_PRIV and is assigned to DAG 12345, which includes two dictionaries: MedDRA and WHO-Drug.

TMS user Alice is assigned to DAG 67890, which includes two dictionaries: MedDRA and CoStart.

Ted can assign Alice tasks related to MedDRA only; not WHO-Drug or CoStart.

> **Note:** The values of the Assigned column restrict the data that the user sees in Classify VT Omissions, Approve VTAs, Approve Action Assignments windows. They do not affect what the user sees in the Task Allocation windows. As long as the user has task allocation privilege, he can see all tasks (assigned to anyone) in the Task Allocation windows.

Assigning Users to DAGs

You can assign users to a DAG either in the User Assignments tab of the Maintain DAGs window or in the Define Users window. In the Maintain DAGs window, you can add multiple users to a single DAG at the same time. In the Define Users window you can assign a single user to multiple DAGs at the same time. See "Defining Users" on page 3-5 for more information.

To add users to a DAG in the Maintain DAGs window, do the following:

1. In the Security menu, select Maintain DAGs. The Maintain DAGs window opens.

2. Click the User Assignment tab. The User Assignment tab opens.

3. Click in the Account Name field in the first empty row. TMS displays … in the right corner of the field.

4. Click the … to retrieve the list of values including all user accounts available for assignment to a DAG (superusers are not included).

5. Select the user account you want to assign to the DAG. You can query in the Find field to narrow your search.

6. Repeat the procedure to add another user account in the next row as many times as necessary. If necessary, select Insert from the Record field to additional rows, one at a time.

7. Save. TMS assigns the users you specified to the DAG, which now controls their access to TMS data.

DAG and Settings Inconsistencies Report

This section includes:

- About the DAG Settings and Inconsistencies Report on page 3-16

- Running the DAG and Setting Inconsistencies Report on page 3-17

About the DAG Settings and Inconsistencies Report

When you create a user account in TMS, you specify profile settings for the user. In addition, if the user is not a superuser you also assign the user to at least one Data Access Group (DAG).

DAGs determine which data the user can view, and profile settings determine which data a user views by default. TMS does not check or enforce that these settings match; therefore it is possible to set a user's profile to default settings for data the user cannot actually see. Use this report to determine if any users are currently in this situation.

It is also possible to create a user account and not assign the user to any DAGs. Users designated as superusers can see all data and cannot belong to a DAG. However, if you do not designate a user as a superuser and also do not assign him or her to a DAG, the user will not be able to do anything in TMS.

The DAG and Setting Inconsistencies Report shows the following:

- A list of all non-superusers who are not associated with an active DAG.

- A list of all non-superusers who do not have DAG access to the following default settings:

    - dictionary

    - domain

- external system

- external value(s)

Running the DAG and Setting Inconsistencies Report

To generate a DAG and Setting Inconsistencies Report, do the following:

1. From the **Security** menu, select the **DAG and Setting Inconsistencies Report**.

2. Enter a value for the General parameter **Output**. Select the format in which you wish to generate the report - HTML, PDF, RTF, XLS or XML. This is a mandatory field.

3. Enter a value for the Job Specific parameter **Template**. Select the template you want to use for this report. If your company has created a custom template, it appears in the list of values. The Oracle Template is the default template.

4. Submit the job. Select **Job**, then **Submit**, or click the Submit icon to generate the report in the selected output format.

   A new browser window opens with the generated report.

Creating and Dropping External System Value Indexes

Use this job to create or drop indexes on the column EXT_VALUE_1 and/or EXT_VALUE_2 in the TMS tables TMS_VT_OMISSIONS and TMS_SOURCE_TERMS. You specify which column or columns to create an index for in the Define Security Columns window (see "Defining Security Columns" on page 3-10) but you must run this job to actually create the index.

For best performance, run this job whenever you add or remove values from the EXT_VALUE_1 or EXT_VALUE_2 columns in a DAG definition.

To run the job, do the following:

1. Under Job-Specific parameters, click in the Actions field and select a value from the drop-down list: Create Indexes or Drop Indexes.

2. Under Schedule, select a Report Server.

3. To run the job at a later time, set the other scheduling parameters; see "Setting Up and Running Reports and Other Batch Jobs" on page 2-13.

4. Submit the job.

Customizing Defaults in TMS Windows Using TMS Settings

TMS Settings are default values that populate many fields in TMS, such as **Domain**, **Dictionary**, and the fields that appear in the Filter window. You can establish settings that are company-wide, or specify profiles from which sets of TMS users will inherit settings values.

This section includes:

■ Assigning Profiles to Users on page 3-24

Profiles

A **profile** is a catalog of TMS settings values. You can define profiles that specify one or more of the TMS settings, then assign any profiles to users to control their default behavior in TMS. For example, you can create a profile that defines default values for the **Action Owner** and **External System**. If you assign this profile to a user (see "User Profiles" on page 3-18), these settings could control that user's default settings for these two fields in TMS.

Two default profiles provide baseline settings for TMS. The Oracle profile includes a standard set of keys and settings that provide default settings throughout the system. The Company Settings profile establishes default values that are specific to your company. Company settings override Oracle settings, and provide a baseline set of values for all users in your company.

The additional, custom profiles you define enable you to set up more specific default values for groups of users. Custom profiles override both the Company Settings and Oracle profiles. You can set up as many custom profiles in TMS as you want, and assign multiple profiles to each user.

User Profiles

The **user profile** is the set of profiles that have been assigned to a user. For each profile that you assign to a user, you must define a Search Order Number. The Search Order Number shows TMS which profiles to consider first when a user profile contains multiple profile definitions.

TMS sets a user's default values by checking through each profile in the user profile:

1. TMS finds the first user profile (the user profile with the lowest Search Order Number) and sets the value of each TMS field that the setting specifies.

2. TMS then checks each subsequent user profile row in ascending order by Search Order Number. For each of these profiles, TMS sets defaults that have not been set by an earlier profile. If the first user profile created this user's default domain setting, the other profiles cannot override it.

3. The system then checks the Company Settings profile for other TMS settings that have not been specified for this user by any of the user's custom profiles.

4. Finally, TMS checks the Oracle profile for other unspecified TMS settings.

Full List of TMS Settings

Each setting value controls the default behavior of one field in TMS. To determine your default domain, for example, TMS checks through your hierarchy of profiles for a value in the TMS_CURRENT_DOMAIN setting. The system then chooses that domain for you when you log in to TMS.

Many TMS settings control the default value for fields that also appear in the TMS filter windows. The following windows have filters: Approve VTAs, Reclassify Verbatim Terms, Classify VT Omissions, and Maintain DSI Files.

The Approve VTAs and Reclassify Verbatim Terms windows both use the Omissions Filter, with the same settings. The TMS_DSI settings control values in the Maintain DSI Files window. Enter values for TMS_DSI settings only if you are using the Disconnected System Integration feature, and only if you want a default value set in your filter for that window (see "Using TMS with Disconnected Systems" on page 5-3).

For information on how to enter values for parent and child settings, see "Defining a Setting When No Parent-Child Setting Relationship Exists" on page 3-22 and "Defining a Setting in Parent-Child Relationships" on page 3-23.

> **Note:** You may need to log out and log in again for new settings to take effect.

Settings are grouped according to the part of the application they control:

- OPA Settings
- TMS Forms Settings
- TMS HTML Settings

OPA Settings

OPA settings control areas of the application that are common to all Oracle Health Sciences (formerly Oracle Pharmaceutical Applications) products.

OPA_JAVA_DATE_FORMAT: The default format for data values in Java activities in TMS. For example, `dd-MMM-yyyy`

OPA_JAVA_TIME_FORMAT: The default format for time values in Java activities in TMS. For example, `kk:mm:ss`

OPA_SQL_DATE_FORMAT: The default format for date values in SQL queries. For example, `DD-MON-YYYY`

OPA_SQL_TIME_FORMAT: The default format for time values in SQL queries. For example, `HH24:MI:SS`

OPA_UIX_MAX_ROWS: For UIX-based applications such as the HTML Browser, this setting determines the maximum number of records that queries will retrieve.

OPA_UI_COUNTRY: Two-letter short name of the default country. An LOV is available.

OPA_UI_LANG: Two-letter short name of the default language. An LOV is available.

TMS Forms Settings

TMS Forms settings control the default settings for the Oracle Forms-based TMS user interface.

The parent-level settings are:

TMS_CURRENT_ACTION_OWNER: This setting controls the default value for the **Action Owner** field.

TMS_CURRENT_ACTION_SYS: Controls the default value for the **External System** field.

TMS_CURRENT_ACTION_WKFLW: Controls the default value for the **Action Workflow** field.

TMS_CURRENT_ASSIGNED: Controls the default value for the **Assigned** field in the **General** tab of the Omission Filter.

TMS_CURRENT_DOMAIN: The domain that is selected for your profile by default. This setting is a parent key for TMS_CURRENT_DICTIONARY.

TMS_CURRENT_EXT_SYSTEM: The external system that is selected by default.

TMS_CURRENT_SYNC_INSTANCE: Controls the **TMS DB** field in the **General** tab of the Omission Filter and other Filter windows.

TMS_DSI_CURRENT_EXT_SYS…: From the list of values, select the name of the external source data system you most commonly use in Disconnected System Integration. This setting populates the **External System** field in the Maintain DSI Files Filter, and has a child and grandchild setting; see TMS_DSI_CURRENT_INSTANCE and TMS_DSI_CURRENT_X_AREA.

TMS_DSI_ERROR_PROCESS_STATUS: From the list of values, select N or Y (No or Yes). If N, the **Error** box for **Process Status** is deselected by default, and the system does not query for jobs that ended with a status of Error. If Y, the **Error** box for **Process Status** is selected by default, and the system queries for jobs that ended with a status of Error.

TMS_DSI_FATAL_PROCESS_STATUS: From the list of values, select N or Y (No or Yes). If N, the **Fatal** box for **Process Status** is deselected by default, and the system does not query for jobs that ended with a status of Fatal Error. If Y, the **Fatal** box for **Process Status** is selected by default, and the system queries for jobs that ended with a status of Fatal Error.

TMS_DSI_LAST_EXPORT_END_TS: If you will often search for files exported during a particular time period, enter the last day of that time period using the date format in your OPA_SQL_DATE_FORMAT setting. This setting populates the (**Last Export Between…**) **And…** field in the Maintain DSI Files Filter.

TMS_DSI_LAST_EXPORT_START_TS: If you will often search for files exported during a particular time period, enter the first day of that time period using the date format in your OPA_SQL_DATE_FORMAT setting. This setting populates the **Last Export Between…** field in the Maintain DSI Files Filter.

TMS_DSI_MANUAL_STATUS: From the list of values, select the import/export job manual status you most frequently want to query for, if any. The choices are: **New**, **Pending**, or **Fixed**. This setting populates the **Manual Status** field in the Maintain DSI Files Filter.

TMS_DSI_PROCESS_END_TS: If you will often search for files that ran during a particular time period, enter the last day of that time period using the date format in your OPA_SQL_DATE_FORMAT setting. This setting populates the (**Process Between…**) **And…** field in the Maintain DSI Files filter.

TMS_DSI_PROCESS_START_TS: If you will often search for files that ran during a particular time period, enter the first day of that time period using the date format in your OPA_SQL_DATE_FORMAT setting. This setting populates the **Process Between…** field in the Maintain DSI Files filter.

TMS_DSI_PROCESS_TYPE: From the list of values, select **Export** or **Import**. This setting populates the **Process Type** field in the Maintain DSI Files Filter.

TMS_DSI_SUCCESS_PROCESS_STATUS: From the list of values, select N or Y (No or Yes). If N, the **Success** box for **Process Status** is deselected by default, and the system does not query for jobs that ended with a status of Success. If Y, the **Success** box for **Process Status** is selected by default, and the system queries for jobs that ended with a status of Success.

TMS_DSI_WARNING_PROCESS_STATUS: From the list of values, select N or Y (No or Yes). If N, the **Warning** box for **Process Status** is deselected by default, and the system does not query for jobs that ended with a status of Warning. If Y, the **Warning** box for **Process Status** in selected by default, and the system queries for jobs that ended with a status of Warning.

TMS_DSI_X_AREA_STATUS: From the list of values, select **Active**, **Complete**, or **Inactive**. This setting populates the **X Area Status** field in the Maintain DSI Files Filter.

TMS_PROCESS_DICTIONARY If some dictionary/domain combinations require approval for VTAs or actions, you can use this profile setting allow one or more particular users to apply Answerable Actions directly in this dictionary (for any domain), with no approval or Internal Action required, and/or to create Approved VTAs directly, with no approval required. See TMS_ACTION_APPROVAL_ REQUIRED on page 3-21 and TMS_VTA_APPROVAL_REQUIRED on page 3-21.

Child-level Settings

Each of the following settings is dependent on one of the parent-level settings. See "Defining a Setting in Parent-Child Relationships" on page 3-23.

TMS_ACTION_APPROVAL_REQUIRED: If set to N, users with this setting in their profile can assign Answerable actions directly, without assigning an Internal Action for approval, regardless of the Action approval setting for the dictionary/domain combination. This setting is a child key for TMS_PROCESS_DICTIONARY.

> **Note:** A setting of Y is not supported. (You can save the setting Y, but it will not have any effect.)

TMS_CURRENT_DICTIONARY: Within the selected domain, the dictionary that is selected for your profile by default. This setting is a child key for TMS_CURRENT_ DOMAIN.

TMS_CURRENT_EXT_VALUE_1 … 8: Within the selected external system, these eight settings determine the default values for each of the eight external system fields in the **External System** tab of the Omission Filter.

TMS_CURRENT_INSTANCE: Within the selected external system, the default source database in the **External System** tab of the Omission Filter.

TMS_CURRENT_OMIS_STATUS: Within the selected external system, the default omission status. This setting populates the **Omission Status** field in the **Actions** tab. No LOV is available for this field, so make sure the omission status you enter as the default matches the status required in the Filter window.

TMS_DSI_CURRENT_INSTANCE: Within the selected DSI external system, the default Database ID; enter the most frequently used remote database or leave blank. The list of values displays the name as well as the ID of each registered remote database.

TMS_DSI_CURRENT_X_AREA: This is a child key of the TMS_DSI_CURRENT_ INSTANCE child key. For the database chosen as the default current instance, enter the ID of the most frequently used X Area on the remote database, if any. The list of values displays the name as well as the ID of each X Area on the selected remote database.

TMS_VTA_APPROVAL_REQUIRED: If set to N, users with this setting in their profile can create Approved VTAs directly in this dictionary (for any domain), regardless of the VTA approval setting for the dictionary/domain combination. This setting is a child key for TMS_PROCESS_DICTIONARY.

> **Note:** A setting of Y is not supported. (You can save the setting Y, but it will not have any effect.)

TMS HTML Settings

TMS HTML settings control the default settings for HTML-based parts of the TMS application, such as the HTML Browser.

The parent-level settings are:

TMS_HTML_DQ_EXT_SYS: This setting controls the default value for the **External System** field in the HTML Browser. This setting is a parent key for TMS_HTML_DQ_EXT_SYS_QRY.

TMS_HTML_LANGUAGE: This setting controls the default value for the **Action Owner** field. This setting is a parent key for TMS_HTML_DICTIONARY.

The child level settings are:

TMS_HTML_DQ_EXT_SYS_QRY: Within the selected HTML external system, this setting controls which external system query is selected by default. These queries are also called groups, and appear in the **Group** field in the HTML Browser. This setting is a child key for TMS_HTML_DQ_EXT_SYS.

TMS_HTML_DICTIONARY: Within the selected HTML language, the dictionary that is selected for your profile by default. This setting is a child key for TMS_HTML_LANGUAGE.

Defining the Settings for a Particular Profile

The **Settings** tab enables you to define the default settings values for a profile. All settings are grouped hierarchically by their Top External Area (Top X Area), which describes what part of the application they control. The Top External Areas are OPA, TMS Forms, and TMS HTML.

Some settings also have parent-child relationships, where the child setting depends on the selected value of the parent. For example, the TMS Forms setting TMS_CURRENT_DOMAIN is a parent key for the child key TMS_CURRENT_DICTIONARY. While you can use TMS_CURRENT_DOMAIN to define which domain is selected by default for your profile, you can specify for non-default domains which dictionary is selected by default when you switch to that domain.

This section includes:

■ Defining a Setting When No Parent-Child Setting Relationship Exists on page 3-22

■ Defining a Setting in Parent-Child Relationships on page 3-23

Defining a Setting When No Parent-Child Setting Relationship Exists

This section describes how to define which settings to use for a particular profile. A profile must exist in the database before you can define its settings; See "Defining and Maintaining Profiles" on page 3-24.

This section describes how to define values for settings that are not part of parent-child relationships. See "Defining a Setting in Parent-Child Relationships" on page 3-23 for more information about that process.

To define a setting for a profile:

1. From the **Definition** menu, select **Maintain Settings**. The Maintain Settings window opens.

2. From the **Profile** list, choose the profile in which you want to define settings.

3. Choose the Top External System Area from the **Top X Area** field. Top External System Areas group together the settings by application area: you can choose **OPA**, **TMS Forms**, and **TMS HTML**.

   Your choice of External System Area populates the values in the **Settings Key** field.

4. Choose the setting that you want to define from the list.

The detail fields under the **Extend Type** field provide more information about the value or default value, but only in cases where the value or default value is not descriptive by itself. The above example for **Current Domain** uses the domain ID as its value, so the detail fields display the **Domain Short Name**.

5. Enter the value for the setting, or choose a value if an LOV exists. The Define Settings window displays default settings values for each setting key in the **Default Value** field.

   Extend Type currently has no effect in TMS.

6. Click **Set Default**. TMS saves this default value for this setting.

Defining a Setting in Parent-Child Relationships

In cases where a setting has one or more child settings that are dependent on it, you can use the child setting to establish a default value for every parent setting choice. For example, the Domain setting has the Dictionary setting as a child. You can establish a default Dictionary setting for each domain value in the database. When you choose a domain in the application, TMS will default the dictionary to the one specified for your setting for the selected domain.

To define a child setting:

1. Select the **Profile** and **External Area** that contain the parent and child key you want to define.

2. From the **Setting Key** field, choose the child key's parent.

3. Choose the parent key's value. You can specify the default child key setting for any parent key value, whether or not the value is the parent key's default.

4. From the **Child Key** field, choose the child key for which you want to create defaults.

   TMS populates the **Setting Key** field with your selected child key, and displays the information about its parent key and parent key value.

5.   Enter the value for the child key for this parent setting. A list of values may be available when you click in the field.

6.   Click **Set Default**. TMS will use this default setting for the child key setting whenever you select this parent key value.

You can return to the parent settings level by clicking the **Top Level** button.

Defining and Maintaining Profiles

Profiles are the basis for creating default settings. By default, TMS includes the default Oracle Profile and a default Company Profile. You can use the **Profiles** tab to add, modify, and delete profiles.

Settings	Profiles	User Profiles	
Id	Display Label		Description
40400	QA		Quality Assurance Group
62200	Documentation		Documentation Group

To add a new profile, insert a row and enter a display name and a description. The display name appears in the **Settings** and **User Profiles** tabs.

To delete a custom profile (that is, one other than the Oracle or Company Profiles), select its row and delete the record. TMS prompts you to confirm the deletion; click **Yes** to complete the profile deletion. You cannot delete the Oracle or Company Settings profiles.

You can also change the profile's display name or description at any time.

Assigning Profiles to Users

The **User Profiles** tab displays user/profile combinations, and enables you to assign profiles to (or remove profiles from) users, and to modify a user profile. Because you can assign multiple profiles to a user, the User Profile determines the relative precedence of these profiles.

This tab is also convenient for assigning a profile to all members of a group.

Adding a Profile to a User's Set of Profiles

To add a profile to a user, and set its Search Order Number:

1.   Insert a record.

2.   Choose the profile to which you want to add the user from the LOV.

3.   Enter the user's OPS$ user name. It must match exactly, and no LOV is available.

4.   Enter a Search Order Number. TMS checks lower numbers earlier when determining which profile's values to use. You can always adjust a user's overall search order by following the instructions in "Configuring a User's Profile Search Order" on page 3-24.

5.   Save.

Configuring a User's Profile Search Order

Search Order determines the hierarchy of profiles for a particular user. TMS first consults the profile with the lowest Search Order value and implements the default

values it specifies. The system then proceeds through the remaining profiles in ascending order by Search Order value, implementing any defaults that have not yet been set for this session.

To configure a user's profile Search Order:

1. Query for the user. The window displays a row for each group to which that user belongs, with a Search Order specified for each.

2. Rank the profiles by updating numbers for each row. The lower the Search Order Number, the earlier TMS will consult this profile for this user. Each Search Order Number must be unique for a user.

3. Save.

Setting TMS Preferences with Reference Codelist Values

This section includes:

-   Reference Codelists in Integrated and Nonintegrated Environments on page 3-25

-   Editing Reference Codelists on page 3-26

-   TMS_CONFIGURATION on page 3-26

-   Other Installation-wide Codelists on page 3-30

-   Local Codelists on page 3-31

Reference codelists provide a body of information for TMS forms and processes, which use the codelist values to create default settings, populate lists of values, and control some aspects of TMS data processing. The values and settings are defined in TMS databases, and can usually be overridden for a more specific situation somewhere else in the system. You cannot delete reference codelists or their values, but you can set one or more values or a reference codelist itself to Inactive so it will have no effect in TMS.

For example, many TMS forms enable you to perform simple text queries as well as queries that use the Oracle Context Server. In these forms, you can choose which of these query options you want by selecting **SIMPLE** or **CONTEXT** in the **Query Type** field. These two values are provided in a TMS installation reference codelist, TMS_QUERY_TYPE.

There are two types of reference codelists in TMS.

-   Installation reference codelist values are set on the master instance and replicated to all the sites in a TMS installation.

-   Local reference codelists are specific to each site within the company.

Reference Codelists in Integrated and Nonintegrated Environments

To integrate TMS with Oracle Clinical, you must set three reference codelists in Oracle Clinical; see "Setting Integration Reference Codelist Values" on page 4-21.

When TMS is integrated with Oracle Clinical, TMS' reference codelist replication is automatically disabled and Oracle Clinical takes over that function.

If you are using TMS separately from Oracle Clinical, TMS must use a batch job to replicate reference codelist information in a global environment; from a local database, use the Synchronize Reference Codelists job under the Definition menu for this purpose.

If you are using the Disconnected System Integration feature of TMS, see "Setting DSI Preferences with Reference Codelist Settings" on page 5-11.

Editing Reference Codelists

To edit a reference codelist:

1. Navigate to the **Definition** menu and click either **Installation Reference Codelists** or **Local Reference Codelists**. The window opens in Query mode.

2. In the **Name** field, query for the codelist that you want to edit.

3. Define the codes in the lower part of the window using the Description field and documentation for each codelist.

   a. The short value for the code is read-only.

   b. The long value for the code. You can edit this value.

   c. **Active?** If selected, this codelist value will appear in lists that use this codelist for an LOV. If not selected, this value will not appear.

   d. **Default?** This box has no effect in TMS.

4. Save.

TMS_CONFIGURATION

TMS_CONFIGURATION provides a list of values and behaviors that you can set for the application. The Short Value column provides a short name for the value or behavior. The Long Value column is enterable, and you can accept the value provided or modify it. If you select **Active?**, the value becomes valid in the system.

There is no default value for TMS_CONFIGURATION, so selecting **Default?** does not change the behavior of this reference codelist.

This section provides additional descriptions of TMS_CONFIGURATION values and the features they drive in TMS. See also "Changing TMS_CONFIGURATION Settings After Data Has Been Processed" on page 3-30.

> **Note:** *The settings followed by asterisks (*) above help determine the way TMS processes source terms and omissions. See "Changing TMS_CONFIGURATION Settings After Data Has Been Processed" on page 3-30.

The settings are:

CREATEGLOBALVTA*
GLOVTAAPPRREQD*
DOMVTAAPPRREQD*
NONAPPRVTAUSED*
ALLOWGLOVTAS*
MAINTAINEXT

SUPPRESNAMDRELS
WRKFLWINFALLVSN
VTAAPPRFLAG
DTAPPRFLAG
CLCLEARCOMMENT
CLCLEARACTION
AUTOGLOVTAS*
SYNCREFRESHMV
ADVDICTFEATURES
EMBEDDEDTMS
OCDISCINTCOMM
MTRECCLASSHOUR
DOMACTAPPRREQD
DEALLOCONLY
NONAPPRDICTCODE

CREATEGLOBALVTA

This setting controls the default behavior of the Global setting in the Maintain VT Omissions window, which is in turn used to classify a VTA as being available globally or specific to a particular domain.

If Y, in the Maintain VT Omissions window TMS will create VTAs as Global by default. If N, in this window TMS will create VTAs as Domain-specific by default.

GLOVTAAPPRREQD

This setting dictates the default approval status of manually created Global VTAs. If Y, TMS creates global VTAs as Nonapproved by default. If N, TMS creates Global VTAs as Approved by default.

> **Note:** When TMS creates a VTA automatically by synchronization, the system does not refer to this setting.

DOMVTAAPPRREQD

This setting determines the default value of the Approval Required? flag in the Define Domain Dictionaries window. If Y, the default setting is checked; if N, the default setting is unchecked. Users can change the value for any particular domain. See "Creating Domains and Assigning Dictionaries to Domains" on page 6-37 for more information.

NONAPPRVTAUSED

If N, a term that matches a Nonapproved VTA will generate an omission in TMS with Search = 'Nonapproved VTA' and a discrepancy in Oracle Clinical.

If Y, such a term will classify in TMS.

ALLOWGLOVTAS

If Y, TMS allows users to create global VTAs. If N, users cannot create global VTAs manually.

MAINTAINEXT

If Y, users can edit external terms from TMS windows under the **Repository Maintenance** menu. If N, external terms are read-only in these windows.

SUPPRESNAMDRELS

If Y, TMS hides named relations in the Maintain Repository Data and Browse Repository Data windows. If N, these windows display named relations. You might want to suppress named relations if you only use strong dictionaries in your TMS environment.

WRKFLWINFALLVSN

This setting controls the default behavior of the **All Versions?** box in the **Maintain Informative Notes** window accessed from the Repository Authoring window under the Repository Maintenance menu. If Y, this box defaults to selected; if N, it defaults to deselected.

WRKFLWINFALLVSN controls the default behavior only. You can still override the default and select **All Versions?** to make a workflow Informative Note apply to all versions, or clear it to make the note version-specific.

VTAAPPRFLAG

This setting controls the default status of the **Appr?** (or **Approved?**) box for verbatim terms you create in either Maintain Repository Data or Repository Authoring. If Y, this box will be selected by default for each verbatim term you create while the window is open. If N, this box will be deselected while the window is open.

VTAAPPRFLAG controls only the default status of the **Appr?** box. You can still override this default.

DTAPPRFLAG

This setting controls the default status of the **Appr?** (or **Approved?**) box for dictionary terms you create in either Maintain Repository Data or Repository Authoring. If Y, this box will be selected by default for each dictionary term you create while the window is open. If N, this box will be deselected while the window is open.

DRAPPRFLAG controls only the default status of the **Appr?** box. You can still override the default.

CLCLEARCOMMENT

(Classify VT Omissions Window Clear Comment) If Y, TMS clears the **Comment** field in the Classify VT Omissions window when you classify a verbatim term. If N, TMS retains the original comment, which you can edit before you commit your classification change to the database.

CLCLEARACTION

(Classify VT Omissions Window Clear Action) If Y, TMS clears any Action associated with a verbatim term after you apply an Action to an omission. If N, TMS does not clear the Action.

AUTOGLOVTAS

(Autocoding Process Creates Global VTAs) If AUTOGLOVTAS is Y, and ALLOWGLOVTAS is Y, the autocoding process will create Global VTAs. If AUTOGLOVTAS is N (but ALLOWGLOVTAS is still Y), autocoding creates Domain VTAs, but you can manually create Global VTAs.

SYNCREFRESHMV

If Y, TMS will refresh materialized views during Synchronization.

The Filter windows in TMS use materialized views to keep external system data up-to-date. Refreshing the materialized views can be time-consuming, so if you do not rely on external system data in Filter windows, you might want to choose N for this setting. See "Controlling the Use of Materialized Views" on page 3-39.

ADVDICTFEATURES

If Y, TMS will display advanced dictionary features, such as the **Object Classification?** box in the Define Dictionaries window. If you do not use the advanced dictionary features, you might want to choose N for this setting.

EMBEDDEDTMS

This setting is for internal use only.

OCDISCINTCOMM

This codelist setting enables you to control where the system saves the comments you create for terms. If Y, the system saves comments in the discrepancy_entries.internal_ comment column. If N, the system saves comments in the discrepancy_ entries.comment column. The default value is Y.

The difference is significant to Oracle Clinical Discrepancy Management, because internal comments appear in some reports and Data Clarification Forms (DCFs), while regular comments do not. If you prefer to hide these non-internal comments from DCFs and other reports, set OCDISCINTCOMM to N.

MTRECCLASSHOUR

Enter the number of hours, if any, you want a user to have to fix a classification mistake he or she has made. When the long value is greater than zero, users can change classifications they have made in the Reclassify Verbatim Terms window, within the number of hours in the **Long Value** field, even if they do not have the normal reclassification privilege. The system displays only classifications created by the user logged into the system.

DOMACTAPPRREQD

This setting determines the default value of the Action Appr Reqd? flag in the Define Domain Dictionaries window. If Y, the default setting is checked; if N, the default setting is unchecked. Users can change the value for any particular domain. See "Creating Domains and Assigning Dictionaries to Domains" on page 6-37 for more information.

DEALLOCONLY

If the long value for short value DEALLOC is set to Y, any user can deallocate tasks from him or herself but cannot reallocate tasks to other users. If set to N, any user can reallocate tasks from him or herself to other users.

> **Note:** Users with the TMS_ALLOCATE_PRIV role can reallocate tasks regardless of this reference codelist setting.

NONAPPRDICTCODE

This setting determines the default value of the **Non Appr DT** field in the Define Domain Dictionaries window. If the long value is NONE, the default value for the Non Appr DT field is NONE. If the long value is VTA, the default value of the field is VTA.

See "Creating Domains and Assigning Dictionaries to Domains" on page 6-37 for more information.

Changing TMS_CONFIGURATION Settings After Data Has Been Processed

Some of the settings for the TMS_CONFIGURATION reference codelist (CREATEGLOBALVTA, GLOVTAAPPRREQD, DOMVTAAPPRREQD, NONAPPRVTAUSED, ALLOWGLOVTAS, and AUTOGLOVTAS) help determine the way TMS processes source terms and omissions in fully integrated external systems, and omissions in partially integrated external systems.

If you change one of these settings after source terms and omissions have already been processed, the system does not automatically reprocess old data using the new setting. You must use the Force Rederivation job to mark study omissions (and source terms, in fully integrated external systems) for processing during the next Batch Validation or equivalent job.

If you are using TMS with Oracle Clinical, you can call the Force Rederivation job by selecting Oracle Clinical's **Plan** menu, choosing **TMS Domains**, then selecting **Studies**, **Domain Elements**, and then **Force Rederivation**. Alternatively, you could call through the API, using the procedure ocl_tms_pack.SetLastTMSDerivTS (see the *Oracle Thesaurus Management System Technical Reference Manual* for details). You must then run Oracle Clinical Batch Validation.

If you are using a different external system, you must mark records for reprocessing and then run the equivalent of Batch Validation to actually reprocess them.

Other Installation-wide Codelists

Descriptions of the TMS installation-wide codelists, other than the TMS_CONFIGURATION codelist, follow.

TMS_LANGUAGES

Values in this codelist populate the **Language** field in the Define Dictionaries window. English is the default dictionary language.

TMS_QUERY_TYPE

TMS_QUERY_TYPE populates the LOV in the **Query** field for several forms in TMS. The default value, STANDARD, provides a standard Oracle query, while the CONTEXT value enables you to use the Oracle *inter*Media option.

TMS_SOURCE_MAT_VIEWS

TMS_SOURCE_MAT_VIEWS controls whether materialized views or regular views are used to populates the LOVs for each external system value in the Filter window, which launches from Reclassify Verbatim Terms and Approve VTAs. Oracle recommends using materialized views for Values 1-3 when using TMS with Oracle Clinical. See "Controlling the Use of Materialized Views" on page 3-39.

TMS_TAL_POOL_CONFIGURATION

Set a value for VTOs (omissions), VTAs (here, unapproved VTAs only), and ACTs (here, unapproved Action assignments):

- **N (No Allocation)**. If set to N, Task Allocation is not available to omissions, unapproved VTAs, or unapproved Action assignments.

- **D (Direct Allocation)**. If set to D, TMS performs Direct Allocation as soon as an omission, unapproved VTA, or unapproved Action assignment is created. See "Direct Allocation" on page 9-1.

- **P (Pool Allocation)**. If set to P, TMS effectively puts omissions, unapproved VTAs, or unapproved Action assignments in a pool. They can then be allocated either manually or by invoking a pooled algorithm. "Manual Pool Allocation" on page 9-1 and "Automatic Pool Allocation" on page 9-1.

> **Note:** The Task Allocation menu item does not appear in the TMS Navigator panel unless the long value for at least one of the short values—VTO, VTA, or ACT—is set to either **D** or **P**. If all are set to **N**, **Task Allocation** does not appear in the Navigator.

TMS_X_SEARCH

TMS_X_SEARCH populates the LOV in the **Search Type** field of the Extended Search window. This reference codelist currently has just one value, Cross Search, which enables extended searches to search the entire TMS repository.

Local Codelists

TMS includes the following local reference codelists.

REPORTS_CONFIG

The local reference codelist REPORTS_CONFIG is required for TMS reports. There is one parameter, JDBC_DATA_SRC. You must set its long value to:

```
jdbc/opadatabase_nameDS
```

This same value must be in the WebLogic Server Administration Console. To check this:

1. Log in to the WebLogic Server Administration Console at:

   ```
   http://host:port/console/login/LoginForm.jsp
   ```

2. Navigate to **Services** under Domain Structure, then **Data Sources**.

3. Confirm that a data source with JNDI Name: `jdbc/database_nameDS`. (During installation the TMS Installer creates and names the data source.)

TMS_DSI

The local reference codelist TMS_DSI contains parameters whose settings determine important aspects of DSI behavior; see "Setting DSI Preferences with Reference Codelist Settings" on page 5-11.

WEB_DOCUMENT_CONFIG

This reference codelist is obsolete.

WEB_DOCUMENT_GROUPS

This reference codelist is obsolete.

Managing Distributed Locations

This section includes:

- Synchronizing Reference Codelist Settings on page 3-32
- Creating New Instances on page 3-32
- Performing Classification at All Instances on page 3-33
- Symmetric Replication on page 3-34

TMS is designed to be used in a global environment, with a single master instance linked to multiple slave, or local, instances. All definition and dictionary maintenance functions must be performed at the master instance, but classification is allowed at any instance (see "Performing Manual Classification at All Instances" on page 3-33). The master instance must be running to allow classification and browsing on a local instance.

TMS uses symmetric replication to replicate data among locations. See "Symmetric Replication" on page 3-34.

> **Note:** If you are using TMS with Oracle Clinical in a distributed environment, the TMS master instance must be on the same location as the Oracle Clinical Global Library instance, and you must use symmetric replication in Oracle Clinical as well as TMS.

Synchronizing Reference Codelist Settings

If you are using TMS in a distributed environment without Oracle Clinical, you need to use a batch job to replicate installation reference codelist settings. The TMS batch job Synchronize Reference Codelists appears under the **Definition** menu on local sites.

> **Note:** If you are using TMS with Oracle clinical in a distributed environment, TMS' reference codelist replication is automatically disabled and Oracle Clinical takes over that function.

To synchronize reference codelist settings, on each local instance:

1. From the **Definition** menu, select **Jobs**, then choose **Synchronize Reference Codelists**. The Synchronize Reference Codelists batch job window appears. There are no parameters to enter.

2. Use the runner icons in the toolbar to save and retrieve parameter sets as necessary, submit the job, and monitor its progress.

See "Setting Up and Running Reports and Other Batch Jobs" on page 2-13 for more information.

Creating New Instances

To add an instance, you must install the new database, create new database objects, import data from the master, and do a few other tasks; see instructions in the *Oracle Thesaurus Management System Installation Guide*.

You must run Synchronization on the new instance before you try to classify any terms. You should also schedule Synchronization to run regularly on the new instance. See "Synchronization" on page 3-36.

Performing Classification at All Instances

TMS allows each instance to classify omissions, though the master site serves as the central repository behind the scenes.

Automatic classifications are propagated to all instances during symmetric replication. Manual classifications and Actions are done interactively in real time at the master and the classifying instance, and are propagated to other instances during replication.

Reclassification can occur at the master site only, and high-level classification at the local sites only.

Using Autoclassification at Each Instance

When you run Autoclassification at any instance, either independently or, more typically, as part of Oracle Clinical Batch Validation or the equivalent job for a different external system, the job looks for direct matches for source terms in the dictionary's classification level(s).

- If Autoclassification finds a direct match in the VTA level, it puts the source term immediately into the Source Terms table with the ID of the VTA (dict_content_id).

- If Autoclassification finds a direct match in the DT classification level, it puts the source term in the Omissions table with a temporary classification ID (dict_content_tmp_id). (The Omissions table is globally maintained; see Symmetric Replication on page 3-34.) The next Synchronization on the master instance creates a new VTA and VTA ID (dict_content_id) for the source term. Each local instance receives the omission and its new dict_content_id during the next replication; each local instance, including the one where it was originally temporarily classified, must run Synchronization to move the source term into the Source Terms table with its classification (dict_content_id).

  If the classification occurs on the master instance in a replicated environment, it gets a temporary ID there too, until Synchronization. (If the master instance is not in a replicated environment, Autoclassification puts the source term directly into the Source Terms table with a dict_content_id.

- If Autoclassification does not find a direct match in any classification level, it puts the source term in the Omissions table as an omission (without either a dict_content_id or a dict_content_temp_id). The term must be manually classified, or an Action applied to it.

No conflicts should arise from Autoclassification, even if more than one instance processes the same new source term at the same time, because it uses direct matches only. The dict_content_id is simply the ID of the term to which the source term is classified, so both local instances assign the same dict_content_id to source terms that are a direct match.

Performing Manual Classification at All Instances

TMS allows manual classification (or Action application) to occur simultaneously at the master instance and all local instances on all omissions.

When a user manually classifies a term at a local instance, the system calls a package to immediately create the classification at the master instance. During the next replication, the source term and its VTA are propagated to all instances and the omission is updated. During the next Synchronization, the source term is moved out of the Omissions table and into the Source Terms table (see "Synchronization" on page 3-36).

> **Note:** To perform manual classification at a local instance, the master instance must be running.

Classification Conflict Resolution

During manual classification, conflicts are kept to a minimum because the process is centralized in the master instance. However, it is possible for two or more sites to collect the same term during the same period between replication jobs, and handle it differently. These conflicts are resolved at the master instance.

The following types of conflicts are resolved interactively in the user interface:

- If two sites classify the same term differently, the master uses the classification with the earlier timestamp and returns an error to the other site when the later classification is attempted.

- If two sites apply a different Action to the same term, the master uses the Action with the earlier timestamp, and returns an error to the other site when the later Action application is attempted.

The following types of conflicts are resolved during the next replication. The master resolves the conflict according to the rules below and propagates the classification or Action to all sites.

- If one site applies an Action to a term, and another site classifies the same term, the master uses the classification, not the Action.

- If a dictionary term is added on the master site that provides a direct match to a verbatim term but conflicts with an existing local classification, the master overwrites the local classification with a classification to the direct-match dictionary term.

- If one site changes an Action's ownership (from Oracle Clinical to TMS, for example) and another site applies a new Action to the same term, the master uses the Action with the changed owner.

- If one site assigns a Discrepancy Message to a term and another assigns a Global Action to the same term, the master uses the Discrepancy Message.

- If two sites apply different Discrepancy Messages to the same term, the master uses whichever Action is processed first during replication. The first Action processed is not necessarily the one with the earlier timestamp. It is not possible to predict which one will be used.

Symmetric Replication

Symmetric replication uses materialized views to maintain data consistency across multiple TMS instances in a distributed environment.

During installation you use scripts to create TMS tables on each local instance and then export data from the master to each local instance. You then start symmetric replication (see the *Oracle Thesaurus Management System Installation Guide*).

After installation, symmetric replication runs automatically behind the scenes to replicate data changes among TMS instances. First, the master instance pulls data changes from all the local instances. Then the local instances pull the cumulative changes from the master.

> **Note:** When an omission is created on a local instance—through Oracle Clinical Batch Validation or a similar process for a different external source data system—the omission is not available to a user at the local instance in the Classify VT Omission window until replication has been run.

TMS tables fall into several categories for replication:

- All TMS tables used in definition are maintained on the master only, but their data is replicated to local instances in a read-only state.

- Several tables are actually maintained at the master instance, but users at local instances appear to be able to change data in these tables. Changes are made on the master in real time. The changes are propagated to other instances during replication. Tables in this category include the dictionary contents and relations tables and Informative Notes tables.

  The changes allowed from local instances are allowed only during classification, and include creating VTAs and VTA IDs, and maintaining VTA Workflow Informative Notes.

- The Omissions and Actions tables are maintained globally. That is, any instance can classify any omission or apply an Action to any omission owned by TMS (not the external system). This process works differently depending on whether the classification is automatic or manual. During replication, each local instance's automatic classifications are replicated to the master and then replicated from the master to the local instances. Users manually classify omissions and apply Actions to omissions interactively. See "Using Autoclassification at Each Instance" on page 3-33 and "Performing Manual Classification at All Instances" on page 3-33.

- Some tables are maintained locally only and are never replicated. These tables include Source Terms (which stores locally collected source terms only), High-level Classifications, Domain Elements, Instance Dictionaries, and error log tables.

- Some tables are maintained at the master only and are never replicated. These are the "predict" tables used in the activation of dictionary terms and relations.

> **Note:** Information on managing replicated databases is available in the Oracle manual *Advanced Replication*.

Autoclassification

The Autoclassification process tries to code, or classify, terms from an external source data system by looking for dictionary terms that exactly match each source term. It also looks for exact matches to previously manually classified source terms. These are called Verbatim Term Assignments (VTAs) or, in the case of dictionaries with nonunique coding levels such as WHO-Drug C, Verbatim Term Individuals (VTIs). In the case of VTIs, the auxiliary information associated with the omission must match that of the VTI and autocoding with auxiliary information must be enabled. See "Nonunique Coding Level Dictionaries" on page 6-8.

If Autoclassification does not find a direct match or a previously created VTA or VTI for a term, it creates an *omission* that must be manually classified.

Using Search Objects

You can use search objects to supplement TMS's basic Autoclassification process, increasing the likelihood that TMS will find a match for a verbatim term, and specify the order in which you want TMS to execute them. Depending on the **Approval Type** setting in the Define Search Objects window, if TMS finds a match via a search object, it creates:

- An Omission with a Candidate Term

- An Approved VTA

- A Nonapproved VTA

See "Defining Search Objects" on page 7-10.

Domain Match

TMS provides one pre-packaged search object called Domain Match. If the Domain Match search object is enabled for the current dictionary/domain combination, TMS searches for a match in all other domains.

Custom Search Objects

You can also create custom search algorithms to supplement TMS's Autoclassification process. Create a PL/SQL function in the database and define the object in the Define Search Objects window. See "Creating Custom Search Algorithms" on page 7-14.

Creating Global VTAs during Autoclassification

Before running Autoclassification, consider whether you want the VTAs that you create to be available only within the domain in which you create them (**Domain VTA**) or available to all domains (**Global VTA**). Two settings in the TMS reference codelist TMS_CONFIGURATION control the default status of VTAs created during Autoclassification: ALLOWGLOVTAS and ALLOWGLOVTAS. If both are set to Y, Autoclassification creates global VTAs. If either is set to N, Autoclassification creates domain VTAs.

You can change the scope of the VTA after autoclassification. For information about promoting a VTA (making a Domain VTA available globally) or demoting a VTA (making a Global VTA specific to a domain), see "Promoting and Demoting VTAs" on page 11-1.

Running Autoclassification

When TMS is fully or partially integrated with an external system, the Autoclassification process is triggered by the batch job through which the two systems share information (for Oracle Clinical, this is Batch Validation). When there is no integration between TMS and an external system, you must run the tms_user_autocode package from the API.

You can also do a trial run of Autoclassification without updating the Omissions table by running the TryClassifying function in the tms_user_autocode API package, and view the results by looking in the function output. This allows you to see what the results of Autoclassification would be, without actually updating any data.

Synchronization

This section includes:

-

-

-

The TMS Synchronization batch job runs on individual TMS instances. It checks for changes in the TMS repository that impact the Omissions and Source Terms tables and:

- Updates records in those tables

- Marks the **Update** flag for the same records so that they are reprocessed during the next data exchange between TMS and the external system (Batch Validation in Oracle Clinical)

In a distributed environment, omissions can be classified at any instance (see "Performing Classification at All Instances" on page 3-33). Therefore, you must run Synchronization locally to update the Source Terms table with changes made at other sites.

If you have TMS integrated fully or partially with an external source data system, you must run Synchronization. TMS Synchronization is included in Oracle Clinical Batch Validation when the two systems are integrated.

If you are using TMS as a look-up system only, without integration to an external source data system (such as Oracle Clinical), you do not need to run Synchronization.

Scheduling Frequency

Oracle recommends that you run Synchronization at least five times during the business day at each site. You should also run Synchronization before and after batch jobs that exchange information between TMS and an external system. When TMS is integrated with Oracle Clinical, this happens automatically during Batch Validation.

Data Processed During Synchronization

The Omissions table contains source terms that Autoclassification could not classify. The Source Terms table includes all locally collected source terms and the classification (VTA ID or dict_content_id) for each one. When an omission is classified manually, the Synchronization job moves the source term from the Omissions table to the Source Terms table, associated with its classification. If a VTA is changed at any point, Synchronization changes the VTA ID in the Source Terms table.

The following types of changes require synchronization with the Source Terms table:

- Classifying source term omissions

- Declassifying and reclassifying verbatim terms

- Approving and unapproving VTAs

- Loading a new dictionary (to mark the dictionary for future processing)

- Loading a new dictionary version (to mark the dictionary version and its source terms and omissions for future processing)

- Changes to Informative Notes that are derived to the external system

In addition, if TMS is integrated with an external system, the following changes to source terms in the external system must be synchronized in TMS:

- Definitional changes such as question definition changes or mapping a study or project to a different domain in Oracle Clinical

- Deletion of source terms in the source system (Oracle Clinical responses)

Synchronization processes only locally owned data. Synchronization does not process omissions created at a different instance and replicated; these omissions are owned by the instance where they were created.

Synchronization runs on data imported using the TMS DSI feature; because the systems are not integrated, the importing instance owns the data.

Synchronization Processing Order

Synchronization always includes two phases, pre-synchronization and post-synchronization.

Pre-synchronization   Pre-synchronization runs first. It updates records as outlined "Data Processed During Synchronization" on page 3-37 and sets the Update flag to Y so that records will be updated in the external system during the next data exchange (Batch Validation in Oracle Clinical).

Post-synchronization   Post-synchronization runs last. It refreshes materialized views if the TMS_CONFIGURATION reference codelist value SYNCREFRESHMV is set to Y (see "SYNCREFRESHMV" on page 3-28).

In addition, when run at the master site, it executes the Purge Classified Omissions batch job (see "Purging Classified Omissions from the Omissions Table" on page 3-42).

If you choose not to refresh materialized views, which can take a long time (see "Controlling the Use of Materialized Views" on page 3-39) and the Synchronization job is not running on the master site, the post-synchronization phase is not executed.

External System Integration   If TMS is integrated with an external source data system, additional processing occurs between these two phases during data exchange between the two systems. The data exchange job must:

- Run full autocode on all source terms that have been changed since the last data exchange and on all records in the Source Terms and Omissions tables whose Update flag is checked

- Reset the Update flag on all processed records to N

Oracle Clinical Batch Validation is predefined to do both.

The batch processing unit (a study in Oracle Clinical) is called an X Area in TMS. The following indexes use the X Area to give fast access to source terms and omissions that need to be processed:

- TMS_SOURCE_TERMS_BI3

- TMS_VT_OMISSIONS_BI1

Both indexes include the following columns: NONUNIQUE X_AREA, NONUNIQUE DEF_INTEGRATION_KEY, NONUNIQUE UPDATE_FLAG.

> **Note:**   Although Batch Validation runs on only a single study, or TMS X Area, Synchronization always runs on all TMS data, even when run during Batch Validation.

Running Synchronization

To run Synchronization:

1. From the **Definition** menu, select **Jobs**, then **Synchronize Dictionary Data**. The Synchronization batch job window appears. There are no parameters to enter.

2. Use the runner icons in the toolbar to save and retrieve parameter sets as necessary, submit the job, and monitor its progress.

See "Setting Up and Running Reports and Other Batch Jobs" on page 2-13 for more information.

Writing a Procedure to Change the Dictionary Display

By default, the **Dictionary Term** field in the Classify VT Omissions and Reclassify Verbatim Terms windows displays the dictionary term to which the selected verbatim term is mapped. In some cases, however, you may require different information about a verbatim term's mapping within a particular dictionary to make a decision about classification—instead of the classification level term, you might want to see its parent term or another term in the hierarchy.

By defining a Dictionary Term Display Procedure for a TMS dictionary, you can dictate the level of dictionary term shown in the derivation path for a dictionary. You specify which procedure to use in the **Base Dictionary** tab window in the Define Dictionaries window.

If you leave the **Dictionary Term Display Procedure** field blank when defining the dictionary, TMS exhibits the default behavior: displaying the classification level term to which the selected verbatim term is mapped.

Writing a Dictionary Term Display Procedure

Any Dictionary Term Display Procedure must be written with the input and output parameters shown below. Oracle recommends that you create the procedure from the TMS role.

```
PROCEDURE procedure_name (
   pDictContentId IN NUMBER
   , pDefLevelId IN NUMBER
   , pTermType IN NUMBER
   , pCutOffDate IN DATE
   , pReturnLevel IN OUT VARCHAR2
   , pReturnTerm IN OUT VARCHAR2
```

After you write a procedure, grant the proper execute privilege to the tms_classify_priv role by issuing the following statement from TMS (or the role you used to write the procedure):

```
grant execute on procedure_name to tms_classify_priv
```

Controlling the Use of Materialized Views

TMS can use materialized views to populate the lists of values for external system columns in the Filter window of Reclassify Verbatim Terms and Approve VTAs. Using materialized views accelerates populating many of these LOVs, but in databases with a high data volume, refreshing materialized views during Synchronization can be too slow.

If the materialized views refresh during Synchronization is slowing Synchronization at your database, you have two options:

- You can reduce the number of external system columns that rely on materialized views, switching some or all of them to regular views. To perform this switch, follow the steps in "Choosing Materialized or Regular Views for Each External System Value" on page 3-40, then create the new views by "Creating Source Term Views" on page 3-40.

- You can remove the materialized views refresh from the Synchronization process. To de-couple this refresh from Synchronization, perform the steps in "Including the Materialized Views Refresh in the Synchronization Process" on page 3-40. If you use materialized views for any external system values, and you have made this refresh process independent of Synchronization, you must still perform the refresh periodically. See "Refreshing Source Term Materialized Views" on page 3-41.

Choosing Materialized or Regular Views for Each External System Value

The installation reference codelist TMS_SOURCE_MAT_VIEWS controls whether regular views or materialized views are used to populate the list of values for external system values in the Filter window. Each value in the codelist controls the view behavior of one external system field: for example, setting the long value of the codelist value MV1 to Y will make TMS use a materialized view to populate the list of values for the external value 1 in these Filter windows.

By default, TMS_SOURCE_MAT_VIEWS specifies that materialized views be created for external values 1 to 5, and regular views for external values 6 to 8.

Whenever any of the values in TMS_SOURCE_MAT_VIEWS are changed, you must run the Create Source Terms Views batch job. See "Creating Source Term Views" on page 3-40.

To change the view settings for an external system value:

1. Open the Installation Reference Codelists window. (From the **Definition** menu, select **Installation Reference Codelists**.)

2. Query for the TMS_SOURCE_MAT_VIEWS codelist.

3. Update the long value for each external system value that you want to change.

4. Save.

Creating Source Term Views

Whenever you change any of the values in the TMS_SOURCE_MAT_VIEWS reference codelist, TMS must create a new view for each external system value you changed. Run the Create Source Terms Views batch job to create the materialized or regular views that your change requires.

To run this job from the user interface:

From the **Definition** menu, select **Jobs**, then choose **Create Source Terms Views**.

To run this job from a SQL*Plus prompt, execute the following API call:

tms_user_data_admin.configSTMatViews

Including the Materialized Views Refresh in the Synchronization Process

You can control whether TMS refreshes materialized views during the Synchronization process by changing the reference codelist value SYNCREFRESHMV in the TMS_

CONFIGURATION reference codelist. If this setting is Y, the materialized views refresh will be included in Synchronization.

If you remove the materialized views refresh from Synchronization, you must periodically run the Refresh Source Term Materialized Views batch job to refresh them manually. See "Refreshing Source Term Materialized Views" on page 3-41.

To change the inclusion status of materialized views in Synchronization:

1. Open the Maintain Installation Codelists window. (From the **Definition** menu, select **Installation Reference Codelists**.)

2. Query for the TMS_CONFIGURATION reference codelist.

3. Scroll down to the SYNCREFRESHMV row, and change its long value setting. A long value of Y includes the materialized views refresh in Synchronization; N removes this refresh from Synchronization.

4. Save. TMS will either couple or de-couple the materialized views refresh and the Synchronization process.

Refreshing Source Term Materialized Views

If you do not refresh the materialized views as part of Synchronization, you must refresh them periodically by running the Refresh Source Term Materialized Views batch job.

To run this job from the user interface:

From the **Definition** menu, select **Jobs**, then select **Refresh Source Term Mat. Views**.

To run this job from a SQL*Plus prompt, execute the following API call:

tms_user_synchronization.RefreshMviews

Enabling Context Searches for Non-English Dictionaries

Context searches enable you to be more flexible in your queries. You can perform context searches for word fragments, terms that share a common root, terms that sound similar, or terms that have the same meaning. For more information on context searches, see "Entering a Context Query" on page 2-10.

By default, TMS is configured to perform context searches for English-language dictionaries only. However, because TMS indexes information in a multi-lexer, a data structure that can store information in multiple languages; you can configure TMS to allow context searches in other languages.

To add a new language to the Context Server Index:

1. From the Install directory, open the following script in a text editor:

   tms\database\tmscincontextinx.sql

   This script includes sample commands for adding Japanese, Danish, or German to the Context Server Index. To add one or more of these languages to your index, copy the appropriate lines to a buffer.

   To add a different language, modify the lines to suit the language you want. For more information, see the System-Defined Preferences section of the Indexing chapter in the *Oracle Text Reference* manual (part number A96518).

2. Start a SQL*Plus session as TMS.

3. Run the copied or edited lines.

4. Refresh the Context Server Index by running the job from the **Definition** menu. See "Refreshing the Context Server Index" on page 3-42.

Refreshing the Context Server Index

When you insert new dictionary terms or VTAs into the TMS repository, the Context Server cannot find them in a query until you refresh the Context Server Index. Oracle recommends that you schedule this refresh as a batch job running as often as is appropriate for your TMS installation.

You can refresh the Context Server Index by running the Refresh Context Server Index batch job.

To refresh the Context Server Index:

1. From the **Definition** menu, select **Jobs**, then **Refresh Context Server Index**. The Refresh Context Server Index batch job window appears. There are no parameters to enter.

2. Use the runner icons in the toolbar to save and retrieve parameter sets as necessary, submit the job, and monitor its progress.

See "Setting Up and Running Reports and Other Batch Jobs" on page 2-13 for more information.

Analyzing Tables

TMS automatically runs the Analyze Tables batch job after Activation to optimize TMS performance. You can also schedule this batch job to run periodically; from the Definition menu, select **Jobs**, then **Analyze Tables**, and specify the batch job's parameters.

See "Setting Up and Running Reports and Other Batch Jobs" on page 2-13 for more information.

Purging Classified Omissions from the Omissions Table

A conflict can arise when users delete omissions in distributed TMS environments. If a user deletes an omission at one site, then another user updates the same omission at another site, the Oracle Symmetric Replication Packages return an error ("ORA-1403 no-data-found") and the deletion does not propagate through the TMS sites.

To avoid conflicts like this one, TMS deletes omissions in two stages:

■ When you choose to delete an omission from your site, TMS performs a "logical deletion" of that omission's row in the tms_vt_omissions table. Logical deletions do not remove rows from this table, but do flag these rows for later deletion from the table.

■ The Definition batch job Purge Classification Omissions physically deletes these marked rows from the tms_vt_omissions table. Only users who have the tms_define_priv role can run this procedure.

The Purge Classification Omissions job runs on the master site only as part of symmetric replication. TMS follows this logical deletion structure whether you use symmetric replication or not. We strongly recommend that you run this procedure weekly for your TMS installation. Running the procedure at one site will purge classified omissions from that site only. TMS automatically runs this job as part of Synchronization when you run Synchronization from the master site.

Running Scripts to Gather Schema Statistics for the 11g Optimizer

After upgrading from a release prior to TMS 5.0 and setting initialization parameter **optimizer_features_enable** to 11.2.0.3 you must gather statistics required for the Oracle 11g Optimizer to be effective for accounts used internally by TMS.

Failure to execute these scripts can negatively impact performance.

Scripts opastats.sql and tmsstats.sql, are available for this purpose. If your database contains large amounts of data, the scripts may take a long time to run. You may want to edit the scripts; for information on opastat.sql and tmsstats.sql parameters see the following documentation of the DBMS_STATS package that they call: http://docs.oracle.com/cd/B19306_01/appdev.102/b14258/d_stats.htm#i1036456.

> **Note:** This information is also in the installation guide. You need to do this only once after installation. You should also do it after loading a dictionary. It is especially important after loading WHO-Drug Format C.

tmsstats.sql   Since TMS already has a procedure, AnalyzeTable in tmspb_user_analyze.sql, that runs as part of every activation job, you may prefer to edit it if required and call it from tmsstats.sql to keep the two synchronized.

The tmsstats.sql script prompts for the password of the account it processes: TMS.

To change how TMS analyzes tables and ensure that tmsstats and AnalyzeTable remain in synch:

1. Modify the Analyze procedure in tmspb_user_analyze.sql.

2. Edit tmsstats.sql to call tms_user_analyze.AnalyzeTable. Replace the section that analyzes tables with:

    ```
    exec tms_user_analyze.AnalyzeTables
    ```

3. Run tmsstats.sql.

opastats.sql   captures new statistics in the OPA application account used by TMS. It is also used by Oracle Clinical and RDC Onsite.

Adding and Configuring WebLogic Server Data Sources

When you run the TMS Installer, it sets up a connection, or data source, for OPA and the HTML Browser from the WebLogic Server to the default database.

If you plan to use more than one database, you must create an OPA and a HTML Browser data source for each additional database. The OPA data source is used to run TMS reports.

> **Note:** In addition, you must modify **tnsnames.ora** at C:\Oracle\Middleware\asinst_1\config. Ensure that it contains an entry for each database.
>
> Oracle recommends maintaining a master tnsnames.ora file and copying it to each location—database and application servers—whenever you add a database.

Add Data Sources

Go through these instructions for one data source at a time.

1. Log in to the WebLogic Server Administration Console at:

   `http://host:7101/console/login/LoginForm.jsp`

2. Select Data Sources, then New, then Generic Data Source.

3. Enter a name for the data source.

4. Enter the JNDI name as:

   - For TMS: *jdbc/tmsdatabase_name*DS

   - For OPA: *jdbc/opadatabase_name*DS

   > **Note:** Enter the JNDI name for OPA in the REPORTS_CONFIG local reference codelist, JDBC_DATA_SRC setting.

5. On the next screen:

   - If the database you want to connect to is a RAC database, choose Oracle's Driver (Thin) for Service connections; Versions:9.0.1 and later

   - If the database is not a RAC database, choose Oracle's Driver (Thin) for Instance connections; Versions 9.0.1 and later

6. On the next screen, uncheck **Supports Global Transactions**.

7. On the next screen, enter the database name, host name, and port for the database. For the user, enter the following user account and its password:

   - For OPA: `bc4j_internal`

   - For HTML Browser: `ops$tmsbrowser`

8. On the next screen, select **Test Configuration** to validate if you can connect using the pool setup.

9. On the next screen, assign `OpaServer1` as the target.

10. Click **Finish**.

11. Select the newly created Data Source, click the **Connection Pool** tab, and then click **Advanced**.

12. Add the property `dataSourceName=OracleDataSource` into the **Properties** field (without quotes).

13. If the following property has not already been created by the Create Data Source wizard, add the property `databaseName=SID_or_servicename` into the **Properties** field (without quotes).

14. Uncheck **Wrap Data Types** and save.

15. Review and set additional parameters in Advanced Settings as appropriate for your expected data load.

16. Activate your changes or restart OpaServer1 for the changes to take effect.

Configure Maximum Connections

By default, the value for the Maximum Capacity parameter for each data source, which determines the maximum number of concurrent connections, is 15.

The value for the **Maximum Capacity** parameter should be equal to or greater than the maximum number of concurrent users you expect for:

- TMS HTML Browser (TMS data source)
- TMS Reports (OPA data source)

To change this value:

1. Log in to the WebLogic Server Administration Console at:

   ```
   http://host:7101/console/login/LoginForm.jsp
   ```

2. Select Data Sources, then select the data source whose setting you want to modify: for example, DS_1_tms for TMS or DS_1_ops for OPA (TMS reports).

3. Click the **Connection Pool** tab.

4. Scroll down to find the parameter **Maximum Capacity**.

5. Enter a value as described above.

6. Save.

You must then **restart the OPA Server**:

1. In the same OPA domain WLS Administration Console, click on **Servers**, then the **Control** tab.

2. Select **OpaServer1** and then **Shutdown - Force Shutdown now**.

3. Click **Refresh** to ensure the **State** changes to **Shutdown**.

4. Select OpaServer1 and then **Start**.

5. Click **Refresh** to see the **State** change to **Running**.

Configure Inactive Connection Timeout

To ensure that a connection is available for a user who needs to run a report, change the default setting of the **Inactive Connection Timeout** of 0 to 5 or a similar value. This is the number of seconds of inactivity that is allowed before the connection is made available to another user. If set to zero, the feature is disabled.

To change this value:

1. Log in to the OPA domain WebLogic Server Administration Console at:

   ```
   http://host:7101/console/login/LoginForm.jsp
   ```

2. Select Data Sources, then select the data source whose setting you want to modify: for example, DS_1_ops for OPA (TMS reports).

3. Click the **Connection Pool** tab.

4. Scroll down to find the parameter **Inactive Connection Timeout**.

5. Enter a value; for example, 5.

6. Save.

You must then restart the OPA Server:

1. In the same OPA domain WLS Administration Console, click on **Servers**, then the **Control** tab.

2. Select **OpaServer1** and then **Shutdown - Force Shutdown now**.

3. Click **Refresh** to ensure the **State** changes to **Shutdown**.

4. Select OpaServer1 and then **Start**.

5. Click **Refresh** to see the **State** change to **Running**.

TMS HTML Browser Administration

This section describes the following administrative topics:

- Granting User Privileges for Using the TMS HTML Browser on page 3-46
- Creating a Launch Page, Button, or Link for the TMS HTML Browser on page 3-46
- Customizing the TMS HTML Browser URL to Support Additional Databases or Automatic Login on page 3-49
- Enabling and Disabling Automatic Login on page 3-49
- Enabling TMS HTML Browser Logs on page 3-50
- Using the About Window on page 3-51

Granting User Privileges for Using the TMS HTML Browser

Users require the rxclin_read privilege to browse dictionary data. See "Security" on page 3-1 for information about creating TMS users and granting them roles appropriate to their work.

Creating a Launch Page, Button, or Link for the TMS HTML Browser

You can customize the shipped TMS Launch page to add a link to the TMS HTML Browser, create a new Launch page, or create a button or link to the HTML Browser from an existing website such as your users' Home page. Oracle provides scripts that help prevent problems that may arise in the HTML Browser.

In addition, you may want to customize the TMS HTML Browser URL:

- If you have multiple databases, you can specify a database other than the default one set up during installation.
- You can enable automatic login, so that the system automatically logs in any user as ops$tmsdbrowser.

> **Note:** By default, automatic login is disabled.

See "Customizing the TMS HTML Browser URL to Support Additional Databases or Automatic Login" on page 3-49.

Creating or Customizing a Launch Page to Use Provided Scripts

Oracle recommends using the scripts provided here to control when and how the HTML Browser Login page opens, to prevent users from having problems in the HTML Browser application resulting from:

- Using the browser's **Back** button—Sample Java Script for Launching in a New Window without Tool or Menu Bar suppresses the browser toolbar, making the **Back** button unavailable.

- Turning on IE's Compatibility View—Sample JavaScript for Checking if Compatibility View Is On detects if Compatibility View is turned on and if it is, does not open the application and displays a message telling the user to turn it off.

- Opening more than one HTML Browser session to the same application server or from the same parent window—the scripts prevent this.

To use these scripts in your launch page:

1. Place the script from "Sample Java Script for Launching in a New Window without Tool or Menu Bar" on page 3-47 in a file called Launch.js.

2. Place the script from "Sample JavaScript for Checking if Compatibility View Is On" on page 3-48 in a file called IeCompatCheck.js.

3. Place the scripts at the same location as the Launch page or in a folder (called JS in the example code below) in the same location as Launch page. The shipped TMS Launch page, launch.htm, is located in *opa_home*/html.

4. Create an HTML page that contains a button or link to the TMS HTML Browser that invokes the functions in the JavaScript files or add code to the shipped Launch page or an existing website to do the same.

   Put the code referencing the scripts near the top of the HTML page; for example:

   ```
    <title> Launch RDC Onsite </title>
     <script type="text/javascript" src="JS/IeCompatCheck.js"></script>
     <script type="text/javascript" src="JS/LaunchJS.js"></script>

    </head>
   <body>
   ```

   Add button code or link code as follows, and see "Customizing the TMS HTML Browser URL to Support Additional Databases or Automatic Login" on page 3-49.

   ```
   <script language="JavaScript">
   document.write('<FORM>');
   document.write('<input type="button" value="TMS Browser" onclick="return
   launch(event,\'../tmsadf/faces/Login\',\'TMSLite\')" class="loginButton"/>');
   document.write("</FORM>");
   </script>

   <noscript>
   <a onclick="return launch(event,'../tmsadf/faces/Login',\'TMSLite\')"
   href="#">Launch TMS Browser</a>
   </noscript>
   ```

Sample Java Script for Launching in a New Window without Tool or Menu Bar

```
function launch(event,url,windowname){
          if(detectIECompatibility(event)){
              cancelEvent(event);
              return false;
          }
          if(!!!windowname){
              windowname="_blank";
          }

window.open(url,windowname,'location=0,resizable=1,toolbar=0,titlebar=1,status=1,s
```

```
crollbars=1,menubar=0','true');
}
```

Sample JavaScript for Checking if Compatibility View Is On

The following script checks if Compatibility View is on and if so, displays a message asking the user to turn it off.

```
function IECompatDetect(){
        this.isCompatOn = function(){
                    var ua = navigator.userAgent;
                    var on = false;
                    for(var index=0; index< this.ieUAProps.length; index++){
                        var ieProp = this.ieUAProps[index];
                        if(ua.search(ieProp.layoutEngine) == -1){
                            continue;
                        }
                        if(ua.search(ieProp.browser) == -1){
                            on = true;
                            break;
                        }
                    }
                    return (on);
                };
        this.ieUAProps= [
                    {
                        browser: 'MSIE 10',
                        layoutEngine: 'Trident/6'
                    },
                    {
                        browser: 'MSIE 9',
                        layoutEngine: 'Trident/5'
                    },
                    {
                        browser: 'MSIE 8',
                        layoutEngine: 'Trident/4'
                    }
                    ]

    }
function cancelEvent(e){
    var event = e || window.event;
    if(event == null){
        return;
    }
    if(event.preventDefault){
        event.preventDefault();
    }else{
        event.returnValue=false;
        event.cancelBubble=true;
    }
}

function compatCheck(event,isCompatRequiredToBeOn){
    var compatOn=(new IECompatDetect()).isCompatOn();
    if(compatOn && !isCompatRequiredToBeOn){
        cancelEvent(event);
        alert("Please turn off IE's Compatibility View. Go to Tools->Compatibility
View/Compatibility View Settings to do so.");
    }
    if(!compatOn && isCompatRequiredToBeOn){
```

```
        cancelEvent(event);
        alert("Please turn on IE's Compatibility View. Go to Tools->Compatibility
View/Compatibility View Settings to do so.");
    }
    return compatOn;
}
function detectIECompatibility(e){
    return compatCheck(event,false);
}

//Call on load
(function(){
    compatCheck(null,false);
})();
```

Customizing the TMS HTML Browser URL to Support Additional Databases or Automatic Login

The format of the HTML Browser URL is:

```
https://server.domain/tmsadf/faces/Login?setUpDone=Y
&parameter1=value1&parameter2=value2
```

This URL, without any parameters, goes to the default database set up during installation and requires the user to log in. You can use one or more parameters to change this behavior:

Table 3–1   Parameters Used to Customize the URL

Parameter	Parameter Value	Description
db	*database_name*	Links to a database that is not the default one; see "Adding and Configuring WebLogic Server Data Sources" on page 3-43. The database name must be the JNDI name defined in the data source.
event	doAutoLogin	Helps enable automatic login; see"Enabling and Disabling Automatic Login" on page 3-49.

For example, when both parameters are used:

```
https://server.domain/tmsadf/faces/Login?setUpDone=Y & db=database&
event=doAutoLogin
```

Enabling and Disabling Automatic Login

If automatic login is enabled, any user can click a link for the TMS HTML Browser and the application opens immediately, bypassing the Login page. The system passes the credentials for the ops$tmsbrowser user, using the password entered in the TMS Installer. Users who log in with automatic login cannot perform searches on patient data.

By default, automatic login is disabled and when users click the link the Login page opens and they must enter their own credentials.

Enabling Automatic Login

To enable automatic login:

1. Open the tmsconfig.properties, which is located at c:\opapps51\config.

2. Search for the parameter tms_auto_login. If it is not present, add it.

3. Set it to Y:

```
tms_auto_login=Y
```

4. Save.

> **Note:** You must also add the string `event=doAutoLogin` to the TMS HTML Browser URL; see "Customizing the TMS HTML Browser URL to Support Additional Databases or Automatic Login" on page 3-49..

Disabling Automatic Login

Automatic login is disabled by default.

The setting of the parameter tms_auto_login in the tmsconfig.properties file overrides the URL event setting. If it is set to any value other than Y, or if it is not present in the tmsconfig.properties file, even if the TMS HTML Browser URL includes `event=doAutoLogin`, the link goes to the Login page instead of automatically logging in the user as ops$tmsbrowser.

To disable automatic login:

1. Open the tmsconfig.properties, which is located at c:\opapps51\config.

2. Search for the parameter tms_auto_login.

3. If it is set it to Y, change its setting to N.

4. Save.

Enabling TMS HTML Browser Logs

Do the following to enable exception log files for the TMS HTML Browser. A file named OpaServer-tms_1.log will be created in *Middlware_HOME*\user_projects\domains\OPADomain\ servers\OpaServer1\logs.

1. Open the logging config file logging.xml that is located at:

Middlware_HOME\userprojects\domains\OPADomain\config\fmwconfig\servers\OpaServer1

2. Add the following entry:

```
<?xml version='1.0' encoding='UTF-8'?>
<logging_configuration>
<log_handlers>
..
..
 <log_handler name='tmshandler'
class='oracle.core.ojdl.logging.ODLHandlerFactory'
filter='oracle.dfw.incident.IncidentDetectionLogFilter'>
   <property name='path'
value='${domain.home}/servers/${weblogic.Name}/logs/${weblogic.Name}-tms_1.log

'/>
   <property name='maxFileSize' value='10485760'/>
   <property name='maxLogSize' value='104857600'/>
   <property name='encoding' value='UTF-8'/>
   <property name='useThreadName' value='true'/>
   <property name='supplementalAttributes'
value='J2EE_MODULE.name,composite_name,component_name'/>
  </log_handler>
```

```
..
..
</log_handlers>
<loggers>
..
..
  <logger name='oracle.pharma.tms.logging' level='ALL'
useParentHandlers='false'>
   <handler name='tmshandler'/>
  </logger>
..
..
</loggers>
</logging_configuration>
```

3. Add the following to the URL to get all debug information:

   ```
   ?setUpDone=Y&debug=all
   ```

Using the About Window

The About window shows information about the TMS and other Oracle Health Sciences releases that have been applied to this database. Choose either TMS or OPA from the Installed Product list to browse:

- Which releases have been applied to the middle and database tiers.

- The dates in which these releases were installed.

Troubleshooting

Report Error

If you are getting an error message like "Java exception" followed by another message like "http://127.0.0.1/opaxdo/getXdoReport not found" when you try to run a TMS report, you may want to modify the setting of registry variable OPA_LOCAL_MT_ URL. This variable is used to redirect the server to the opaxdo servlet to get reports. When not defined, http://127.0.0.1 is used as default value.

Depending on the settings on middle tier, you may encounter this problem. To fix it, add a port number to the registry variable OPA_LOCAL_MT_URL value on the middle tier; for example, `http://127.0.0.1:7777`.

To confirm that the path in the registry variable is correct, submit the value such as `http://127.0.0.1:7777` (or, without port, `http://127.0.0.1`) from a browser on the middle tier. If the registry value is correct, you receive the message "Got Request."

Restart Application Server After Redeploying TMS HTML Browser

If you redeploy the HTML Browser .ear file—for example, to apply a patch—restart the application server, including the WebLogic Admin Server. This helps clear connection pools and any DLLs that may be loaded.

If you do not restart the server, an error message may be displayed the next time a user opens the HTML Browser that includes:

```
java.lang.UnsatisfiedLinkError: Native Library
```

Cannot Log In to Application

If you cannot log in to the TMS HTML Browser, you may have entered the application's URL twice in the same browser window.

Close all browser windows and try the URL again.

4

Integrating TMS with Oracle Clinical

This section includes:

-

-

-

-

-

-

When Oracle Clinical and Oracle Thesaurus Management System (TMS) are fully integrated, you can classify the raw patient data entered in Oracle Clinical to standard terms from dictionaries such as MedDRA and WHO-Drug.

After a response is collected to a question that is marked for processing by TMS, during the next Batch Validation the response is sent to TMS as a verbatim term. TMS tries to classify the verbatim term using Autoclassification. If that is not possible, TMS creates an omission that must be handled manually and, during the next Batch Validation, Oracle Clinical creates a discrepancy for the response.

After the verbatim term is classified, during the next Batch Validation TMS returns the classification and any other information you have defined (such as term ID, alternate code, or user-defined details) to Oracle Clinical as derived question response values in the same RDCM as the original collected question response. If a discrepancy had been associated with the response, it is automatically resolved.

In addition, Oracle Clinical passes information about the context of each collected question response to TMS, where you can use it to help with manual classification and other tasks. This key information—Patient, Study, Project, DCM, Investigator, Visit, Document Number and Discrepancy ID—remains stored in TMS (see "Defining External System Information in TMS" on page 5-28 for more information). If you reclassify a verbatim term in TMS, during the next Batch Validation the system uses the key information to update the derived responses for each occurrence of the term/question response in Oracle Clinical.

To set up Oracle Clinical for integration with TMS, you must define **parent questions** to collect the response in Oracle Clinical, **derived questions** to receive values from TMS, and **question sets** to link the two and specify the information to be retrieved from TMS. You must also associate TMS domains with Oracle Clinical studies or projects.

To define these objects, you must have access to Oracle Clinical windows that appear only when TMS is installed with Oracle Clinical. The following table shows the privilege required to view each window.

Table 4–1   TMS-Related Windows in Oracle Clinical

Window	Oracle Clinical Navigation Path	Privilege Required
Question Sets	From the **Glib** menu, select **Question Sets**, then choose **Question Sets**	rxc_gl_full
Qry Question Sets	From the **Glib** menu, select **Question Sets**, then choose **Qry Question Sets**	rxc_any
TMS Domain Elements	From the **Plan** menu, select **TMS Domains**	tms_define_priv
Qry TMS Domains	From the **Plan** menu, select **Qry TMS Domains**	rxc_any

> **Note:**   You must also have TMS superuser privileges or the drop-down list to select a dictionary in the TMS Domain Elements window will be empty; see "Defining a New User" on page 3-5 for information on granting superuser privileges.

Managing the TMS/Oracle Clinical Workflow

This section includes:

- First Review in TMS on page 4-3

- First Review in Oracle Clinical on page 4-4

- Setting First Review on page 4-4

You can choose to review discrepancies/omissions first in either Oracle Clinical or TMS.

Regardless of whether you set First Review to Oracle Clinical or TMS, during Batch Validation all Oracle Clinical question responses collected since the last Batch Validation and marked for processing by TMS (see "Defining a Parent Question" on page 4-12) are sent to TMS and inserted into the TMS SOURCE_TERMS TABLE with all their associated key data. Batch validation triggers TMS Synchronization and Autoclassification.

During Autoclassification, TMS searches for an exact match to a dictionary term or an existing VTA linking the same Oracle Clinical response/TMS verbatim term to a dictionary term.

For each term that is automatically classified:

- In TMS, the verbatim term is associated with a VTA linking it to a dictionary term.

- During the next Batch Validation, TMS sends to Oracle Clinical all the derived data defined in the question set variables.

> **Note:**   If TMS is unable to derive a related term in one of the requested dictionary levels, it creates a high-level omission within TMS and sends all the derived values possible back to Oracle Clinical. When the high-level omission is resolved in TMS, TMS sends the remaining derived value(s) to Oracle Clinical. See "Resolving High-Level Omissions" on page 12-17.

For each term that cannot be automatically classified:

- TMS creates an omission.

- During the next Batch Validation, Oracle Clinical creates a discrepancy.

- Manual intervention is required; the workflow depends on whether First Review is set to TMS or Oracle Clinical. (See "Setting First Review" on page 4-4.)

Normally you will want to handle omissions in TMS, because there you can specify how to handle the same verbatim term automatically in the future. However, you may prefer to check the data in Oracle Clinical first to correct any errors and send the data back to TMS for Autoclassification after it has been corrected.

A description of the workflow for First Review in TMS and First Review in Oracle Clinical follows.

First Review in TMS

If a term cannot be automatically classified and First Review is set to TMS, Batch Validation creates the Oracle Clinical discrepancy of category type Missing_PT with a status of TMS IN PROGRESS, which cannot be changed within Oracle Clinical.

You must handle the omission in TMS. You have two options: Manual Classification and Applying an Action.

Manual Classification

To classify a term manually after it fails Autoclassification, you must do one of the following:

- Create a new VTA to link the verbatim term to a dictionary term. I

- Add a new company term to the dictionary and create a VTA linking the verbatim term to the new term.

After you have manually classified a verbatim term, the next Batch Validation:

- Sends to Oracle Clinical all the derived data defined in the question set variables.

> **Note:** If TMS is unable to derive all the data requested, it creates a high-level omission within TMS and sends all the derived values possible back to Oracle Clinical. When the high-level omission is resolved in TMS, TMS sends the remaining derived value(s) to Oracle Clinical. See "Resolving High-Level Omissions" on page 12-17.

- Gives the discrepancy a review status of CLOSED and a resolution status of:
  - VTA CREATED if the omission was resolved in TMS by creating a new VTA.
  - DT CREATED if the omission was resolved by creating a new dictionary term.

Applying an Action

If you cannot classify a verbatim term because of a flaw in the term, you can assign an Action to the term to send a message back to Oracle Clinical requesting a modification to the verbatim term. For example, Oracle Clinical may send a question response/verbatim term to TMS that includes two terms, such as "headache and nausea." Because TMS does not support combined verbatim terms, you might want to apply an Action to this verbatim term with the text "please split this term." During the

next TMS Synchronization, the system sends this message back to Oracle Clinical as a comment associated with the discrepancy for the question response.

When the you apply an Action to an omission, the next Batch Validation:

- Changes the status of the discrepancy according to the status specified in the definition of the TMS Action. Normally this is INV REVIEW (Investigator Review) to indicate that the investigator must take the next step in resolving the discrepancy, but it could be any valid status.

- Inserts a message in the discrepancy's **Comment** field requesting clarification or other modification of the verbatim term.

In Oracle Clinical, the person responsible for resolving the discrepancy must make the requested change, if appropriate, and change the discrepancy's review status to TMS EVALUATION. During the next Batch Validation, the updated term is sent back to TMS for Autoclassification and the discrepancy's status is changed to TMS IN PROGRESS.

See "Defining and Using Actions" on page 7-6 for more information.

Sending a Discrepancy Message

You can send a message to Oracle Clinical regarding a single source term in the Classify VT Omissions window. A Discrepancy Message is not predefined; you enter text specifically for a particular source term. See "Applying a Discrepancy Message" on page 10-8.

First Review in Oracle Clinical

If a term cannot be automatically classified and First Review is set to Oracle Clinical, Batch Validation creates Oracle Clinical TMS discrepancies with a status of UNREVIEWED. You can then review each discrepancy and correct any errors. After correction (or at any time), set the discrepancy status to TMS EVALUATION. The next Batch Validation will change its status to TMS IN PROGRESS and pass it to TMS for Autoclassification. Corrected terms may be successfully auto. If not, an omission is created and you must handle it in TMS (see "First Review in TMS" on page 4-3).

Setting First Review

Two settings determine which system handles the First Review, both set in Oracle Clinical: the **Review Before TMS?** setting in the Question Sets window and the FIRST_REVIEW short value in the TMS_OPTION reference codelist. The interaction between the two settings is as follows:

- If the reference codelist value is NO (so TMS manages omissions), the value set in the Question Set window is ignored.

- If the reference codelist value is YES (so Oracle Clinical manages omissions), clearing the **Review Before TMS?** box overrides the database setting and directs omissions to TMS.

In other words, to have Oracle Clinical do the first review of omissions, you must set both the reference codelist value to YES and select **Review Before TMS?** in the Question Set window. All other setting combinations result in a first review by TMS.

Batch Validation in an Integrated OC/TMS Environment

This section includes:

OC and TMS Data Exchanged During Batch Validation

Oracle Clinical Batch Validation runs on a single study, usually on a regular schedule such as nightly.

When Oracle Clinical and TMS are integrated, the Batch Validation job finds data that is new or changed in each system since the last Batch Validation and sends the relevant data to the other system. During the same Batch Validation job, TMS Autoclassification and Synchronization run in TMS on new Oracle Clinical data, and derived information, omissions, and Actions associated with that same data are sent back to Oracle Clinical.

During Batch Validation:

- Oracle Clinical sends new parent question responses, plus their key information, to TMS. If responses have been deleted in Oracle Clinical, the deletions are sent to TMS.

- Oracle Clinical sends responses whose question set definition has been changed.

- Oracle Clinical sends discrepancies newly given a status of TMS EVALUATION.

- TMS sends the derived values specified in the parent question's question set to questions whose parent question response has been classified, reclassified, or declassified in TMS.

- TMS sends derived values to questions corresponding to high-level TMS omissions that have been resolved in TMS.

- In the Oracle Clinical Discrepancy Database, Batch Validation resolves any discrepancies associated with question responses/verbatim terms that have been manually in TMS.

- In the Oracle Clinical Discrepancy Database, Batch Validation applies any Actions that have been assigned to omissions in TMS to the discrepancy associated with the original question response. The Action Text is displayed in Oracle Clinical as the Internal Comment for the discrepancy.

- Batch validation creates Oracle Clinical discrepancies for any terms TMS was unable to classify automatically (omissions). If First Review is set to TMS, these discrepancies have a status of TMS IN PROGRESS; if First Review is set to Oracle Clinical, they have a status of UNREVIEWED.

Running Only the TMS Portion of Batch Validation

Running only the TMS portion of Batch Validation may be useful when you are recoding a large amount of data but do not want to run other components of Batch Validation (such as Oracle Clinical Procedures). For example, if you have upgraded your dictionary version and anticipate a large number of terms to be processed.

To run only the TMS portion of Batch Validation, open Oracle Clinical. From the **Conduct** menu, select **Data Validation**, then **TMS Derivation**.

> **Note:** The TMS Derivation job runs the TMS portion of Batch Validation once, not twice as in the full Batch Validation job.

Batch Validation Execution Order

Batch Validation runs TMS processing twice, before and after running all validation and derivation procedures defined in Oracle Clinical for the study.

See "Using a Derived Question as a Parent Question" on page 4-13 for information on how to use this feature.

Batch validation includes the following processes in the following order, for a single study:

1. Sends data from Oracle Clinical to TMS for processing, including Synchronization and Autoclassification. (See "Synchronization" on page 3-36 and "Autoclassification" on page 3-35 for more information.)

2. Returns data from TMS to Oracle Clinical, creating and resolving TMS-related discrepancies in Oracle Clinical.

3. Runs all Oracle Clinical validation procedures, creating and resolving discrepancies.

4. Runs all Oracle Clinical derivation procedures, generating values for derived question responses.

5. Sends a limited set of data—parent question values derived during the current Batch Validation (if any)—to TMS for processing.

6. Returns data specified in the derived parent questions' question sets to Oracle Clinical.

See "OC and TMS Data Exchanged During Batch Validation" on page 4-5 for an explanation of which data is exchanged.

Effect of TMS Errors on Batch Validation

You can control whether Batch Validation fails altogether or just displays a warning if it encounters a serious problem during TMS processing by setting a reference codelist value in Oracle Clinical; see "OCL_STATE" on page 4-22.

If the long value is active and set to Warn for the short value TMS_FAIL_BV_ACT in the OCL_STATE local reference codelist in Oracle Clinical, then if Batch Validation encounters a serious error during its TMS processing, it will give a warning but it will not cause the entire Batch Validation job to fail. Serious errors include:

- a SQL error
- data corruption
- serious setup issue

If the long value is inactive or set to anything other than Warn, Batch Validation will fail if it encounters a serious error during TMS processing.

If Batch Validation encounters a less serious error during TMS processing, it will issue a warning but not fail, regardless of the long value set for TMS_FAIL_BV_ACT. For example, less serious errors include:

- No value set for the local reference codelist TMS_OPTIONS.

Batch validation displays the following error message: "Reference codelist for First Review Flag not found."

- The dictionary specified for a question set used in the study is not included in any TMS domain element for the study in Oracle Clinical.

  Batch validation displays the following error message: "No dict. for Question Set: *question_set_name*."

- The domain specified in the study's TMS domain element is not part of a valid dictionary/domain combination (as defined in the TMS Define Domains window) for one of the following: the dictionary specified in the domain element, or the dictionary specified for a question set used in the study.

  Batch validation displays the following error message: "No Domain for Question Set: *question_set_name*."

Setting Up Data Collection in Oracle Clinical

This section includes:

- Defining and Using Question Sets on page 4-8
- Defining Questions on page 4-11
- Using a Derived Question as a Parent Question on page 4-13
- Associating Child Questions with Question Set Variables on page 4-15
- Associating Questions with Question Groups and DCMs on page 4-15

To use TMS with Oracle Clinical, you must define two types of questions and associate them with DCM question groups:

- A parent question to collect data (called source terms in TMS) at an Oracle Clinical patient visit
- A derived question to receive each piece of information you want to retrieve from TMS

In addition, you must define one or more **question set**s. A question set is a reusable structure that:

- Specifies the TMS dictionary against which to classify the parent question response (TMS verbatim term)
- Includes variables that specify the information you want to retrieve from TMS

During definition of the parent question you:

- Link the parent question to a question set
- Link a derived question to each question set variable

The diagram in Figure 4–1 shows the definitional links between Oracle Clinical and TMS. In the Oracle Clinical Global Library, you define parent and derived questions and questions sets, which specify questions' relation to dictionary data, and link parent and derived questions. In Oracle Clinical studies, you include parent and derived questions in the study definition and link the study to TMS domain/dictionary combinations. In TMS, you define valid dictionary/TMS domain combinations.

Figure 4–1   Relations Between TMS and Oracle Clinical Defined Objects
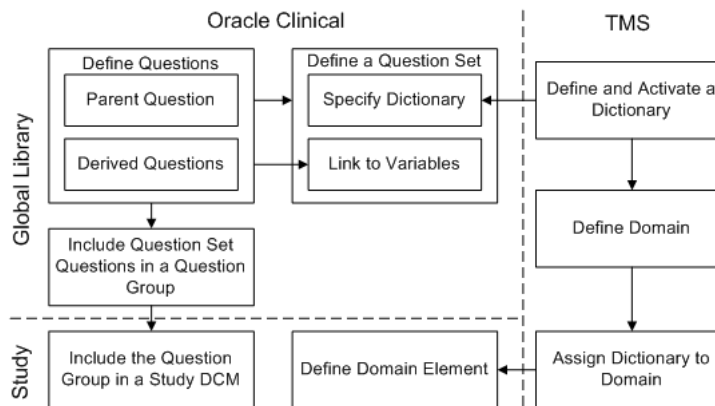
Defining and Using Question Sets

A question set defines the TMS information to be sent to Oracle Clinical in relation to a question response collected in Oracle Clinical. A question set definition has two parts:

- Defining a Question Set. General information about the question set: its name, description, and TMS dictionary against which question responses are to be classified.

- Defining Question Set Variables. Variables that specify the dictionary level and data to be derived (such as term, ID, or code; see "Parameter Name 2" on page 4-10 for a complete list of derivable data).

  Alternatively, question set variables can derive Informative Notes from TMS to Oracle Clinical derived questions.

When you define a parent question (a question whose response is to be to a TMS dictionary term) you assign a question set to the parent question. Then, in the Details section of the parent question definition, you assign a derived question to each question set variable. Thus, the question set is the link between the parent question that collects Oracle Clinical data at a patient visit and the derived questions that contain information from TMS related to the response/verbatim term's classification in TMS.

You can assign the same question set to many different parent questions to collect the same TMS information for each parent question.

> **Note:**   Define the question set completely before assigning it to a parent question. You cannot make changes to the question set after it is assigned to a parent question.

Defining a Question Set

To define the question set:

1. In Oracle Clinical, from the **Glib** menu, select **Question Sets**, then choose **Question Sets**.

2. In the **QS Name** field, enter a unique name assigned to this question set. It can be up to 20 alphanumeric characters.

3. In the **Description** field, enter a description for the question set of up to 60 alphanumeric characters.

4. In the **QS Type Code** field, choose **TMS Q SET** from the list of values. The code entered in this field determines which of the **Parameter 1**…**5** fields must be completed. None are required for TMS.

5. In the **TMS Dictionary** field, from the list of values choose the TMS dictionary to be accessed by this question set. You can use only strong dictionaries.

6. Leave the **Parameter 1**…**5** fields blank. These fields are not used in this context.

7. Set the **Review before TMS?** box. Leave deselected to handle the initial review of thesaurus omissions in TMS.

   To handle the initial review in Oracle Clinical you must select this option and also set the short value FIRST_REVIEW in the local reference codelist TMS_OPTIONS to Yes (see "Managing the TMS/Oracle Clinical Workflow" on page 4-2).

8. Proceed to Defining Question Set Variables to complete the question set.

Defining Question Set Variables

Question set variables link child questions to the parent question and specify the information to be sent to and retrieved from TMS.

When you define a parent question to collect a drug name or medical condition to be classified in TMS, you associate it with a question set. The system populates the parent question's Details window with the question set variables. You must associate each variable with a child question that either sends additional information to TMS or holds information derived from TMS.

You define question set variables in the lower part of the Question Sets window. Each row defines one variable.

1. Under **Question Set Questions**, click in the first available row.

2. Specify a **Sequence Number** for the variable. This number represents the display order you want to appear in parent question's Details window, relative to the other variables in the question set. Sequence numbers can be up to three digits.

3. Enter any value you want, from 1 to 30 alphanumeric characters, in the **Parameter Name** field. This field must be completed and each Parameter Name must be unique within the Question Set, but is not used by the system.

4. In the **Display Prompt** field, enter the text that you want to appear as the variable name in the Details window of the parent question. This field will be mapped to a derived question. Consider using a naming convention to make it easy to decide which question you should associate with which variable; for example, *dictionary level_type of information.* See the possible types of information at Step 8 below. Up to 15 alphanumeric characters.

   > **Note:** The reportable levels of the dictionary specified for this question set are listed in the **Parameter Name 1** list of values.

5. Choose a **Data Type** of CHAR.

6. **Len**. Enter a length for the question set variable. It must be long enough to handle the data that will be returned from TMS. The largest amount allowed is 200. This is the recommended size.

> **Note:** You set the display length for the Data Entry form in the **Len** field in the DCM Questions window.

7. **Question Type**. Select one:

   - **OUTPUT** if the variable is for information derived from TMS, such as a Preferred Term or ATC. Questions defined for this purpose must be defined as derived.

   - **INPUT** if the variable is to send auxiliary information to TMS in a dictionary with nonunique terms in the coding level, such as WHO-Drug Format C. You can use questions to collect auxiliary information such as Country, Pharmaceutical Form, and Strength in order to classify terms to nonunique dictionary terms.

   - **INDICATION** if the variable is to send information to TMS about the indication for which a drug was taken. For example, WHO-Drug supports classifying different occurrences of the same source term to different ATC terms depending on the indication. See "Classifying Indication Omissions" on page 10-10.

8. **Parameter Name 1**. The list of values varies depending on the question type:

   - For INPUT question types, all auxiliary information sublevels defined for the dictionary are listed.

   - For INDICATION question types, there is no list of values. Leave blank.

   - For OUTPUT question types, the list of values includes all dictionary levels and *Info Note*. Select one dictionary level from which to derive the term related to the parent question term. Alternatively, use Informative Notes to send additional information from TMS to Oracle Clinical. See "Defining Informative Note Attributes" on page 7-22 for information about Informative Notes.

> **Note:** Do not enter the name of a group level here. To derive information from a dictionary level that is contained in a group level, you must enter the sublevel here, not the group level.
>
> If you have a Primary Link required to a group level, but a particular term's Primary Link is to any of the sublevels in the group level, you must create a variable for each sublevel.

9. **Parameter Name 2**. The information you enter here depends on whether you chose a dictionary level or Informative Note in the **Parameter Name 1** field and applies only to OUTPUT question types.

   Dictionary Level. If you specify a dictionary level in the **Parameter Name 1** field, TMS returns information about the parent term's related term in that level. In the **Parameter Name 2** field you must specify the type of data you want to derive from TMS.

   For example, if the verbatim term "Headache" is classified to Preferred Term "Head Discomfort," whose related High Level Term is "Neurological Signs and Symptoms NEC," and you specify High Level Term for the dictionary level in **Parameter Name 1** and Term Upper for **Parameter Name 2**; if the question response value was "Headache," TMS would return the value "NEUROLOGICAL

SIGNS AND SYMPTOMS NEC" as the value for the derived question specified in the parent question Details window for this variable.

The list of values displays the following derivable data:

- **CATEGORY**. Optional field used to by some dictionaries to further categorize terms; MedDRA supplies Diagnose, Adverse Event, or Undefined as categories for its terms.

- **COMMENT_TEXT**. User-defined comment entered by the term's creator about the term.

- **DICTIONARY VERSION**. The version of the dictionary, as defined by an Informative Note (see "Defining Informative Note Attributes" on page 7-22).

- **DICT_CONTENT_ALT_CODE**. Optional, user-defined, indexed, unique ID for a term. Designed to provide a link to another thesaurus system.

- **DICT_CONTENT_CODE**. Optional field designed to serve as the primary key within a dictionary. TMS does not require that the dict content code be unique.

- **DICT_CONTENT_ID**. Unique ID across TMS for this term; the primary key in TMS.

- **TERM**. A term as entered or loaded (uppercase, lowercase, or mixed case).

- **TERM_UPPER**. A term, all uppercase.

- **VALUE_1**…**4**. Optional. Data defined by the user for a dictionary level. See "Defining Level Details" on page 6-25.

Dictionary Informative Notes. If you entered "Dictionary Informative Notes" in the **Parameter Name 1** field, in the **Parameter Name 2** field the list of values displays all the derivable Informative Note Attributes defined in this installation.

For example, you can derive an Informative Note of type URL containing a link to information about the dictionary term.

See "Defining Informative Note Attributes" on page 7-22 for information about Informative Notes. For information about the particular Informative Note Attributes that have been defined for this TMS installation, from the **Definition** menu, select **Define Informative Notes Attributes in TMS**.

10. **Info Note?** The system populates this field depending on the setting of the Parameter Name 2.

11. Save.

Defining Questions

This section includes:

- Defining a Parent Question on page 4-12

- Defining a Child Question on page 4-12

Two types of questions are used by TMS: **parent** questions are used to collect responses at patient visits; and **child** questions, derived questions that receive the information specified by question set variables from TMS. Use the standard Oracle Clinical question definition process for both parent and child questions, with the TMS-specific settings below.

Parent questions can also be derived questions in special cases; see "Using a Derived Question as a Parent Question" on page 4-13.

You cannot convert an active, non-TMS question set question for processing in TMS. You must define a new question with a question type of Question Set and the name of its question set as part of the definition. You may want to model your TMS parent questions on your standard Glib questions, changing their names using a convention such as suffix *_TMS*.

> **Note:** A particular derived or parent question can be used only once within a question group: a particular question cannot be used in more than one question set, and can be used only once in a single question set.

Defining a Parent Question

The parent question collects the verbatim term (clinical data, question response) during an Oracle Clinical visit.

To define a parent question and link it to a question set:

1. In Oracle Clinical, from the **Glib** menu, select **Questions**, then choose **Questions**.

2. Complete the definition fields as usual in Oracle Clinical (see *Oracle Clinical Creating a Study* in the Oracle Clinical documentation set), completing these fields as follows:

   - **Question Type**. You must set this field to QUESTION_SET. This value activates the **Question Set Name** field.

   - **Question Data Type**. You must set this field to CHAR.

   - **Question Set Name**. Enter the name of the question set to which you want to link this parent question.

   - **Derived?** Normally you should leave this box deselected. However, there are some cases where you may want a derived parent question; see "Using a Derived Question as a Parent Question" on page 4-13.

   - **Display Prompt**. This text becomes the display prompt in the Data Entry form.

3. Save.

> **Note:** You can either activate the question now by setting the status to A, or leave it provisional so you can change it if necessary. You must activate it and save your work before you can link the question to a question group and use it in a study.

4. Click the **Details** button to associate derived questions with question set variables. You must define the derived questions before you can do this step. See "Defining a Child Question" on page 4-12 and "Associating Child Questions with Question Set Variables" on page 4-15 for instructions.

Defining a Child Question

Create a child derived question for each variable of the question set. Consider using a naming convention to make it easy to remember which question you should associate with which variable; for example, *dictionary level_type of information*.

To define a child derived question:

1. In the Oracle Clinical **Glib** menu, select **Questions**, then choose **Questions**.

2. Complete the definition fields as usual in Oracle Clinical (see *Oracle Clinical Creating a Study* in the Oracle Clinical documentation set), completing these fields as follows:

   - **Question Type**. Set this field to THES VALIDATED.

   - **Data Type**. Set this field to CHAR.

   - **Len** (length) Enter a length at least as long as the question set variable it is being associated with (see below); the maximum setting, 200 characters, is recommended.

   - **Derived**. Select this box.

   - **Default Prompt**. Enter text to become the label for the derived question in the Data Entry form.

3. Save.

> **Note:** You can either activate the question now by setting the status to A, or leave it provisional so you can change it if necessary. You can associate a provisional derived question with a parent question, but you must activate it and save your work before you can link the question to a question group and use it in a study.

Using a Derived Question as a Parent Question

Batch Validation can run its TMS process twice, once before and once after deriving question responses with Oracle Clinical derivation procedures (see "Batch Validation Execution Order" on page 4-6), and allows parent questions to be defined as derived, making it possible to do either of the following during a single Batch Validation job:

- Derive a value to an Oracle Clinical question in TMS, run an Oracle Clinical derivation procedure to take that value and use it to populate the value of a (derived) parent question, and send the derived parent question value to TMS to derive additional Oracle Clinical question set question values.

- Run an Oracle Clinical derivation procedure to populate the value of a (derived) parent question as necessary, and send the derived parent question value to TMS to derive additional Oracle Clinical question set question values.

> **Notes:** Oracle supports only the amount of derivation that can be accomplished within a single Batch Validation. Oracle does not support using an Oracle Clinical derivation procedure to populate the value of another derived parent question with the value of the original derived parent question.
>
> Also, Batch Validation runs the TMS portion twice only if a derived parent question exists and requires processing.

Examples

Deriving a parent question may be useful in the following situations:

- Deriving Data from Two Dictionaries for One Verbatim Term
- Classifying a Term Substituted for the Original Question Response

- Classifying Previously Unclassified Terms

Deriving Data from Two Dictionaries for One Verbatim Term   To classify a single verbatim term to two different dictionaries, and derive information from both dictionaries into Oracle Clinical, you can do so by defining two question sets, as follows:

- The first question set references the first dictionary and includes a (non-derived) parent question to collect a response during data entry and derived child questions to receive TMS values from the first dictionary as specified by the question set variables (a standard Oracle Clinical TMS question set).

- The second question set references the second dictionary and includes a parent question defined as a derived question and derived child questions to receive TMS values from the second dictionary as specified by the question set variables.

To populate the value of the derived parent question, you must write a derivation procedure in Oracle Clinical that propagates the value collected for the first parent question to the second (derived) parent question.

During Batch Validation, the system first derives values from TMS (from the first dictionary) for the first set of derived questions, then processes the derivation procedure that populates the second parent question value, and finally derives values from TMS (from the second dictionary) for the second set of derived questions.

Classifying a Term Substituted for the Original Question Response   If many of the originally collected question responses are inconsistent with current terminology—for example, in a historic study or a study conducted by a company your company has acquired—you can leave the original data and thesaurus derivations, if any, intact but add current information by doing the following:

1. Define a new question for each original question whose responses may be outdated.

2. Examine each response to the original question and manually enter an appropriate value in current terminology for each corresponding new question.

3. Define a new derived parent question also corresponding to each original question whose responses may be outdated. Associate the parent question with a question set and child derived questions to define and receive the related information you want to derive from TMS.

4. Write a derivation procedure that compares the values of the original question and the new non-derived question, and populates the new derived parent question with the value of the new non-derived question if it has a value; otherwise with the original response.

5. Run Batch Validation. TMS processes the value of the new derived parent question and returns the data specified in its question set.

Classifying Previously Unclassified Terms   Add derived parent terms to legacy studies to derive TMS information for question responses that previously had no derived thesaurus information.

1. Define a new derived parent question for each original question for which you want to derive thesaurus values. Associate the parent question with a question set and child questions to define and receive the related information you want to derive from TMS.

2. Write a derivation procedure to populate the value of the new derived parent question with the original question's value.

3. Run Batch Validation. Oracle Clinical runs the derivation procedure to populate the value of the new derived parent question, and TMS returns the data specified in the question set.

Associating Child Questions with Question Set Variables

After you have created a question set, a parent question to collect data, and all the child questions you need for the parent question, you must assign one child question to each variable of the question set associated with the parent question. If it is unclear what information should be associated with each variable, look up the question set definition from the **Glib** menu, by selecting **Question Sets**, and then **Question Sets**.

1. In Oracle Clinical, from the **Glib** menu select **Questions**, then choose **Questions**.

2. Execute a query for the parent question.

3. Highlight the parent question name. This activates the Details button.

4. Press the Details button. The Details window displays the prompts for the variables of the question set linked to this parent question (see "Defining and Using Question Sets" on page 4-8).

5. In the row for each variable, use the LOV to enter the name of the question you created to collect it.

6. Save.

Associating Questions with Question Groups and DCMs

This section includes these special topics:

- Planning Question Groups and DCMs for RDC Special Listings on page 4-15

- Adding Derived Questions to an Ongoing Study on page 4-16

Before you can use parent or derived questions in a study you must associate them with a question group, DCM, DCI, and study as usual in Oracle Clinical (see the *Oracle Clinical Creating a Study* manual).

You do not have to use all the derived questions associated with a parent question in the Global Library. If you choose not to derive all the information defined in the question set, in the DCM question group you can simply not include the derived question mapped to the variable you do not need.

However, the parent question and all its derived questions you do need must be in the same question group.

All questions are visible in the data entry form, but only the parent question is enterable. When values are derived for the derived questions during Batch Validation, they are displayed in the data entry form. You can reference their values as you can any other question response values, in Oracle Clinical data extract and validation procedures.

You can associate the same question set with multiple parent questions in the same DCM.

Planning Question Groups and DCMs for RDC Special Listings

If you are using Oracle Remote Data Capture (RDC) Release 4.5.3.10 or above, you can use Special Listings to display concomitant medications and adverse events by patient. (If you are using TMS to code any other type of data, you can display Special Listings for it too.)

If RDC detects that any question in a DCM is a mapped to a TMS dictionary (is a parent question) it displays an additional item in the drop-down list of actions a user can take in the Home and Casebooks pages. By default, this item's text is: `Review TMS_dictionary_name/DCM_name`. You can substitute other text (such as Adverse Events or ConMeds) for the dictionary name by defining a special type of Informative Note. This text appears on the Special Listings page as well; see "RDC Action Informative Notes" on page 7-27.

When a user selects a patient and selects a dictionary/DCM combination, the Special Listings page displays the patient's responses to all questions that are mapped to that dictionary in that DCM, in every visit and CRF where the DCM has been collected. In addition, the Auxiliary Information field displays the question name and the patient's response for every other question in the question group that is defined as Displayed and not as Derived.

In Figure 4–2, "RDC Special Listings Page" the response to the parent question is displayed in the Verbatim Term column. In the Auxiliary Information column, the system displays all other nonderived, displayed questions in the same question group as the parent question; in this case, Start Date, End Date, and SAE (Yes/No).

To optimize Special Listings behavior, plan Question Groups and DCMs as follows:

- **Question Groups**. Include any questions related to the parent question that you would like to have displayed with it in RDC Special Listings, and do not include any questions you do not want displayed.

- **DCMs**. To enable viewing all adverse event or all concomitant medication data for a patient at the same time, use the same DCM to collect adverse event or conmed data in any DCI/CRF where you need to collect it.

Figure 4–2   RDC Special Listings Page

Adverse Events / AEs for Patient R1001					
Visit					
Name	**Date**	**CRF Name**	**Verbatim Term**	**Auxiliary Information**	
AES	1-Apr-2006	AES	Headache	Start Date: 01-May-2006 | End Dat: -	| SAE: No
AES	1-Apr-2006	AES	Neck pain	Start Date: 06-Jan-2005 | End Dat: 10-Feb-2005	| SAE: No
AES	1-Apr-2006	AES	Psychogenic pain disorder	Start Date: 01-Jun-2006 | End Dat: -	| SAE: No
AES	1-Apr-2006	AECODES	Malignantmast cell tumors	Start Date: 09-Dec-2006 | End Dat: -	| SAE: Yes
AES	1-Apr-2006	AES	Sialectasia	Start Date: 15-May-2006 | End Dat: -	| SAE: No
AES	1-Apr-2006	AECODES	Auriculotemporal syndrome	Start Date: 15-Jun-2006 | End Dat: 29-Sep-2007	| SAE: No

Adding Derived Questions to an Ongoing Study

If you initially do not include all the derived questions associated with a parent question and then decide midway through a study that you do want to derive that information, do the following:

1. Add the derived question to the DCM in the DCM's Question Group Questions window.

2. In Oracle Clinical, from the **Plan** menu, select **TMS Domains**. The Maintain Domain Elements window opens.

3. Select the project to which the DCM's study belongs, and click **Studies**. The Studies window opens.

4. Select the study to which the DCM belongs.

5. Click **Domain Elements**. The Domain Elements window opens.

6. Select the dictionary referenced by the parent question's question set.

7. Click **Force Rederivation**. During the next Batch Validation run, all the parent questions associated with the study will be reprocessed and the associated derived questions repopulated or, in the case of the new question, populated for the first time.

Linking an Oracle Clinical Study to TMS

This section includes:

In TMS, you access dictionaries through a domain. Many domains can have access to the same dictionary. Each domain can have a different combination of dictionaries, different classifications, different company terms, and different rules; see "Creating Domains and Assigning Dictionaries to Domains" on page 6-37. You can specify a virtual dictionary—a dictionary at a specified point in time—for a domain; see "Virtual Dictionaries" on page 6-3.

In Oracle Clinical, you must define a TMS Domain Element. A TMS domain element specifies a TMS dictionary/domain combination and links that combination to an Oracle Clinical project or study. You must define a domain element for each dictionary/TMS domain combination used by a particular study or project.

If all studies within a project use the same TMS domain for a particular dictionary, you can assign a domain element to the whole project. If a project's studies use different TMS domains for a particular dictionary, you must assign a domain element to each study separately.

You can assign a domain element for one dictionary to the project, and domain elements for other dictionaries to the project's studies individually.

You can link a single study to more than one TMS domain and/or more than one TMS dictionary. For example, the following is a valid configuration:

- Study A—Domain A—Dictionary A

- Study A—Domain A—Dictionary B

- Study A—Domain B—Dictionary C

In addition, you can change the rendition of a particular TMS dictionary you are using for a study. For example, if you are coding against a MedDRA base dictionary, you can switch to any virtual dictionary that uses the same base dictionary. Using a virtual dictionary for coding enables you to continue using the same terms and relations even as the base dictionary in your installation upon which it is based is modified or updated.

Translation Derivation

If you have set up two versions of the same dictionary, each using a different language, you can derive a dictionary term from one and the corresponding translated term (with the same dictionary code) from the other dictionary. The dictionaries you use must both be active, and have exactly the same structure, including the same level short names.

To do this, you define TMS domain elements that reference two domain/dictionary combinations: one for each language. In the domain element definition, you must call one dictionary the "Global Language" dictionary and the other the "Local Language" dictionary. Your choice for Global Language Dictionary must match the dictionary specified in question sets used in the study or project.

If you are not setting up translation derivation, enter a Global dictionary and domain only. Leave the Local dictionary and domain information blank.

Rules

The dictionaries you choose must satisfy these criteria:

- Before you establish translation derivation for a TMS domain element, you must link the dictionaries you plan to use. See "Defining a Translation Derivation Link" on page 6-36.

- You can choose two base dictionaries or two virtual dictionaries in a TMS domain element, but not one of each.

- You can choose either the global or local dictionary as the coding dictionary. TMS will derive the translation values from the non-coding dictionary, and rederive these values if they change.

Linking a Project with a TMS Dictionary and Domain

To link a project with a Global Language TMS dictionary and domain, and if necessary, a Local Language TMS dictionary and domain:

1. In Oracle Clinical, from the **Plan** menu, select **TMS Domains**.

2. Highlight the project code you want.

3. Click the **TMS Domain Elements** button.

4. Under the **Global** heading:

    - Select the short name of the Global Language TMS dictionary from the list of values.

    - Select the name of the Global Language TMS domain from the list of values.

5. Save.

If you need to use more than one dictionary or TMS domain in this project, define additional domain elements on the following lines.

Note: You can delete a domain element for a project only if there are no classifications or omissions associated with it.

Linking a Study to a TMS Dictionary and Domain

To link a study with a Global Language TMS dictionary and domain, and if necessary, a Local Language TMS dictionary and domain:

1. In Oracle Clinical, from the **Plan** menu, select **TMS Domains**.

2. Highlight the project containing the study you want.

3. Click the **Studies** button.

4. Highlight the study you want and click the **TMS Domain Elements** button.

5. Under the **Global** heading:

- Select the short name of the Global Language TMS dictionary from the list of values.

- Select the name of the Global Language TMS domain from the list of values.

6. If you are not setting up translation derivation for this study at this site, skip this step.

   Under the **Local** heading:

   - Select the short name of the Local Language TMS dictionary from the list of values, which includes only dictionaries that have Translation Derivation Links to your selected Global Language Dictionary.

   - Select the short name of the Local Language TMS domain from the list of values.

7. To use the Global Language domain/dictionary combination as the coding dictionary for this study at this site, or if you are not setting up translation derivation, leave the **Coding Dictionary?** box selected. If you prefer to use the Local dictionary/domain combination as the coding dictionary, clear this box.

8. Save.

If you need to use more than one dictionary or TMS domain in this study, define additional domain elements on the following lines.

You can also remove a TMS domain element from a study by deleting its row from the TMS Domain Elements for Studies window. When you remove its domain element, the study inherits the domain element for the same dictionary assigned to its project.

> **Note:** You can delete a domain element for a study only if there are no classifications or omissions associated with it.

Force Rederivation The Force Rederivation button invokes a job that tags all records (parent question responses) in the study associated with the selected dictionary for rederivation during the next Batch Validation job. Normally the batch validation job processes only records that are new or changed. Use the Force Rederivation job when you have made structural changes related to TMS in an ongoing study, including:

- Adding derived questions to a question set; see "Adding Derived Questions to an Ongoing Study" on page 4-16

- Using a new dictionary version that has a different level structure from the old version; see "Upgrading to a New Dictionary Version with a New Structure" on page 4-20

- Changing settings in the TMS_CONFIGURATION reference codelist, including CREATEGLOBALVTA, GLOVTAAPPRREQD, DOMVTAAPPRREQD, NONAPPRVTAUSED, ALLOWGLOVTAS, and AUTOGLOVTAS; see "Changing TMS_CONFIGURATION Settings After Data Has Been Processed" on page 3-30

Using a New Version of a Dictionary for an Ongoing Study

After you link a project or study to an external dictionary, the dictionary vendor may issue a new version of the dictionary; see Chapter 8, "Upgrading to a New Dictionary Version" for information on TMS upgrade features.

Upgrading to a New Dictionary Version with the Same Structure

If the new dictionary version does not involve any changes to the dictionary's structure, you do not need to make any changes to your project or study definitions in Oracle Clinical.

Upgrading to a New Dictionary Version with a New Structure

If the new version does include changes to the dictionary structure—such as adding or removing a level, changing a level's name, or changing the relations between the levels—and you want to use the new version in an ongoing study, you must redefine the new dictionary version as if it were a different dictionary (see Chapter 6, "Defining and Loading Dictionaries"), relink your project or study, and redefine the questions and question sets used:

1. For each question set used in each study, define a new question set referencing the dictionary version with the new structure.

2. For each parent and derived question used in each study, create a new question. Add a question set referencing the new dictionary version to each parent question and associate new derived questions with the question set variables.

3. Create a new TMS Domain Element with the new dictionary version.

4. Add each new question to the same study question group(s) as the question it is replacing and mark the old questions as not collected.

5. For the old questions, deselect **Collect in Study** and **Collect in Subset**. For new questions, select **Collect in Study** and, if you want the question to be displayed in data entry screens, select **Collect in Subset**.

6. Edit the DCM layout and move the layout to production.

7. Use Mass Changes to copy existing responses for the old parent question to the new parent question.

8. From the TMS Domain Elements window, select the study, navigate to the TMS Domain Elements for Study window, and click **Force Rederivation**.

> **Note:** If you prefer, you can perform this step from the TMS API instead. The procedure ocl_tms_pack.SetLastTMSDerivTS checks for changes in a study's responses since a date that you specify, then marks those changed responses for rederivation. See the *Oracle Thesaurus Management System Technical Reference Manual* for specific parameters and instructions.

9. Run Batch Validation for the study.

After you complete these steps, the Autoclassification process uses the new dictionary when trying to classify verbatim terms.

Accessing Oracle Clinical Data in TMS

Oracle Clinical sends information to TMS about the origin of each verbatim term occurrence. This information is stored in the TMS database, allowing you to query on Oracle Clinical-specific information within TMS. This external system information is available in the following TMS windows:

- Classify VT Omissions

- Approve VTAs

- Approve Action Assignments

- Browse VT Classification Data

- High-Level Reclassification

- Reclassify Verbatim Terms

The information includes the following database columns: **Project**, **Study**, **DCM**, **Document Number**, **Visit**, **Patient**, **Discrepancy ID**, and **Investigator**. In addition, you can choose to make details available for any of the above columns. This requires building an Oracle Clinical view and granting TMS users access privileges to the view. See "Define Column Details" on page 5-32 for more information.

Setting Integration Reference Codelist Values

You must set values in four reference codelists in Oracle Clinical that determine aspects of the TMS-Oracle Clinical interaction:

- OCL_OPTIONS_TYPE_CODE

- TMS_OPTIONS

- DISCREPANCY REV STATUS CODE

- OCL_STATE

OCL_OPTIONS_TYPE_CODE

When TMS and Oracle Clinical are installed together, Oracle Clinical takes care of TMS's reference codelist replication.

The OCL_OPTIONS_TYPE_CODE installation reference codelist contains entries for optional subsystems that may be available to Oracle Clinical. To use TMS with Oracle Clinical, the short value TMS_INSTALLED must be set to Y. This value is set by a script during installation.

- The value should remain set to Y whether or not you have a distributed environment.

- The short value SR_INSTALLED must be set to Y you have a distributed environment. This information is covered in the *Oracle Clinical Administrator's Guide*.

TMS_OPTIONS

This local reference codelist contains TMS-specific options. Currently the only option defined is FIRST_REVIEW. If the Long Value for this option is set to Y, and the **Review Before TMS** setting is also set to Y in the question set, then the First Review for a thesaurus omission happens in the Oracle Clinical Discrepancy Management System, rather than in TMS. For more information on the interaction between the Oracle Clinical Discrepancy Management System and TMS, see "Setting First Review" on page 4-4.

DISCREPANCY REV STATUS CODE

This installation reference codelist provides the codes used by the Oracle Clinical Discrepancy Management System. Three are used in processing TMS omissions:

- TMS EVALUATION

- UNREVIEWED

- TMS IN PROGRESS

UNREVIEWED is active by default, but you must activate the two TMS-specific values before you begin using TMS.

For more information on the interaction between the Oracle Clinical Discrepancy Management System and TMS, see "Managing the TMS/Oracle Clinical Workflow" on page 4-2.

OCL_STATE

This local reference codelist includes one TMS-related short value: TMS_FAIL_BV_ ACT. If it is active and its Long Value is set to Warn, then if Batch Validation encounters a serious error during its TMS processing, it will give a warning but will not cause the entire Batch Validation job to fail.

If the Long Value is inactive or set to anything other than Warn, Batch Validation will fail if it encounters a serious error during TMS processing.

If Batch Validation encounters a less serious error during TMS processing, it will issue a warning but not fail, regardless of the long value set for TMS_FAIL_BV_ACT.

See "Effect of TMS Errors on Batch Validation" on page 4-6 for more information.

5

Integrating TMS with Other Systems

This section includes:

Oracle Thesaurus Management System (TMS) is designed for flexibility in integrating with other clinical data systems. The integration mechanisms for some other Oracle Health Sciences products are predefined, but you can create your own in order to integrate TMS fully or partially with other systems. All TMS functions are performed by packages that you can call from an external system without directly manipulating the TMS database.

If you have multiple source data system sites, you can install TMS at each site and coordinate classifications for the whole system using replication.

To integrate any external system with TMS, you must enter information about the system in TMS.

Using TMS with Other Oracle Health Sciences Applications

TMS is designed for full integration with other Oracle Health Sciences applications including Oracle Clinical, Remote Data Capture (RDC) and the Adverse Event Reporting System (AERS).

This section includes:

Using TMS with Oracle Clinical

All the data structures and data exchange processes needed for full integration between TMS and Oracle Clinical are predefined. TMS stores contextual information about each source term—the Patient ID, Document Number, and more—and Oracle Clinical stores whatever information you specify to derive from TMS—for example, the classification-level term to which a source term is classified, an Informative Note associated with the term, and related terms in higher dictionary levels. If you reclassify a term in TMS, Oracle Clinical automatically rederives the information for each affected source term (Oracle Clinical question response).

For more information about using TMS with Oracle Clinical, see Chapter 4, "Integrating TMS with Oracle Clinical."

Using TMS with the Remote Data Capture Option

If TMS is integrated with Oracle Clinical supporting Remote Data Capture, RDC users can submit verbatim terms immediately to TMS by executing the RDC option Validate Patient; see the *Oracle Clinical Remote Data Capture Onsite User's Guide* for details. If TMS can autocode a term, it immediately returns the appropriate derivations to RDC. If TMS cannot autocode a term, the term remains in TMS as an omission and its derivations are returned to RDC after it is manually coded.

You can set up the RDC Special Listings page to view adverse events or concomitant medications or any other category of data that uses TMS for coding; see "RDC Action Informative Notes" on page 7-27.

Using TMS with AERS

Oracle Adverse Event Reporting System (AERS) uses TMS internally as its dictionary coding system for adverse events and medications. You can do TMS coding interactively from the AERS user interface.

Instructions for installing TMS with AERS are included in the *Oracle Adverse Event Reporting System Installation Guide*. Instructions for coding terms in TMS through AERS are included in the *Oracle Adverse Event Reporting System User's Guide*.

Integrating TMS with Non-Oracle Systems

You can integrate TMS fully or partially with non-Oracle source data systems, but you must handle the data exchange. In the case of full integration, some customization of the external system is also required.

Full Integration

Full integration requires the installation of TMS objects in the external system, a level of integration that is recommended if the external system is running in a stable global Oracle environment. In full integration, TMS maintains external data in both the tms_source_terms table and the tms_vt_omissions table.

A fully integrated system benefits from the full range of TMS functionality. It feeds source terms to TMS with contextual data you specify (such as Document Number) so that if you reclassify or declassify a term in TMS, TMS can send information about each affected source term back to the source data system.

In TMS, you run Autoclassification, manually classify remaining terms (omissions), assign Actions, and reclassify or declassify as necessary. You specify the information you want to derive from TMS for each source term, and TMS sends that data to the source data system associated with each source term.

To fully integrate a non-Oracle system with TMS, you must devise ways to:

- Associate the source term collection unit—a study or case, for example, depending on the source data system—with TMS domains and dictionaries (the X Area is the source term collection unit in both the tms_source_terms table and the tms_vt_omissions table)

- Define objects to receive values derived from TMS

- Integrate TMS with the external system's discrepancy management function

- Exchange data between the two systems

> **Note:** When calling the TMS autocode API from an integrated external system, process terms in dictionary and domain order for best performance.

It may be helpful to read about the way these issues are handled for integration with Oracle Clinical as an example; see Chapter 4, "Integrating TMS with Oracle Clinical."

In addition, to set up full integration with an external system you must define the external system in TMS; see "Defining External System Information in TMS" on page 5-28.

Partial Integration

If full integration is not desirable—for example, if space is an issue—you may want to set up TMS with only partial integration to your external source data system. In partial integration, TMS maintains external data in the tms_vt_omissions table only. It does not store any data in the tms_source_terms table.

The external system feeds data to TMS, and TMS returns derived data and omissions. You can classify terms manually and perform all other functions within TMS. However, because the tms_source_terms table is not used, TMS has no external system information (such as document number) associated with source terms and therefore cannot associate derived terms with source terms after the initial classification.

The external system must handle all the data exchange issues and the impact of reclassification and declassification on previously derived data.

- If verbatim terms are reclassified or declassified in TMS after their initial classification, TMS can return this information to the external system but cannot associate it with the original data in that system.

- If source data changes after a data entry update in the external system, TMS can process the new data, but the external system must associate the classifications with the correct original data.

All TMS functions can be performed by API packages called from outside TMS. TMS packages and tables are documented in the *Oracle Thesaurus Management System Technical Reference Manual*. See "The API" on page 1-4 for a sampling of the packages available.

Using TMS in a Distributed Environment

Regardless of whether you use TMS with Oracle Clinical or a different clinical data management system (CDMS), if you maintain multiple sites of your CDMS, you can install an instance of TMS at each site and coordinate dictionary coding among all the sites.

You must install one TMS instance as the master instance and the others as local instances (see the *Oracle Thesaurus Management System Installation Guide*). You then use symmetric replication to coordinate classifications among all the sites (see "Managing Distributed Locations" on page 3-32).

Using TMS with Disconnected Systems

This section includes:

- [Setting Up Disconnected System Integration](#) on page 5-5
- [Using DSI from the Source Site](#) on page 5-14
- [Using DSI from the Sponsor Site](#) on page 5-16
- [Viewing XML File and Record Statuses](#) on page 5-20
- [Viewing Error Files](#) on page 5-23
- [DSI Batch Jobs](#) on page 5-24

The TMS DSI feature provides a customized XML-based batch data load process that enables you to use TMS as a central dictionary classification tool for systems located remotely and operated by different businesses or research organizations.

DSI is designed specifically to allow a pharmaceutical company sponsoring a clinical trial that is being conducted by multiple contract research organizations (CROs), to use TMS as a central classification tool for terminology quality control. In this documentation the central TMS instance is called the **sponsor site** and the remote classification system instance is called the **source site**.

CROs send their source data to the sponsor, with contextual information (such as patient ID and document number) and preliminary classifications. The import process at the sponsor site compares the source site's classifications to the sponsor's and loads data in the sponsor database and/or creates warnings or errors associated with faulty data. When the sponsor sends the data back to the source site, the CRO's import process overwrites CRO classifications with those of the sponsor, if there are conflicts, and passes on any Actions created at the sponsor site for particular terms. The import error file shows records that have been overwritten.

The sponsor TMS instance stores all CRO source terms and contextual information, and runs TMS Synchronization on source data (this is automatic during Oracle Clinical Batch Validation).

The sponsor must use TMS. The source site must use either TMS or another dictionary coding system that is capable of processing XML files in the required DSI structure. This section is written for source sites using TMS as their dictionary coding system. For information on the required XML file structure, see Appendix B, "Disconnected System Integration XML Files." Appendix B also includes the section "Validation of Non-TMS XML Files" on page B-5 for source sites that do not use TMS.

The external source data system feeding data into TMS can be Oracle Clinical, the Oracle AERS, or a third-party system.

A sponsor can use DSI with any number of source sites. Source sites can use DSI with multiple sponsors, but they must maintain a different database for each sponsor.

The sponsor and source site must synchronize the contents of the verbatim term and classification levels of each dictionary used in a study using DSI.

You can use DSI with databases that support either UTF-8 or Western European (WE) character sets, but both the sponsor and the source site must support the same character set.

Limitations   Disconnected System Integration has the following limitations:

- High-level reclassifications cannot be performed on imported data in the sponsor site. However, high-level reclassifications can be performed at the source site.
- The sponsor instance does not have a UI drill-down capability for source site data.
- If the sponsor is using Oracle Clinical, OC Batch Validation does not run on source site data.

DSI Workflow   The basic DSI workflow follows:

- A source site loads the version of one or more dictionaries specified by the sponsor into TMS or another dictionary coding system.

- The source site may or may not classify its source terms against these dictionaries.

  If the source site is using TMS, it processes terms from the source data system as usual, loading data into TMS and running Autoclassification (with Oracle Clinical as the source data system, Autoclassification runs during Batch Validation). The site can then either send any remaining unclassified terms back to the sponsor as omissions or manually classify them.

- At predefined time intervals, the source site exports newly collected source terms with their contextual information (such as patient ID and document number) and classifications to an XML file with a predefined structure, and sends the file to the sponsor company. If there are source term deletions, they are also included in the file.

- The sponsor site imports the file, compares source site VTAs to its own VTAs, and runs TMS Autoclassification on the omissions in the XML file. If there are any conflicting classifications or other errors, the import process produces error file warnings or errors associated with the appropriate records. The sponsor sets reference codelist values to determine how the import process handles particular problems.

- Sponsor personnel use TMS to manually classify CRO source terms as necessary; creating and approving VTAs, making reclassifications and requesting Actions, according to sponsor standards.

- The sponsor extracts an XML file containing updated classifications, omissions, and Actions associated with the original source terms and their contextual information, and sends it to the source site.

- The source site synchronizes classifications and Actions with the data from the sponsor. If the source site uses TMS, the DSI import process automatically overwrites source site classifications (VTAs) with those of the sponsor's when they conflict.

- The source site sends derived TMS information to its source data system, associated with the appropriate source terms. If the source site uses Oracle Clinical, new Actions and derivations appear in Oracle Clinical after the next Batch Validation.

Both the source site and the sponsor site maintain a full audit trail of each export and import. DSI supports both full and incremental data processing.

Setting Up Disconnected System Integration

To use DSI, you must set it up at both the sponsor and the source sites. Some setup tasks are the same at both sites; others are different. In addition, you must manually ensure that both sites are using the same version of each dictionary for each study.

This section includes:

- Setting Up the Sponsor Instance on page 5-6
- Setting Up the Source Instance on page 5-6
- Creating Directories on the Database Server on page 5-7
- Running the DSI Initialization Job on page 5-7

- Register Remote Databases on page 5-8
- Defining Disconnected System Integrations on page 5-8
- Defining X Areas on page 5-9
- Defining XML Tag Mappings on page 5-10
- Setting DSI Preferences with Reference Codelist Settings on page 5-11
- Synchronizing Dictionary Contents Between Sites on page 5-13
- Granting User Privileges on page 5-13

Setting Up the Sponsor Instance

At the sponsor site you must do the following:

- Create operating system file directories for temporary data and error file storage; see "Creating Directories on the Database Server" on page 5-7.

- Run the Initialize Disconnected System Integration Systems job with DSI Instance Type set to Sponsor; see "Running the DSI Initialization Job" on page 5-7.

- Register each remote database by running the job; from the **Definitions** menu, select **Jobs**, then choose **Register Remote DSI Databases**; see "Register Remote Databases" on page 5-8.

- Enter information about the external source data system(s) (such as Oracle Clinical) used by each source site. You need only define any particular external system once; for example, if you have five source sites using Oracle Clinical, you need only define Oracle Clinical once. If you have already defined Oracle Clinical as an external system because your own organization is using it with TMS, you do not need to define it again. See "Defining External System Information in TMS" on page 5-28.

- Define each DSI for each remote database/external system combination; see "Defining Disconnected System Integrations" on page 5-8.

- Map XML tag metadata so that the system can interpret the data received from the source site and send the source site data it can interpret; see "Defining XML Tag Mappings" on page 5-10.

- Set reference codelist values; see "Setting DSI Preferences with Reference Codelist Settings" on page 5-11.

- Grant privileges to users who will run the DSI import and export processes; see "Granting User Privileges" on page 5-13.

- Ensure that the source site has the same version of the dictionary or dictionaries that you are using; see "Synchronizing Dictionary Contents Between Sites" on page 5-13.

- Ensure that any source site not using TMS knows how to validate XML files against the TMS XML schema before sending them to you; see "Validation of Non-TMS XML Files" on page B-5.

Setting Up the Source Instance

At the source site you must do the following:

- Create operating system file directories for temporary data and error file storage; see "Creating Directories on the Database Server" on page 5-7.

- Enter information about the external source data system(s) (such as Oracle Clinical) whose data you are sending to the sponsor via DSI. If you have already set up the external system for integration with TMS, you do not need to define it again. See "Defining External System Information in TMS" on page 5-28.

- Run the Initialize Disconnected System Integration Systems job with DSI Instance Type set to Source; see "Running the DSI Initialization Job" on page 5-7.

- Define Disconnected System Integration with the sponsor; see "Defining Disconnected System Integrations" on page 5-8.

> **Note:** You must define the external source data system, initialize DSIs, and register the sponsor's database before you can do this step.

- Define X Areas to specify discrete data collection and processing units (such as studies); see "Defining X Areas" on page 5-9.

- Set reference codelist values; see "Setting DSI Preferences with Reference Codelist Settings" on page 5-11.

- Grant privileges to users who will run the DSI import and export processes; see "Granting User Privileges" on page 5-13.

- Ensure that the dictionaries you are using are synchronized with the sponsor's dictionaries; see "Synchronizing Dictionary Contents Between Sites" on page 5-13.

Creating Directories on the Database Server

You must create three operating system directories on the database server to temporarily hold XML files:

- Input directory. The import process operates on files sent from the remote system to the Input directory.

- Output directory. The export process creates one or more XML files in the Output directory.

- Error directory. Both the import and export processes write an error file to the Error directory if they process any data with errors or warnings.

You must grant Delete privileges on the Input directory to the Oracle User Database Process.

Defining the External Source Data System

You must define each external source data system (such as Oracle Clinical) as an external system in TMS. You need to define each external system only once. If you have already defined an external system because you are using it with TMS, you do not need to define it again.

At a sponsor site, if you have multiple source sites using a particular external source data system, enter information about that system only once.

See "Defining External System Information in TMS" on page 5-28.

Running the DSI Initialization Job

Before you can export or import data you must initialize your site as either a sponsor or source site. To run the job:

1. Select **Definition**, then **Jobs**, then **Initialize Disconnected System Integrations**. The Initialize Disconnected System Integrations job pop-up window appears.

2. For the job-specific parameter **DSI Instance Type**, select either **Source** or **Sponsor**.

3. If necessary, enter the name of your report server; see "Running a Job" on page 2-16.

4. Submit the job. See "Running a Job" on page 2-16.

> **Note:** If you used the DSI feature before installing TMS Release 4.6 and have not yet done so, you must declare your TMS instance to be either a sponsor or a source site by running the DSI initialization job. If you need to act as a sponsor for some studies and a source for other studies, you must set up a different database for each role.
>
> The initialization job will fail if:
>
> - you try to initialize as a sponsor but any of your DSI definitions includes your own database as the Instance name.
>
> - you try to initialize as a source but any of your DSI definitions includes a database other than your own as the Instance name.
>
> If you have this situation, first delete the X Areas associated with the incorrect instance and then delete the DSI definition that includes the incorrect instance.

Register Remote Databases

At the sponsor site only, register the database used by each source site for DSI by running the job; from the **Definition** menu, select **Jobs**, then choose **Register Remote DSI Databases**. You must register remote databases one at a time.

> **Note:** Source sites can use DSI with multiple sponsors, but must set up a different database for each sponsor.

The job takes the following parameters:

- **Instance**. Enter the source site database name.

- **External System**. From the list, choose the external source data system (such as Oracle Clinical) whose data will be sent to the remote system via DSI.

> **Note:** You must define the external source data system before you can register the remote database. See "Defining the External Source Data System" on page 5-7.

- **Report Server Name**. Enter the name of the report server that will run the job.

Defining Disconnected System Integrations

From the **Definition** menu, select **Define Disconnected System Integrations**, then the **Definitions** tab.

> **Note:** Before defining Disconnected System Integrations here, you must define the external source data system, register the remote database used by each partner, and create directories on the database server. See "Defining the External Source Data System" on page 5-7, "Register Remote Databases" on page 5-8, and "Creating Directories on the Database Server" on page 5-7.

Set up Disconnected System Integration by entering information on both sponsor and source sites:

1. Enter information about each external source data system (such as Oracle Clinical) that the site is using:

   - **System**. From the list of values, choose the correct external system. The list of values includes all systems defined as external systems in TMS.

   - **Instance**. From the list of values, choose the correct database location of the external system. The list of values includes all registered remote databases.

2. **Remote Partner.** Enter the name of the organization that owns the remote database, or other information to help you identify the Disconnected System Integration. The system does not use this field.

3. Enter the names, with the full path, of the Input, Output, and Error directories you created on the database server (see "Creating Directories on the Database Server" on page 5-7). For example:

   /root/app/oracle/admin/*DSI_database_name*/xmlworkdir/*partnerx_database_input*

   where *xmlworkdir* is your DSI directory, and *partnerx_database_input* is your input directory for XML files from partner X's remote database.

   > **Note:** When you enter this information, the system verifies that the locations are correct. However, if you later change the directories on the server, any export or import job that uses the directory will fail.

4. Save.

Defining X Areas

From the **Definition** menu, select **Define Disconnected Integration Systems**, then the **X Areas** tab.

Define X Areas on the source site only. The first time the XML Import process on the sponsor site encounters data from a particular X Area, it automatically creates a corresponding X Area on the sponsor site.

An X Area is a data collection unit. If you are using Oracle Clinical as your external source data system, X Areas correspond to studies, and the X Area ID is the same as the Study ID. DSI limits the data in any one data exchange to data from a single X Area; by defining an X Area for each study, for example, you ensure that data from only one study is loaded and processed at a time.

During the export/import process, each X Area being processed is locked on both systems, so that data from a particular X Area cannot be exported and imported at the same time. TMS uses the same locks that Oracle Clinical Batch Validation uses;

therefore DSI export/import cannot occur at the same time as Oracle Clinical Batch Validation.

You can then change the X Area status at the sponsor site as well as the source site (see Step 3 below).

To define an X Area:

1. In the Define Disconnected System Integrations window's **Definitions** tab, query for the external system and instance combination for which you need to define an X Area.

2. Click the **X Areas** tab. The X Areas window opens.

3. Insert a row by selecting **Record**, then **Insert**.

4. In the X Area column, enter the ID used by the external source data system for the data collection unit. If Oracle Clinical is the external source data system, enter the Study ID.

5. In the X Area Status column, the status of a new X Area is automatically set to Active. You can change the status later, when appropriate. The possible statuses are:

   - Active. X Areas are automatically set to Active when they are first created. If data from an inactive X Area is sent to either system, the X Area status is automatically changed back to Active.

   - Inactive. Set the X Area's status to Inactive when you no longer expect any data changes. If data is received for the X Area, the import job processes the data, sets the X Area status to Active, and gives a warning.

   - Complete. When you no longer need to use DSI for a particular X Area, you can set the X Area's status to Complete. If data is sent to either system for an X Area with a status of Complete, the import job does not process the data and returns an error.

6. Click in the **X Area Name** field to display the list of values. The system displays the X Areas defined in the sponsor system and database you specified in the Definitions tab.

7. Select an X Area Name.

8. Enter a description of the X Area in the Description column.

9. Save.

Defining XML Tag Mappings

(Sponsor site only.) From the **Definition** menu, select **Define Disconnected Integration Systems**, then the **Tag Mappings** tab.

XML files exported from both sponsor and source sites use the same tags, but different sites may use different tag values for the same entities. For example, for a particular study a sponsor might use the dictionary MedDRA in a domain called ALLCODING, while a source site might used MedDRA in a domain named *SPONSOR_NAME* for the same study.

Use this window on the sponsor site only to map the values used by each source site to the names used in your system. The sponsor system uses these mappings to interpret data sent by the source site and to send data to the source site with the names used by the source site.

The system translates the instance and X Area names provided in the XML file to IDs.

For information on the required structure, tags, and generated names of DSI XML files, see Appendix B, "Disconnected System Integration XML Files."

For each mapping:

1. Click in the first empty row. The fields become enterable.

2. **XML Tag**. From the list, select the metadata to map. The choices are:

   - Dictionary Short Name

   - Domain Name

3. **Remote System Value**. Enter the value used at the source site for a dictionary short name or domain name (whichever you selected in the **XML Tag** field).

   > **Note:** If you are mapping the dictionary short name, and either site is using a virtual dictionary, enter the short name of the associated base dictionary, not the virtual dictionary.
   >
   > If you are mapping the domain name, use the virtual domain name if there is one.

4. **Central TMS Value**. From the list, select the value used at the sponsor site for the same dictionary or domain whose source site name is listed in the XML Value column. See Table 5–1, " Tag Mapping" for details.

5. Save.

Table 5–1   Tag Mapping

XML Tag	Remote System Value	Central TMS Value
DICTIONARY_SHORT_NAME	Source site short name for Dictionary X	Sponsor short name for Dictionary X
DOMAIN_NAME	Source site name for Domain Y	Sponsor name for Domain Y

> **Note:** If the domain name in either system contains a special character such as (& @ or *) you will not be able to use DSI. You will need to change the domain name.

Setting DSI Preferences with Reference Codelist Settings

The local reference codelist TMS_DSI contains parameters whose settings determine important aspects of DSI behavior. The parameters are: OUTSYNCCSP, CREVTA, GLOBALVTA, and XAPPVTA. The parameters and their settings are described below.

To change the settings for TMS_DSI parameters:

1. From the **Definition** menu, select **Local Reference Codelists**.

2. In the **Name** field, query for TMS_DSI.

3. For each parameter, enter a single-letter setting in the **Long Value** field and select the **Active** box. (Selecting **Default** has no effect.)

4. Save.

OUTSYNCCSP The DSI import process checks dictionary terms required for new VTAs created at the exporting site against dictionary terms on the verbatim term or classification level at the importing site. If a dictionary term used in a new VTA is not included at the importing site, and the dictionary versions were properly synchronized when DSI was set up, any such difference shows that a dictionary term has been added at the exporting site. The value of the OUTSYNCCSP setting determines the way the import process handles out-of-sync dictionary data.

- **Fatal Error (E)**. If the import process finds a new VTA whose dictionary term is not present at the importing site:

  - The process ends with a Fatal Error status.

  - Information about the missing dictionary term is included in the error file.

  - Any records that were successfully processed (before the record with an error) remain in the database, but also appear in the error file.

- **Warning (W)**. If the import process finds a new VTA whose dictionary term is not present in the importing system:

  - The import job imports source data records as omissions.

  - In the error file, it associates a warning with the VTA record whose dictionary term is missing.

  - On the sponsor site only, the import job creates a Discrepancy Message associated with an omission for the source term whose VTA's dictionary term is missing in the importing system. This Action is visible in the importing system's Classify VT Omissions window. It is not sent to the external source data system, as other Actions are. The Action remains associated with the omission until it is resolved.

- **Neither (N)**. If the import process finds a new VTA whose dictionary term is not present in the importing system:

  - The import job imports source data records as omissions.

  - It creates a Discrepancy Message associated with an omission for the source term whose VTA's dictionary term is missing in the importing system. This Action is visible in the importing system's Classify VT Omissions window. It is not sent to the external source data system, as other Actions are. The Action remains associated with the Omission until it is resolved.

  > **Note:** If you choose the setting N, the system does not include these dictionary synchronization errors in the error file. They are visible only in the importing system's Classify Omissions window, where you can query for them in the **Action Text** field with the string `%out of sync%`.

The default value (and recommended setting) is E.

CREVTA (Sponsor site only.) Determines whether the sponsor site imports new VTAs from the source site approved, unapproved, or both. DSI does not use the value entered for this parameter at the source site.

- Approved (A). All VTAs created at the source site are imported as Approved VTAs, regardless of whether they were approved at the source site or not. This setting is not recommended.

- Unapproved (U). All VTAs created at the source site are imported as Unapproved VTAs, regardless of whether they were approved at the source site or not.

- XML (X). VTAs created at the source site are imported with the same approval status—Approved or Unapproved—given by the source site, as reflected in the XML file.

The default value is X.

GLOBALVTA   Determines whether Global VTAs are accepted by the importing system.

- Allow Global VTA (A). Global and Domain VTAs are imported without warnings, if they have no errors.

- Fail Global VTA (F). Global VTAs are not accepted for import. The system creates an error for the record.

- Domain Only (D). Global VTAs are not accepted for import. The system creates a warning for the record and a Domain VTA instead of a Global one.

The default value is A.

XAPPVTA   Enforces whether or not export files must contain only Approved VTAs or can contain Unapproved VTAs as well.

- Yes (Y). Export files must contain only Approved VTAs. Unapproved VTAs are not included in the file. Only when a VTA has been approved is it exported.

- No (N). Export files can contain Approved and Unapproved VTAs.

The default value is N.

Synchronizing Dictionary Contents Between Sites

The source site must use the same version of each dictionary that the sponsor uses for the same study, and dictionary data (including VTA subtypes Accepted and Misspelled) at the two sites must be identical at the verbatim term and classification levels.

When you set up DSI, you must ensure that the sponsor and source site use the same dictionary data; for example, the sponsor can provide dictionary data to load onto the source site. If dictionary synchronization errors do occur, you must correct them.

The DSI import process checks that the coding levels—the verbatim term and classification levels—on the importing site contain all the terms used on the other site. The import process does not compare all terms on the dictionary coding levels of the two dictionaries; it checks only those dictionary terms used in VTAs being processed. You can use the OUTSYNCCSP reference codelist setting to specify what you want the import process to do if the importing site is missing dictionary terms used by the other site. See OUTSYNCCSP on page 5-12.

> **Note:**   If the two systems are using virtual dictionaries, the cut-off dates will probably be different. For example, if the sponsor is using a virtual dictionary and then loads its virtual dictionary onto a source site, the cut-off date on the source site will be later.

Granting User Privileges

You must grant the role TMS_DSI_PRIV to all users who should be able to see the Maintain DSI Files window and run the incremental export and import jobs, all of

which are located in the **VTA Maintenance** menu path, under **Disconnected System Integration**.

You must grant the role TMS_DEFINE_PRIV to all users who should be able to see the Define Disconnected System Integrations window, run the **Force Rederivation** job from that window, or to run the Register Remote Databases job. This is the same role that allows access to all the other tasks in the **Definition** menu path.

For information on granting user privileges, see "Assigning Roles to a User" on page 3-7.

Using DSI from the Source Site

This section includes:

- Processing Source Data on page 5-14

- Exporting Data on page 5-14

- Sending the XML File to the Sponsor Site on page 5-15

- Importing Data on page 5-15

- Updating External System Data on page 5-16

Processing Source Data

At the source site, you process terms from the source data system as usual in TMS, loading data into TMS and running Autoclassification. (If you are using Oracle Clinical, this happens automatically during Batch Validation.) The sponsor may or may not want you to manually classify omissions and assign Actions.

Exporting Data

At the source site, you use a batch job to extract source terms and any source term deletions and classifications (VTAs) from your dictionary coding system to an XML file in a predefined format (see Appendix B, "Disconnected System Integration XML Files" for more information).

On a source site, the export job extracts the following types of data:

- All verbatim term omissions (VTOs) with a timestamp greater than the last export's timestamp

- All source terms with a timestamp greater than the last export's timestamp

- For each source term, contextual information (such as patient ID and document number) and any classifications (VTAs) made at the source site

- New and changed Informative Notes of type Workflow associated with verbatim terms

- Source data deletions, if any

- All VTAs created locally for source data from the selected X Area since the last export of the same X Area

> **Note:** If the local reference codelist TMS_DSI parameter XAPPVTA is set to Y, the system exports only Approved VTAs. Nonapproved VTAs are not exported. If set to N, both Approved and Nonapproved VTAs are exported.

Sending the XML File to the Sponsor Site

DSI does not handle transporting the XML file to the sponsor site. You must do it manually; for example, zipping the file and sending it via either FTP or email.

To avoid sending the same files to the sponsor for unnecessary reprocessing, manually delete files from the output directory after sending them to the other site.

Importing Data

After the sponsor processes your data—using Autoclassification and any necessary manual classifications, reclassifications, declassifications, or Action assignments—the sponsor sends an XML file with the updated information to your DSI input directory (see "Import DSI Data" on page 5-25).

You must run the import job; from the **VTA Maintenance** menu, select **Disconnected System Integration**, then choose **Import DSI Data**.

You can see the status of exports and imports; from the **VTA Maintenance** menu, select **Disconnected System Integration**, then **Maintain DSI Files**. For instructions, see "Viewing XML File and Record Statuses" on page 5-20.

This section describes how the import process handles the following situations: Classification Conflicts, Global VTA Conflicts, Source Term Actions, and Missing Dictionary Terms.

Classification Conflicts The import process compares the VTAs in the XML file with those for the same verbatim terms in your database. If a verbatim term is differently classified by the sponsor, the import process automatically overwrites your VTA with the sponsor's VTA. A warning is associated with the record in the error file so that you can see which terms the sponsor has reclassified.

If the sponsor declassifies a VTA you created, the omission appears in your system. Declassifications occur only if you create a manual classification and the sponsor declassifies the verbatim term without reclassifying it.

Global VTA Conflicts If the sponsor sends Global VTAs in the XML file, but your organization has set the TMS_DSI local reference codelist value GLOBALVTA to not accept Global VTAs, you will receive errors or warnings for those VTA records, depending on the setting of the codelist value, as follows:

- Fail Global VTA (F). If GLOBALVTA is set to F, and the sponsor sends a Global VTA in the XML file, the system does not import the record. The record remains in the error file with an appropriate error.

- Domain Only (D). If GLOBALVTA is set to D, and the sponsor sends a Global VTA in the XML file, the system creates a Domain VTA for each Global VTA sent, and associates a warning with the record in the error file.

If GLOBALVTA is set to Allow Global VTAs (A), the system imports any Global VTAs sent by the sponsor as Global VTAs, and you never receive Global VTA-related errors or warnings.

Source Term Actions The sponsor may have assigned an Action (either a global or a Discrepancy Message) to one or more omissions, indicating that the source term itself must be changed in the external system before it can be processed in TMS. You can see these Actions in the Classify VT Omissions window.

If you are using Oracle Clinical, these Actions are automatically transmitted to Oracle Clinical during the next Batch Validation. If you use a different external source data system, you must ensure that the Actions are sent to that system.

The import job associates a warning with the omission in the error file in the following situations:

- The omission no longer exists in your system (because it has been manually classified at your site or deleted in the source system).

- The omission exists but is currently owned by the external source data system (Oracle Clinical, for example). This would occur if someone assigned an Action to the omission at your site and sent it to the external system.

Missing Dictionary Terms The import process compares the dictionary terms used in sponsor VTAs in the file to the dictionary terms on your site. If any dictionary terms required by the sponsor are missing on your site, you must add them, either individually (see "Modifying Repository Data in the Maintain Repository Data Window" on page 12-9) or by reloading the sponsor's version of the dictionary.

Your organization's setting of the TMS_DSI local reference codelist value OUTSYNCCSP determines how your system handles this situation. The options are:

- OUTSYNCCSP=E (Fatal Error). The import job fails at the first such error it finds. The error file provides information on the missing dictionary term. You cannot import the remaining data in the file until you add the required dictionary term to your database. However, records that were successfully processed before the failed record remain changed in the database. The error file contains a record of each successfully processed record.

  After you add the required dictionary term, you can remove the fatal error message from the XML error file, move the file to the import directory, and run the import process again to process the remaining records.

- OUTSYNCCSP=W (Warning). The import job succeeds, and records without errors are successfully imported. Any VTA whose dictionary term is missing in your system is not imported and an omission is created for that VTA on your site. (Unlike the behavior on the sponsor site, the system does not create an Action associated with an omission for the source term whose VTA's dictionary term is missing in the importing system.)

  You must add the dictionary term manually to bring your dictionary into synchronization with the sponsor's, or reload the sponsor's version of the dictionary.

- OUTSYNCCSP=N (Neither). The import job succeeds, and records without errors are successfully imported. Any VTA whose dictionary term is missing in your system is not imported and an omission is created for that VTA on your site. There is no mention of the missing dictionary term in the error file.

Updating External System Data

You must run the necessary job in the external source data system to synchronize its data with the sponsor site's TMS repository.

If the source data system is Oracle Clinical, the Batch Validation job derives TMS values into Oracle Clinical.

Using DSI from the Sponsor Site

This section includes:

- Importing Data on page 5-17

- Processing Source Site Data on page 5-19

- Exporting Data on page 5-19

- Removing X Area Data from the Sponsor Site on page 5-20

Importing Data

A source site sends data in one or more XML files to your input directory for processing. Each XML file contains data for a single X Area. Typically, an X Area corresponds to a study; see "Defining X Areas" on page 5-9.

To import the data into your system, run the import job; from the **VTA Maintenance** menu, select **Disconnected System Integration**, then **Import DSI Data** (see "Import DSI Data" on page 5-25). The import job runs on all XML files in the input directory.

As part of the import process, Autoclassification runs on all omissions in the X Area, including those sent by the source site.

You can see the status of import files and data; from the **VTA Maintenance** menu, select **Disconnected System Integration**, then **Maintain DSI Files**. For instructions, see "Viewing XML File and Record Statuses" on page 5-20.

This section describes how the import process handles the following situations: Adding New Source Terms, Deleting Source Terms, Classification Conflicts, Global VTA Conflicts, Importing Approved and/or Unapproved VTAs, and Missing Dictionary Terms.

Adding New Source Terms Any terms that have been updated (for example, through Oracle Clinical Data Entry Update) or added to TMS from the CRO's external source data system since the last source site data export/sponsor import are included in the XML file and imported by the system, associated with their VTAs, omissions, and associated Informative Notes, if any.

Deleting Source Terms The source site may mark source terms for deletion if they have been deleted in the external source data system. The import process deletes these terms from your database as well.

Classification Conflicts The import process compares the VTAs in the XML file with those for the same verbatim terms in your database. If a source site VTA conflicts with your VTA for the same verbatim term—the verbatim term is to a different dictionary term, or one partner has given the VTA a subtype of Misspelled and the other a subtype of Accepted—the system does not import the source site VTA. It remains in the error file, associated with a warning, and the system sets the verbatim term's **Update** box to Y in your database, forcing the record to be included in your next DSI data export XML file. When the source site imports that file, your VTA overwrites the source site's in the source site's database.

Global VTA Conflicts If the source site sends Global VTAs in the XML file, but your organization has set the TMS_DSI local reference codelist value GLOBALVTA to not accept Global VTAs, you will receive warnings for those VTA records. The system either does not import the record or imports it as a Domain VTA, depending on the setting of the codelist value, as follows:

- Fail Global VTA (F). If GLOBALVTA is set to F, and the source site sends a Global VTA in the XML file, the system does not import the record. The record remains in the error file associated with an appropriate error.

- Domain Only (D). If GLOBALVTA is set to D, and the source site sends a Global VTA in the XML file, the system creates a Domain VTA for each Global VTA sent, and associates a warning with the record in the error file.

In either of the above cases, the system sets the record's **Update** box to Y in your database, forcing the record to be included in your next DSI Data Export XML file. When the source site imports that file, your VTA overwrites the source site's in the source site's database.

If GLOBALVTA is set to Allow Global VTAs (A), the system imports any Global VTAs sent by the source site as Global VTAs, and you never receive Global VTA-related errors or warnings.

Importing Approved and/or Unapproved VTAs The setting of the TMS_DSI local reference codelist value CREVTA determines whether your system imports VTAs created by the source site as Approved, Unapproved, or some of each, as follows:

■ Unapproved (U). If CREVTA is set to U, the system imports all VTAs created by the source site as Unapproved, regardless of whether or not they were created as Approved at the source site. You must examine each one manually and either approve it or reclassify the verbatim term. See "Approving a Nonapproved VTA" on page 10-13 and "Reclassifying a Verbatim Term" on page 11-9.

> **Note:**   Even if you have this reference codelist set to U, the import job imports VTAs that are direct matches to dictionary terms—that is, VTAs that were created during Autoclassification at the source site—as Approved.

■ XML (X). If CREVTA is set to X, the system imports each VTA created by the source site with the same approval status given them at the source site. You must manually process only the Unapproved VTAs. This is the default setting.

■ Approved (A). If CREVTA is set to A, the system imports all VTAs created by the source site as Approved, regardless of whether or not they were approved at the source site. No manual processing is necessary. This setting is not recommended.

Source Site-Created Actions The source site may have assigned an external Action to one or more verbatim terms, indicating that the source term itself must be changed in the external system before it can be processed in TMS. These Actions are imported in association with an omission. You can see these Actions in the Classify VT Omissions window.

You can modify source site-created Actions only if the Action is currently owned by TMS. This can happen at two points:

■ when the source site has not yet sent the Action to the source data system (via Batch Validation, if the external system is Oracle Clinical)

■ when the external system sends the Action back to TMS with a comment or question

Missing Dictionary Terms The import process compares the dictionary terms used in source site VTAs to the dictionary terms on your site. If any dictionary terms required by the source site are missing on your site, it indicates that the source site has added dictionary terms that your organization does not support. The dictionaries at the two sites must be re-synchronized.

Your organization's setting of the TMS_DSI local reference codelist value OUTSYNCCSP determines how your system handles this situation. The options are:

■ OUTSYNCCSP=E (Fatal Error). The import job fails at the first such error it finds. The error file provides information on the missing dictionary term. Records that

were successfully processed before the failed record remain changed in the database. The error file contains a record of each successfully processed record.

Call the source site, resolve the dictionary discrepancy, and ask the source site to reexport the whole X Area.

■ OUTSYNCCSP=W (Warning). The import job succeeds, but VTAs whose dictionary terms are missing in your system are not imported. The system creates an Action associated with an omission for the source term whose VTA's dictionary term is missing in your system. This Action is visible in the Classify VT Omissions window, as well as in the error file. It is not sent to the external source data system, as other Actions are. The Action remains associated with the omission until it is resolved. You can also read the error file to find these VTAs and the dictionary term they require. Records without errors are successfully imported.

Call the source site, resolve the dictionary discrepancy, and ask the source site to reexport the whole X Area.

■ OUTSYNCCSP=N (Neither). The import job succeeds, and records without errors are successfully imported. Any VTA whose dictionary term is missing in your system is not imported and an omission is created for that VTA on your site. If an error file is created, it contains no mention of the missing dictionary term.

The system creates an Action associated with an omission for the source term whose VTA's dictionary term is missing in your system. This Action is visible in the Classify VT Omissions window. It is not sent to the external source data system, as other Actions are, or to the source site. The Action remains associated with the omission until it is resolved.

Call the source site, resolve the dictionary discrepancy, and ask the source site to reexport the whole X Area.

Processing Source Site Data

After running Autoclassification, you must manually process the following source site data. When you update a record, the system sets its **Update** box to Y, causing the next export job to extract the record to send back to the source site.

Omissions You must manually classify, or assign Actions to, all the source terms still unclassified after Autoclassification. See "Classifying Terms Manually" on page 10-3.

Unapproved VTAs You must examine each Unapproved VTA received from the source site and either approve it or reclassify the verbatim term.

Conflicting Classifications In the error file, the import process associates warnings with all source site classifications that conflict with VTAs in your database. You can view the conflicts in the error file. The system also sets the **Update** box to Y for the relevant verbatim term in the database, forcing the record to be included in the next export XML file. During the next source site import, your VTA will automatically overwrite the source site's.

If VTAs conflict because the source site used a dictionary term that does not exist in your repository, the job may fail. See "Missing Dictionary Terms" on page 5-18.

Exporting Data

When you have finished processing data from the source site, you must extract the updated data for the relevant X Area into an XML file and send it back to the source site. You can either export data from a single X Area (see "Export DSI Data" on page 5-24) or from all active X Areas at once (see "Export Active X Areas" on

page 5-26). Both jobs export only changed data, using both timestamps and the **Update** setting to determine which records have been updated.

If necessary, you can force the export of all data by using the Force Rederivation job (see "Force Rederivation" on page 5-28) to set all records' **Update** boxes to Y. Then you must run either Export DSI Data or Export Active X Areas to export the data.

Data Exported The data extract jobs automatically extract the following types of data in the required XML format:

■ All verbatim term omissions with new Actions

■ All VTAs for verbatim terms based on source terms from the receiving source site that have been created or changed since the last export, including classifications, reclassifications, and approvals

> **Note:** If the local reference codelist TMS_DSI parameter XAPPVTA is set to Y, the system exports only Approved VTAs. Unapproved VTAs are not exported.

■ New and changed status audit Informative Notes associated with verbatim terms

■ All VTAs declassified since the last export of the same X Area

Export Statuses You can check on the export's status (see "Viewing XML File and Record Statuses" on page 5-20). Export jobs can have only two statuses: Success and Fatal Error. The export process should fail only if there is a problem with the Output directory.

Sending the XML File to the Source Site

DSI does not handle transporting the XML file to the source site. You must do it manually; for example, zipping the file and sending it via either FTP or email.

To avoid sending the same files to the partner for unnecessary reprocessing, manually delete files from the output directory after sending them to the other site.

Removing X Area Data from the Sponsor Site

Faulty data may be imported into the sponsor site. Therefore, if you have special privileges, you can delete all data belonging to a specific X Area. See "Delete DSI X Areas" on page 5-27. You can then reexport all X Area data from the source site (see "Force Rederivation" on page 5-28) and reimport the X Area.

Viewing XML File and Record Statuses

From the **Disconnected System Integration** menu, select **Maintain DSI Files**.

In the Maintain DSI Files window you can view all the import and export jobs associated with each X Area, and see how many records in each file were successfully updated, updated with a warning, and failed. In addition, you can assign a manual status to a file to reflect the state of your work on the file, and view the error file for each job. For more information, see:

■ Viewing File and Record Status on page 5-21

■ Setting the Filter on page 5-22

■ Viewing Error Files on page 5-23

Viewing File and Record Status

The Maintain DSI Files window has two sections: X Areas and Files. After you execute a query, the system displays all the X Areas that meet the filter and query criteria in the X Areas section. In the Files section, the system displays all the files that meet the criteria for the X Area selected in the X Areas section.

To view the files for a different X Area, click its name in the X Area column. In the Files section, the system displays its files that meet the filter and query criteria. For each file, the system displays the following information:

- File Name. The file name is generated by the system at the time of export or import (see "XML File Names" in Appendix B, "Disconnected System Integration XML Files" for information on the naming convention).

- Process Type: Import or Export.

- Process Status. The way the system determines the process status depends in part on the settings of the local reference codelist TMS_DSI. See"Using DSI from the Source Site" on page 5-14 and "Using DSI from the Sponsor Site" on page 5-16 for the implications of the statuses on your site. The possible statuses are:

  - Success. All records were processed successfully, with no warnings or errors. There is no error file.

  - Warnings. All records were processed successfully, but at least one record has an error. You can see warnings associated with each record with an error in the error file. Depending on the OUTSYNCCSP reference codelist setting, there may be a discrepancy message associated with VTA records whose dictionary term is not included in your dictionary.

  - Errors. Some records may have been processed successfully, but at least one record failed import and remains in the error file associated with an error message.

  - Fatal. The job failed due to a problem with the directory or the XML file itself, or the source site and sponsor dictionaries were out of synch and OUTSYNCCSP was set to fail in that situation.

- Manual Status. This status is meant to indicate the state of your work on files with warnings or errors, for use in querying. The system does not enforce any behavior related to the manual status. The possible statuses are:

  - New.

  - Pending.

  - Fixed.

- # Source Data Records. The following three columns concern source term records that are new, updated, or deleted. To see warnings and errors, click **View File** or select **Options**, then choose **View File**.

  - OK: The number of source terms successfully processed without warnings.

  - Warning: The number of source terms successfully processed but with warnings.

  - Failed: The number of source terms with errors; these records were not successfully processed.

- # VTA Data Records. This includes new verbatim term assignments (VTAs, or classifications) reclassified and declassified VTAs, and newly approved or

unapproved VTAs. The following three columns concern records that are new VTAs:

– OK: The number of VTAs successfully processed without warnings.

– Warning: The number of VTAs successfully processed but with warnings. You can see the warnings in the error file.

– Failed: The number of VTAs with errors; these records were not successfully processed. You can see the errors in the error file.

■ Process Start Time. The time the import or export process began.

■ Process End Time. The time the import or export process finished.

■ Process Elapsed. The total amount of time spent processing the file, in HH:MM:SS format.

■ Audit Information. The following three fields contain audit information for the file selected above:

– **Created By**

– **Creation Time**

– **Modified By**. The user ID of the person who last changed the manual status.

– **Modification Time**. The timestamp of the last manual status change.

■ Filter. The system displays the current filter settings.

Deleting Files

You can delete X Area files from view by selecting **Record**, then **Delete**. This does not delete the file itself, only the database information on the file. You can no longer view the file from this window.

Setting the Filter

The filter determines which X Areas and files are displayed in the Maintain DSI Files window. Your default filter settings are set by selecting the **Definition** menu, then **Maintain Settings**.

You can reset the filter at any time by clicking the **Filter** button in the Maintain DSI Files window. Enter values in the filter pop-up box to limit the files displayed in the Maintain DSI Files window. By default all files are displayed. All settings are optional.

■ **External System**. From the list, select the external system where the source data was originally collected.

■ **Source Data DB**. From the list, select the database of the external system where the source data was originally collected.

■ **X Area** and **X Area Name**. From the list, select the X Area. The list of values in the **X Area** field includes both the X Area ID and name, and when you select a value, the system populates both fields.

■ **X Area Status**. From the list, choose the current status of the X Area(s) for the file(s) you want to see. The choices are **Active**, **Inactive**, and **Complete** (for an explanation of the statuses, see "Defining X Areas" on page 5-9).

■ **Last Export Between**. Enter the start and end dates of the period in which jobs you want to see were executed. Use the format specified for your installation in OPA Settings (OPA_SQL_DATE_FORMAT); for example, if you use the format DD-MON-YYYY, enter 15-OCT-2004.

- **Manual Status**. From the list, select the manually set status of the file(s) you want to see. The choices are: **New**, **Pending**, and **Fixed**.

- **Process Type**. From the list, select the process that produced the file you want to see; either **Export** or **Import**.

- **Process Between**…**and**.… Enter the start and end dates of the period in which jobs you want to see were executed. The system returns all files that were even partially executed during that data range. Use the format specified for your installation in OPA Settings (OPA_SQL_DATE_FORMAT); for example, if you use the format DD-MON-YYYY, enter 15-OCT-2004.

- **Process Status**. Select one or more status. The system will display files resulting from jobs with the statuses you select, and that meet the other criteria.

Click one of the following buttons:

- **Clear**. Returns values to what they were when you opened the filter.

- **Restore**. Resets the filter to the default values (from the **Definition** menu, select **Maintain Settings** to set).

- **OK**. Saves the current settings. The filter pop-up window disappears, leaving the Maintain DSI Files window in view. You must execute a query (from the **Query** menu, select **Execute Query** or press F8) to display the files that meet the filter criteria. The filter criteria are displayed at the bottom of the Files section of the window. These settings persist the next time you open the Maintain DSI Files window.

- **Cancel**. Closes the Filter window without saving any changes.

Viewing Error Files

In the **Maintain DSI Files** window under the **VTA Maintenance** menu, you can see the results of each DSI export and import job. If a job had errors, you can look at the error file to find the errors and then fix them and rerun the job.

To find a job, do the following:

1. Click **Filter** to limit the jobs returned by any or all of the following criteria:

    - External system

    - Source database

    - Remote partner

    - X Area

    - X Area status

    - Last export date range

    - Manual status

    - Process type

    - Process date range

    - Process status (Fatal, Error, Warning, and/or Success)

2. Do one of the following:

    - Click **OK** to implement these Filter settings for your current TMS session and close the Filter window.

- Click **Cancel** to revert settings to their values when you opened the Filter window. The Filter window closes.

- Click **Clear** to revert the settings to their values when you opened the Filter window. The Filter window remains open.

- Click **Restore** to revert Filter window values to the default settings for your TMS user profile. After clicking Restore, clicking Cancel cannot bring back the settings you had when you opened the filter window. The Filter window remains open.

3. Enter a query in the **X Areas** block at the top of the window. TMS displays the X Areas matching your filter and query criteria.

4. Select the X Area.

5. Enter a query in the **Files** block. TMS displays the files matching your filter and query criteria for the X Area you selected.

6. Select the file you want to view and click **View File**. A browser window opens and displays the file.

The file is in XML format. It is the same XML file that was imported or exported, except that a warning or error is displayed with each record that has a warning or error.

DSI Batch Jobs

This section includes information on DSI batch jobs:

- Export DSI Data on page 5-24

- Import DSI Data on page 5-25

- Export Active X Areas on page 5-26

- Export X Areas by External System/Instance on page 5-26

- Import X Areas by External System/Instance on page 5-27

- Delete DSI X Areas on page 5-27

- Delete Remote DSI Databases on page 5-28

- Force Rederivation on page 5-28

See also Running the DSI Initialization Job on page 5-7.

> **Note:** Only superusers can run DSI batch jobs. For information on setting up a superuser account, see "Defining a New User" on page 3-5.

Export DSI Data

This job extracts data for a single X Area into an XML file located in the Output directory. It processes only data that has been added, deleted, or updated since the last export job was run ("incremental" data processing). The export process should fail only if there is a problem with the receiving output directory. To run the job:

1. Go to **Disconnected System Integration**, then select **Export DSI Data**.

2. Enter values for the following parameters. A list of values is available for the first three.

- **External System**. The name of the external system where the source data originated.

- **Instance**. The database where the source data originated.

- **X Area**. The X Area that corresponds to the unit whose data you want to export; typically a study or case.

- **Report Server Name**. Enter the name of the report server you want to run the job.

3. Run the job. You can schedule this job to be run on a regular basis such as nightly or weekly; see "Setting Up and Running Reports and Other Batch Jobs" on page 2-13.

Import DSI Data

The import process operates on all XML files currently in the input directory, and deletes them from that directory upon completion. If there are errors or warnings, or the job fails, the system inserts the errors into the XML file associated with the appropriate records, and moves the XML file to the error directory.

The import process begins by calling the XML parser to validate the structure of the XML file against the XML schema. If it is invalid, the job fails. The import job also validates the XML metadata against the mappings defined in the sponsor.

The data import job operates on data differently at the source site and the sponsor site; see "Using DSI from the Source Site" on page 5-14, and "Using DSI from the Sponsor Site" on page 5-16. The way it handles different errors depends on the settings of several reference codelist values in the importing system; see "Setting DSI Preferences with Reference Codelist Settings" on page 5-11.

To run the job:

1. Go to **Disconnected System Integration**, then select **Import DSI Data**.

2. Enter the name of the report server you want to run the job.

3. Run the job. You can schedule this job to be run on a regular basis such as nightly or weekly; see "Setting Up and Running Reports and Other Batch Jobs" on page 2-13.

Import Process Statuses   Import jobs can end with four possible statuses: Success, Warning, Error, and Fatal Error. Jobs that end with a status of Success do not have error files. Jobs ending with all other statuses do have error files.

To create the error file, the import process inserts errors or warnings, associated with the appropriate record, into the XML file in the input directory and then moves the XML file into the error directory.

Success If all records are imported without errors or warnings, DSI gives the job a status of Success, updates the timestamp of each imported record in the importing database, and deletes the XML file from the input directory. It does not produce an error file.

Warning If the import process encounters data errors or conflicts, and you have set up DSI to continue even with errors (see "Setting DSI Preferences with Reference Codelist Settings" on page 5-11), the import job:

- successfully imports all records that do not have warnings, and removes those records from the XML file

- for records with errors or conflicts, does one of the following, depending on reference codelist settings:

  - does not import the record; leaves it in the error file associated with the appropriate error

  - imports the record but associates a warning with the record in the error file; in addition, may associate a Discrepancy Message with the record

Error If the import process encounters data errors or conflicts, and you have set up DSI to fail in that situation (see "Setting DSI Preferences with Reference Codelist Settings" on page 5-11), the import job:

- does not import records with errors, but keeps them in the error file associated with an error message

- imports records successfully before encountering the error but also leaves these imported records, and the unprocessed ones listed after the record that caused the error, in the file

Fatal Error The following types of errors are fatal and always result in job failure:

- An input directory does not exist in the importing system, or an output directory does not exist on the exporting system.

- (Import only) The XML file structure is invalid.

- (Import only) The dictionary required in the XML file does not exist in the importing system.

- (Import only) The Oracle database user has not been given Delete privileges for the input directory.

- (Import only) A new VTA is created using a dictionary term that does not exist in the importing system.

- (Import only) The input XML file contains the wrong external system name or instance name for the specified X Area.

The error file includes information about the fatal error and all the records originally contained in the file.

Export Active X Areas

This job extracts data for all active X Areas, creating a separate XML file in the Output directory for each active X Area. It processes only data that has been added, deleted, or updated since the last export job was run ("incremental" data processing). The export process should fail only if there is a problem with the receiving output directory.

1. Go to **Disconnected System Integration**, then select **Export DSI Data**.

2. Enter the name of the report server you want to run the job.

3. Run the job. You can schedule this job to be run on a regular basis such as nightly or weekly; see "Setting Up and Running Reports and Other Batch Jobs" on page 2-13.

Export X Areas by External System/Instance

This job allows you to export data for a particular X Area from a single site. For example, a sponsor may want to export data from a single CRO rather than all CRO participating in the same study (X Area) at the same time.

To run the job:

1. Go to **Disconnected System Integration**, then select **Export X Areas by External System/Instance**.

2. From the **External System** drop-down list, select the external system to which you want to export data.

3. From the **Instance** drop-down list, select the database to which you want to export data.

4. Enter the name of the report server you want to run the job.

5. Run the job. You can schedule this job to be run on a regular basis such as nightly or weekly; see "Setting Up and Running Reports and Other Batch Jobs" on page 2-13.

Import X Areas by External System/Instance

This job allows you to import data for a particular X Area from a single site. For example, a sponsor may want to import data from a single CRO rather than all CRO participating in the same study (X Area) at the same time.

To run the job:

1. Go to **Disconnected System Integration**, then select **Import X Areas by External System/Instance**.

2. From the **External System** drop-down list, select the external system from which you want to import data.

3. From the **Instance** drop-down list, select the database from which you want to import data.

4. Enter the name of the report server you want to run the job.

5. Run the job. You can schedule this job to be run on a regular basis such as nightly or weekly; see "Running a Job" on page 2-16.

Delete DSI X Areas

Available on the sponsor site only. If faulty data has been loaded into the sponsor site or exported from the source site, it may be necessary to remove all data belonging to the X Area and start again. This job deletes all data belonging to a single X Area: the X Area definition, source terms, omissions, and the database records of the X Area files (not the files themselves).

To run the job:

1. Go to **VTA Maintenance**, select **Disconnected System Integration**, then **Delete X Area**.

2. Enter the following parameters:

   - **External System**. The name of the external system where the source data originated.

   - **Instance**. The database where the source data originated.

   - **X Area**. The X Area that corresponds to the unit whose data you want to export; typically a study or case.

   - **Report Server Name**. Enter the name of the report server you want to run the job.

If data has become corrupted on the sponsor site, you can export all relevant data from a particular X Area to the other system.

1. Delete the X Area on the sponsor site by running the Delete DSI X Areas job on the sponsor.

2. On the source site, run Force Rederivation on the same X Area.

3. On the source site, run Export DSI Data on the same X Area.

4. Send the data from the source site to the sponsor.

5. Run Import DSI Data on the sponsor site to recreate the X Area and all its data.

Delete Remote DSI Databases

To delete a registered remote database from a sponsor site, do the following:

1. Manually delete all DSI definitions to the remote database in the **Define DSIs** window, using **Delete** from the **Record** menu. This has the effect of deleting all the associated X Areas as well.

2. Run the **Delete Remote DSI Databases** job.

To run the job:

1. In the **Definitions** menu, click **Jobs** and then click **Delete Remote DSI Databases**. The job window opens.

2. In the job-specific parameter **Remote Instance** field, select the name of the remote instance from the drop-down list. The system displays only remote instances that are not part of a DSI definition.

3. Run the job. See "Running a Job" on page 2-16.

Force Rederivation

This job marks all source terms and omissions in a single X Area for processing in the next export process, whether or not they have been added or modified since the last export job ("full" data processing). To actually process the data, you must run the Export DSI Data job.

If data has become corrupted on the other site, or dictionaries are no longer synchronized, you can export all relevant data from a particular X Area to the other system.

1. From the **Definition** menu, select **Define Disconnected System Integrations**, then **X Areas**, and click the **Force Rederivation** button. The system marks for mandatory processing (Update=Y) all data in this X Area of the types normally exported from the system (which vary, depending on whether the exporting system is a sponsor or a source site).

2. From the **DSI Maintenance** menu, select **Export DSI Data** and run the job; see "Export DSI Data" on page 5-24.

Defining External System Information in TMS

This section includes:

- Where External System Information Appears on page 5-29
- Defining the External System in TMS on page 5-30
- Setting Up External System Columns and Details on page 5-31
- Defining Views and Functions in the External System on page 5-33
- Setting Up External System Drill-down Queries on page 5-36

When a pharmaceutical application, or **external system**, is fully or partially integrated with TMS, you can bring information into TMS from the external system to provide context for each verbatim term occurrence (source term, question response). For example, if TMS is integrated with Oracle Clinical, when Oracle Clinical sends a question response to TMS for classification, it also sends information about the response: its associated Patient, Study, Project, DCM, Visit, Document Number, Investigator, and Discrepancy ID.

In the case of Oracle Clinical, these database columns are predefined. If you are using TMS with a different external system, you must specify any database column information you want to bring into TMS from the external system, associated with each verbatim term occurrence. You can specify up to eight columns from the external system's database for this purpose. TMS then stores all such data in its database, linked to the original verbatim term occurrence (source term).

In addition, you can specify up to twelve details per column for propagation into TMS associated with each verbatim term occurrence. Column details are not predefined for Oracle Clinical, but as with other external systems, you can specify them.

Detail information is retrieved via either a view or a function defined for this purpose within the external system (see "Define Views in the External System" on page 5-33 and "Define Functions in the External System" on page 5-35).

You can set up external system information in TMS for any number of external systems, and specify full, partial, or no integration for any of them.

For external systems other than Oracle Clinical, use the Define External Systems window to specify what information you want to have available in TMS as well as its label and display size. See "Setting Up External System Columns and Details" on page 5-31.

For all external systems, do the following for each column for which you wish to import additional details:

- Create a view or function. See "Defining Views and Functions in the External System" on page 5-33.

- Grant access privileges within the external system to the view for all users who need to see the information within TMS.

Where External System Information Appears

If the external system is fully integrated with TMS, TMS displays external system column information in the following areas of the TMS application.

TMS Windows

TMS windows can display external system information from partially and fully integrated systems. This information is for query purposes only.

External system columns from fully integrated systems appear in the following windows: Classify VT Omissions, Approve VTAs, Reclassify Verbatim Terms, High-Level Reclassification, and Browse VT Classification Data. For partially integrated systems, external system information is available in the Classify VT Omissions window only.

If detail information is available for a particular column, the text in the column's field appears in blue. The user can select **Drill Down** from the field to see the detail information in the External Drill-Down pop-up window.

HTML Browser

The Source Data feature enables you to search for verbatim terms directly in the source external system by a character string, and return related column information. You can define standard queries to determine which column information is returned to TMS. The HTML browser user selects the query, or column group, he or she needs. See "Searching for Source Data" on page 14-29 and "Setting Up External System Drill-down Queries" on page 5-36.

Defining the External System in TMS

To integrate TMS with any system other than Oracle Clinical, you must complete the steps in this section.

The general external system information is pre-defined for Oracle Clinical. If you are using Oracle Clinical, you can skip this section and proceed to "Setting Up External System Columns and Details" on page 5-31.

To define an external system:

1. Select **Definition**, then **Define External Systems**. The Define External Systems window appears.

2. In the **External System** field, enter the integration key that will identify the external system to TMS. For example, Oracle Clinical's is pre-defined as OCL.

3. Enter the name of the external system in the **Application Name** field. This name will appear in the Classify VT Omissions, High-Level Reclassification, and Browse VT Classification Data windows in the **External Systems** list that allows you to see and query on the external system information.

4. From the **Integration Level** list, choose **Full**, **Partial** or **None** according to the level of integration between TMS and the external system. See "Full Integration" on page 5-2 and "Partial Integration" on page 5-3.

5. Enter a **Description** of this external system.

6. **VT HTML Data Function** (for use with the HTML browser only) – Enter a database function name for returning source data for TMS to display in HTML format via VT classification. You must use *package_name.function_name* format. See "Define Functions in the External System" on page 5-35. There is a sample function available on My Oracle Support; see article ID 258986.1.

7. Skip the **Object HTML Data Function** field. This field is not currently used by TMS.

8. **Display X Area Name Function** (optional). Enter a database function name for displaying a name corresponding to an X Area value in reports where the X Area is displayed. If you leave this field empty, TMS reports display the numerical X Area ID.

   The function should accept only the X Area as parameter and return a string representing the name of the X Area. Note that if the function is not owned by TMS, then execute on the function will need to be granted to TMS.

   If Oracle Clinical is the external system, TMS automatically displays the study name. If you prefer to display something else, you can write a function for that purpose, entering the function name in this field in place of the default function name, rxc_tms_access.studyName.

9. The **Discrepancy Message Omission Statuses** list of values displays the list of allowed omission statuses for Actions of Action Type Single Term. For more information, see "Defining and Using Actions" on page 7-6.

10. The **Multiple Term Action Omission Statuses** list of values field defines the list of allowed omission statuses for Actions of Action Type Answerable or Unanswerable. See "Action Types" on page 7-6.

11. Save.

Setting Up External System Columns and Details

After you specify the general external system information, you can define up to eight columns that map to data columns in that external system. Subsequently, for each external system column, you can define up to twelve additional details that provide more information about that external system data.

For example, if you define Patient ID as one of the columns, you could define details for the related first name; last name; sex; screening, enrollment, and termination dates; and dates of birth and death—if the external system stores that information.

These processes are divided into two sections:

- Define External System Columns on page 5-31
- Define Column Details on page 5-32

Define External System Columns

Column names will appear as field labels in Classify VT Omissions, and if the system is fully integrated with TMS, in High-Level Reclassification, Approve VTAs, Reclassify Verbatim Terms, and Browse VT Classification Data. Oracle Clinical columns are pre-defined as Patient, Study, Project, DCM, Investigator, Visit, Document Number and Discrepancy ID.

You should define columns in the order you want them displayed. TMS assigns a read-only Map ID Number to each column row that you define in the Attributes block, numbering them consecutively and in ascending order. The Map ID Number is important: you use this number for other processes in TMS that use external system data, such as Defining Views and Functions in the External System and Defining HTML Layout Functions.

Each column can store values of up to 500 characters (VARCHAR2).

The external system columns are defined for Oracle Clinical by default. To define column information for any other external system:

1. In the External System block, highlight the name of the external system for which you are defining information.

2. Click the **Attributes** tab and in the upper block of that tab, click in an empty row or insert a record.

3. **Column Name**. Enter text to appear as the label for the field that will contain information from this database column.

4. Enter a **Description** of the column.

5. **Drill Down Type**. If you plan to bring details about this column into TMS from the external system, specify a drill-down type here. Choose **Function** if you plan to use a function to provide detailed information from the external system to TMS, or **View** if you plan to use a view. Leave this field blank if you do not want detailed information from this column to be available in TMS.

6. If you entered a value in the **Drill Down Type** field, in the **View/Func Name** field, enter the name of the view or function that you will use to retrieve drill-down data for this column from the external system. If you specify a function, you must enter

it in *package_name.function_name* format. Leave this field blank if you do not want detailed information for this column to be available to TMS.

To create the view or function you want to use, see "Define Views in the External System" on page 5-33 or "Define Functions in the External System" on page 5-35.

7. **View Where Clause** (Views only). If necessary, enter a Where clause to filter the information. For example:

```
ext_col_det1 = :xv1 and ext_col_det2 = :xv2
```

In addition to the eight xv variables, you can also reference the following bind variables in the Where clause:

source term ID (:sourceTermId)
name of the instance (:defInstanceName)
integration key of the external system (:defIntegrationKey)
dictionary content ID (:dictContentId)
update flag (:updateFlag)
domain ID (:defDomainId)
external area (:xarea)
source term alt key (:stAltKey)

8. Save.

9. Repeat. You can define up to eight columns per external system.

Define Column Details

If you define column detail information in this window, it is available from the TMS windows Classify VT Omissions, Approve VTAs, Reclassify Verbatim Terms, Browse VT Classification Data, and, if the external system is fully integrated with TMS, from High-Level Reclassification.

When you define detail information for a column, TMS displays the text in the field corresponding to the column in blue. The user can then use the drill-down function to see detail information.

Whether you chose a view or a function for retrieving external system data for this column, you must be sure to include the detail information you are defining here in the view or function. To create the view or function you want to use, see "Define Views in the External System" on page 5-33 or "Define Functions in the External System" on page 5-35.

To define column detail display:

1. In the upper block of the **Attributes** tab, highlight the column for which you have created a view.

2. In the Details block, click in an empty row or insert a record.

3. Fill in the fields as follows:

   - **Map ID** – Display-only. TMS numbers column details consecutively by default. Maps to ext_col_det1…12 in the view.

   - **Label** – Enter the label text to appear in the drill-down window. This does not have to be the same as the database element's name.

   - **Sort Order** – In the **Sort Order** field, indicate where you want TMS to display this detail in relation to other details for this column. Number 1 appears highest on screen, Number 2 next, etc.

   - **Value Length** – Enter the display length for the detail.

4. Save.

5. Repeat for each detail, up to twelve details per column.

Defining Views and Functions in the External System

This section describes how to define and use views and functions to access data from the external system. You can also refer to the sample drill-down views and functions available on My Oracle Support.

This section includes:

- Define Views in the External System on page 5-33

- Define Functions in the External System on page 5-35

Define Views in the External System

To see up to twelve (12) detail values associated with a column, you must define a view in the external system and specify it in the Define External Systems window in TMS, in the Details block. The existence of a view name associated with an external column triggers the display of the column name in blue in TMS, indicating that drill-down information is available.

When you create the view, you must use the detail names shown below.

Note that you can define a Where clause for each column in the TMS user interface (see "Define External System Columns" on page 5-31). The Where clause is appended to the view definition at execution time.

Grant select privileges and create a synonym as shown in the last three lines of the statement. See "Security" on page 3-1 for more information.

Connect to the database as a user with access to the external system's tables from which the view is created and use the script in Example 5–1 as a model.

Example 5–1   Sample External System View Definition Logic

```
CREATE VIEW view_name (
```

ext_col_det1	These detail names are fixed; TMS uses them in its internal code. You must define at least ext_col_det1…12. You can define more than 12 if you want, and use the select statement to limit the columns used.
, ext_col_det2	
.	
.	
.	The detail number will be imported as the sort order in TMS; ext_col_det1 will be listed first, then ext_col_det2, etc.
, ext_col_det12	

```
) AS
```

SELECT
     column_detail1

, column_detail2

.

.

.

, column_detailn

FROM *table_name1*

.

.

.

    table_namen

WHERE…

Select the actual column detail names from the external system that you want to use in TMS. You can specify up to twelve (12).

If you need fewer details, enter *null* in each extra line of code.

Specify the external system's table or tables from which you want to create the view.

Describe the condition(s) you want. If the external system is Oracle Clinical, include a call to a predefined function to limit the data retrieved to studies to which the user has the proper security privileges (see Example 5–2). The call is:

```
rxc_tms_access.studyAccess ([table_
alias].[clinical_study_id])=1
```

GRANT SELECT ON *view_name* TO rxclin_read;

GRANT SELECT ON *view_name* TO tms WITH GRANT OPTION;

CREATE PUBLIC SYNONYM *view_name* FOR *user.view_name*;

Example 5–2   Sample External System View Definition Code

```
CREATE OR REPLACE VIEW dmo_dcms (
        ext_col_det1
,       ext_col_det2
,       ext_col_det3
,       ext_col_det4
,       ext_col_det5
,       ext_col_det6
,       ext_col_det7
,       ext_col_det8
,       ext_col_det9
,       ext_col_det10
,       ext_col_det11
,       ext_col_det12
,       clinical_study_id
,       response_id
,       order_ts
)
as select distinct
        resp.response_id
,       decode( resp.end_ts
        , to_date( 3000000, 'J')
          , decode( rdcm.accessible_ts
            , to_date( 3000000, 'J'), 'C'
```

```
                      , 'S'
                      )
              , 'O'
              )
,       to_char( resp.response_entry_ts, 'DD-MON-YYYY HH24:MI:SS')
,       to_char( resp.end_ts, 'DD-MON-YYYY HH24:MI:SS')
,       resp.discrepancy_indicator
,       resp.validation_status
,       dcmq.question_name
,       decode( dcmq.enterable_flag
        , 'Y', 'E'
        , decode( dcmq.derived_flag
          , 'Y', 'D'
          , '-'
          )
        ) E
,       resp.value_text
,       rdcm.patient
,       rdcm.document_number
,       rdcm.visit_number
,       known.CLINICAL_STUDY_ID
,       known.response_id
,       resp.response_entry_ts
from    responses           resp
,       dcm_questions       dcmq
,       received_dcms       rdcm
,       responses           known
where   resp.received_dcm_id       = known.received_dcm_id
  and   resp.dcm_question_group_id = known.dcm_question_group_id
  and   resp.repeat_sn             = known.repeat_sn
  and   rdcm.received_dcm_id       = known.received_dcm_id
  and   rdcm.dcm_subset_sn         = dcmq.dcm_que_dcm_subset_sn
  and   rdcm.dcm_layout_sn         = dcmq.dcm_que_dcm_layout_sn
  and   dcmq.dcm_question_id       = resp.dcm_question_id
  and   rxc_tms_access.studyAccess(known.clinical_study_id) = 1
order by resp.response_id, resp.response_entry_ts;

grant select on dmo_dcms to rxclin_read;
grant select on dmo_dcms to tms with grant option;
exec opa_ddl.createDropPublicSynonym ('dmo_dcms');
```

Define Functions in the External System

Functions can derive larger data points than TMS can handle in drill-down views. Using a function of type varchar2 enables you to return up to 32k of information back to TMS.

Each function can derive one column detail.

Following is the package body for a sample function. Note that the function must include all the parameters shown.

Example 5–3   Sample External System Drill-down Function

```
CREATE OR REPLACE PACKAGE BODY dmo_project IS

  FUNCTION showData(
          pSourceTermId    IN NUMBER
        , pOccurrenceId    IN NUMBER
        , pDefInstanceName IN VARCHAR2
        , pIntegrationKey  IN VARCHAR2
```

```
                       , pDictContentId    IN NUMBER
                       , pUpdateFlag       IN VARCHAR2
                       , pDefDomainId      IN NUMBER
                       , pXArea            IN NUMBER
                       , pSTAltKey         IN VARCHAR2
                       , pXV1              IN VARCHAR2
                       , pXV2              IN VARCHAR2
                       , pXV3              IN VARCHAR2
                       , pXV4              IN VARCHAR2
                       , pXV5              IN VARCHAR2
                       , pXV6              IN VARCHAR2
                       , pXV7              IN VARCHAR2
                       , pXV8              IN VARCHAR2
                       ) RETURN VARCHAR2 IS
     rVal      VARCHAR2(2000) := NULL;
   BEGIN
     IF rxc_tms_access.studyAccess(pXArea) = 0 THEN
       rVal := 'You do not have access to this data';
     ELSE
       SELECT opo.program_code
               ||': '||opo.description
               ||'  -  '||opr.project_code
               ||': '||opr.description
       INTO    rVal
       FROM    ocl_programs opo
       ,       ocl_projects opr
       WHERE   opr.project_code = pXV1
         AND   opo.program_code = opr.program_code
       ;
     END IF;
     RETURN rVal;
   EXCEPTION
     WHEN OTHERS THEN
       RETURN 'The data could not be found in Oracle Clinical';
   END showData;

END dmo_project;
/
```

Setting Up External System Drill-down Queries

The **Queries** tab enables you to define external system drill-down queries, which are groupings of external system columns. You must define at least one query, or the **Source Data** tab will not be displayed in the HTML browser.

By using a query when you search for source data in the HTML browser, you can focus searches to include only the external system columns you need.

A query is specific to one external system, and will appear only when you choose that external system in the HTML browser's **Source Data** tab.

This section includes:

- Defining the Name and External System for a Query on page 5-36
- Defining Query Columns on page 5-37

Defining the Name and External System for a Query

Each query you define appears in the **Group** list for source data searches in the HTML browser.

To define a query:

1. In the External System block of the Define External Systems window, click the external system for which you want to define a query.

2. Click the **Queries** tab and in the upper block of that tab, either click in an empty row or insert a record.

3. Describe the query:

    ■ **Short Name** – Unique short name for the query.

    ■ **Name** – The display name for this query. This name displays in the **Group** list for source data searches in the HTML browser.

    ■ **Description** – Optional description about this query.

4. Save.

You can change the display name and description at any time. Proceed to "Defining Query Columns" on page 5-37 to add the columns that compose this query.

Defining Query Columns

To define the columns that the external system query will display:

1. In the Query Columns block of the **Queries** tab, click in an empty row or add a record.

2. Enter and describe each external system column you want to include in the query:

    ■ **Map ID**. The unique ID of the external system column. The Map ID numbers and external system columns are available on the **Attributes** tab.

    ■ **Col Order**. The order, from left to right, in which the HTML browser will display the columns for this query. Columns with lower numbers appear in the query on the left.

3. Save. TMS makes this query available upon your next login to the HTML browser.

6

Defining and Loading Dictionaries

Oracle Thesaurus Management System (TMS) enables you to design, define, and load dictionary data into its repository.

You perform all dictionary definition and loading tasks from the master or single TMS instance.

You must have the tms_define_priv role to define dictionaries and domains. See Chapter 3, "Administration" for information on granting this role to users.

This section includes:

Dictionary Types

You can define the following types of dictionaries:

Strong Dictionaries

A strong dictionary definition in TMS consists of a fixed number of hierarchical levels with defined relations between the levels. You store every dictionary term in a particular level and define relations from each term to terms in other levels, according to the relations you have defined for the levels themselves. This type of TMS

dictionary definition is appropriate for external dictionaries such as MedDRA and WHO-Drug.

Strong dictionaries can serve as the base for creating virtual dictionaries and filter dictionaries; see "Virtual Dictionaries" on page 6-3 and "Filter Dictionaries" on page 6-4.

For more information, see "Planning Strong Dictionary Structure" on page 6-9.

Weak Dictionaries

Some external dictionaries, such as SNOMED, require a different type of structure, one that is dynamically created by relations between terms. TMS supports these with *weak* dictionaries and *named relations* between terms.

Where strong dictionaries have fixed hierarchies, represented by levels, weak dictionary have dynamic hierarchies, which are variable. For example, a strong dictionary with four levels from the coding level to the highest derivation level will only have four terms in a hierarchical chain:

Level 1 - Term 1
Level 2 - Term 2
Level 3 - Term 3
Level 4 - Term 4

A weak dictionary does not depend on dictionary levels to identify the hierarchy. The hierarchical structure of the dictionary is defined by the relationships permitted and used in the dictionary. Named relations describe the connection between terms. For example, you can use the named relation **Narrower term** to link the terms "aspirin" and "pain reliever," or the named relation **Is part of** to define a hierarchical series of relations as follows:

Term 1 - is part of - Term 2
Term 2 - is part of - Term 3
Term 3 - is part of - Term 4
Term 4 - is part of - Term 5
Term 5 - is part of - Term 6

In the above example, the number of terms that can be linked in a hierarchical chain is not limited by the number of levels. In fact, all of the terms (1 through 6) are on the same level, and the length of the hierarchical chain is unlimited. Dictionaries with this characteristic are called *dynamic*.

TMS requires that each dictionary definition have at least one level to contain dictionary terms. If your dictionary is really a collection of dynamic dictionaries, you can define a TMS dictionary level for each component dictionary and fully define the dynamic dictionary within that level. (Do not define level relations as you do in strong dictionaries.) If your dictionary is a single dynamic dictionary, define a single dictionary level for it. You can define named relations between terms in the same or different levels of a weak dictionary, as business needs dictate.

> **Note:** You cannot code verbatim terms against weak dictionaries because they do not have a specific classification level.

See "Defining Named Relations" on page 7-16 for more information.

Virtual Dictionaries

Base dictionaries usually contain the latest dictionary and verbatim terms, because they are updated on an ongoing basis. These dictionaries are useful for studies that require the most up-to-date dictionary information.

By contrast, virtual dictionaries have defined cut-off dates, after which no dictionary terms can be added. Using a virtual dictionary enables you to continue coding verbatim terms for a clinical study against one version of a dictionary while the base dictionary changes. You can also use virtual dictionaries to view the state of a dictionary's terms and relations for any date in that dictionary's history. Virtual dictionaries can only be based on a single, active base dictionary, from which the virtual dictionary inherits its definition.

To set up a virtual dictionary in TMS, choose a base dictionary (and activate it, if the base dictionary is still provisional), define the virtual dictionary's name and cut-off date, and assign the virtual dictionary to a domain (see "Assigning a Dictionary to a Domain" on page 6-38).

1. Define and activate the base dictionary on which you want to base the virtual dictionary; see "Defining a Dictionary" on page 6-17 and set it to active; see "Setting the Dictionary's Status to Active" on page 6-27.

2. Load data into the base dictionary and activate the data; see "Loading a Dictionary" on page 6-29.

3. With the base dictionary displayed in the Base Dictionary tab of the Define Dictionaries window, click the Virtual Dictionaries tab.

4. From the **Record** menu, choose **Insert Record**. The system displays values for most fields that are inherited from the base dictionary. You can modify the Short Name, Name, and Description to make them more meaningful.

5. Define a **Cut-off Date**. All terms current at the point in time you specify will constitute the virtual dictionary. You cannot change the cut-off date after saving. Click in the Cut-off Date field. If other virtual dictionaries have been created for the same base dictionary, their cut-off dates are available in the LOV. If not, you can enter the date using the format:

    DD-MON-YYYY HH:MM:SS

    for example, 01-NOV-2006 10:01:15

    You can enter the date portion only and the system will enter a time of midnight.

6. If a Release Label has already been associated with the timestamp you selected, the system displays it in the **Release Label** field. You can modify it if necessary. **Save**.

7. When you are satisfied with your virtual dictionary definition, set its **Status** to Active and **Save**.

8. You can then assign an Informative Note of type Dictionary Version to the virtual dictionary by clicking the **Info Notes** button. See "Defining Dictionary-wide Informative Notes" on page 7-26 for instructions.

To browse the state of a dictionary at a particular point in time, adjust the data currency to view the terms that were current in a dictionary at any date in the dictionary's history. To view such a representation of the dictionary, launch the Browse Repository Data form and enter the date in the appropriate field at the top of the window. For more instructions on browsing the repository, see Chapter 14, "Using the HTML Browser."

Filter Dictionaries

This section includes:

-

-

-

-

-

-

About Filter Dictionaries

Standardized MedDRA Queries (SMQs) are designed to aid in the identification and retrieval of case safety reports in clinical studies. The MedDRA Maintenance and Support Services Organization (MSSO) defines SMQs as follows:

SMQs are groupings of terms from one or more MedDRA System Organ Classes (SOCs) that relate to a defined medical condition or area of interest. They are intended to aid in case identification. The included terms may relate to signs, symptoms, diagnoses, syndromes, physical findings, laboratory and other physiologic test data, etc., related to the medical condition or area of interest. Lowest Level Terms (LLTs) that are not subordinate to an included Preferred Term (PT) are excluded.

See Figure 6–1, "Acute Pancreatitis SMQ Terms and Algorithm" for an example of a MedDRA SMQ.

TMS provides filter dictionaries as a generic way to support SMQs and potential similar features for other standard dictionaries. You define, load, and maintain a filter dictionary in the same way you define, load, and maintain a base dictionary such as MedDRA. You must define a filter dictionary as a strong dictionary with at least one level and link the filter dictionary to one and only one base dictionary. In addition, before you load MedDRA SMQs you must define broad and narrow named relations and algorithm Informative Notes (see below).

You can use the HTML Browser to perform SMQ (filter dictionary) searches to retrieve dictionary terms and the VTAs and source terms related to them.

MedDRA SMQs may use one or more of the following design features:

- **Broad and Narrow**. SMQ relations with dictionary terms may be identified as *broad* or *narrow*. Narrow terms are highly likely to represent the condition of interest while broad terms help to identify all possible cases, including some that may prove to be of little or no interest. A broad search retrieves broad terms. A narrow search retrieves narrow terms.

  To support broad and narrow SMQ searches in TMS, define and load broad and narrow named relations from SMQ terms to base dictionary terms.

- **Algorithm**. Algorithms within SMQs are used to further define combinations of specific terms based on rules defined in an algorithm that can be processed programmatically during a search operation.

  To support SMQ algorithms, define an Informative Note of type Algorithm for SMQ terms with algorithms; see "Defining Informative Notes for SMQ Algorithms" on page 6-7.

MSSO provides predefined categories of terms called A…F. You must load these categories with SMQ filter dictionary terms and store them in the Status column of the table TMS_DICT_RELATIONS. Some SMQ algorithms use these categories.

■ **Hierarchy**. Some SMQs are a series of queries related to one another in a hierarchical relationship similar to the hierarchical structure of MedDRA itself.

To support hierarchical SMQ searches in TMS, load SMQ terms into different levels of the filter dictionary with relations between the terms.

Figure 6–1   Acute Pancreatitis SMQ Terms and Algorithm

Narrow Terms	Signs and symptoms relevant for pancreatitis (Broad)	Lab values relevant for pancreatitis (Broad)
Cullen's sign	Abdominal distension	Blood amylase abnormal
Hereditary pancreatitis	Abdominal pain NOS	Blood amylase increased
Oedematous pancreatitis	Abdominal pain upper	Lipase abnormal
Pancreatic abscess	Abdominal rebound tenderness	Lipase increased
Pancreatic haemorrhage	Abdominal rigidity	Lipase urine increased
Pancreatic necrosis	Abdominal tenderness	Pancreatic enzyme disorder NOS
Pancreatic phlegmon	Acute abdomen	
Pancreatic pseudocyst	Ascites	Pancreatic enzymes NOS abnormal
Pancreatic pseudocyst drainage	Blood bilirubin increased	
Pancreatitis acute	Bowel sounds abnormal	Pancreatic enzymes NOS increased
Pancreatitis acute on chronic	Fat necrosis NOS	
Pancreatitis haemorrhagic		Urine amylase abnormal
Pancreatitis necrotising	Gastrointestinal pain NOS	Urine amylase increased
Pancreatitis NOS	Haemorrhagic ascites	
Pancreatitis relapsing	Hyperbilirubinaemia	
Pancreatorenal syndrome	Ileus paralytic	
	Jaundice NOS	
	Nausea	
	Peripancreatic fluid collection	
	Vomiting NOS	
	Vomiting projectile	

A case of Acute Pancreatitis = Narrow term OR (Sign/Symptom AND Relevant Lab Value)

Virtual Dictionaries of Filter Dictionaries   You can create a virtual dictionary of a filter dictionary—that is, a filter dictionary at a specific point in time. The filter dictionary's virtual dictionary must have the same cut-off date and time as an existing virtual dictionary of its base dictionary.

For example, given the following dictionary definitions:

■ MedDRA SMQ is the filter dictionary of MedDRA

■ MedDRA Version 7 is a virtual dictionary of MedDRA with the cut-off date 8/1/2006 00:00:00

you can define a virtual dictionary called MedDRA SMQ Version 7 of MedDRA SMQ with the same cut-off date as virtual dictionary MedDRA Version 7: 8/1/2006 00:00:00. It will reference terms in virtual dictionary MedDRA Version 7.

In other words, if you are using a virtual dictionary for coding and you want to use a filter dictionary with it, you must create a virtual dictionary of the filter dictionary with the same cut-off date as the virtual dictionary you are using for coding.

Using MedDRA SMQs in TMS

To use MedDRA SMQs in TMS, do the following:

1. Define a filter dictionary for MedDRA SMQs; see "Filter Dictionary Rules" on page 6-6.

2. Specify its base dictionary in the **Dictionary Links** tab, using the link type **Filter Dictionary Of**. For the MedDRA SMQs filter dictionary, the base dictionary must be a MedDRA dictionary. See "Defining Filter Dictionary Links" on page 6-37.

> **Note:** Filter dictionaries inherit their domain assignments from their base dictionary.

3. Define two named relations of type Standard with Many Cardinality, each with no reciprocal relation defined, to support broad and narrow relationships. Include the exact short name you define for these named relations in your MedDRA SMQs load script; see "Defining Named Relations" on page 7-16.

   Include the DEF_NAMED_REL_ID of these named relations in your MedDRA SMQs load script.

4. For the broad and narrow named relationships you have defined, create a dictionary named relationship record from the filter dictionary to the base dictionary; see "Defining Dictionary NRLs" on page 7-20.

5. Define Informative Note attributes of type Algorithm for SMQ algorithms; see "Defining Informative Notes for SMQ Algorithms" on page 6-7..

6. Create a load script and load MedDRA SMQs; see "Loading a Dictionary" on page 6-29.

   You can use the load script to load Algorithm-type Informative Notes and Broad and Narrow named relations or define them manually; see "Defining Informative Notes for SMQ Algorithms" on page 6-7 and "Creating Named Relations Between Terms" on page 12-20.

7. Use the HTML Browser to search for filter dictionary terms and their related base dictionary terms; see "Searching for Terms Using Filter Dictionaries" on page 14-11.

Filter Dictionary Rules

See "Defining a Dictionary" on page 6-17 for detailed instructions on how to define a dictionary. Select **Filter** as the **Dict Type** and observe the following rules for filter dictionaries:

- **Language**. Filter dictionaries and their base dictionaries must have the same language attribute setting.

- **VT Level Required**. Uncheck. Filter dictionaries cannot have a VT level.

- **Folder Type**. Filter dictionaries must be designated as Strong type.

- **Dictionary Link**. A filter dictionary must have one and only one "Filter Dictionary Of" dictionary link to a base dictionary. This dictionary link defines the fact that the filter dictionary will contain only terms defined in the base dictionary.

  A base dictionary can have one and only one "has Filter Dictionary" link to a filter dictionary.

- **Virtual Dictionaries**. Filter dictionaries can have associated virtual dictionaries. These "versions" of filter dictionaries must have the same cut-off date as an existing virtual dictionary of the filter dictionary's base dictionary.

- **Domains**. Filter dictionary domains are maintained by TMS. A filter dictionary inherits all the domains of its base dictionary. You can see a filter dictionary's domains in the Define Domain Dictionaries form, but you cannot change them.

Defining Informative Notes for SMQ Algorithms

Some MedDRA SMQs include algorithms that return refined search results. You can use these algorithms in TMS to search for source terms that report relevant symptoms or diagnoses; see "Searching for Terms Using Filter Dictionaries" on page 14-11.

To use SMQ algorithms in TMS you must do the following:

■ Create one Informative Note attribute of type Algorithm; see "Defining Informative Note Attributes" on page 7-22.

■ Create an Informative Note of type Algorithm for each SMQ filter dictionary term with an algorithm defined, either manually (see "Creating Informative Notes for Terms and Relations" on page 12-24) or as part of the load script; see "About Load Scripts" on page 6-29.

To support MedDRA SMQs, include the algorithm defined by MSSO for the SMQ filter dictionary term, but transform the format of the algorithm so that TMS can execute it. For example, in MedDRA 9.0, the SMQ algorithm defined by the MSSO for "Acute pancreatitis (SMQ)" is:

```
A or (B and C)
```

where A…F are categories of terms predefined by MSSO. You must load these categories with SMQ filter dictionary terms and store them in the Status column of the TMS table TMS_DICT_RELATIONS.

To translate this into a format that TMS understands, replace "or" with "UNION", replace "and" with "INTERSECT", and add the column name of the relations table that stores the SMQ category:

```
status = 'A' UNION (status='B' INTERSECT status='C')
```

> **Note:** You can substitute your own algorithm if you prefer. However, each term can have only one Informative Note of type Algorithm.

Maintaining Filter Dictionaries

Maintain filter dictionaries as you do base dictionaries:

■ **Maintain Repository Data**. Use the Maintain Repository Data window to maintain terms within a filter dictionary. In the lower block, the **R. Status** column contains the SMQ category of the target term, which is used in SMQ algorithms; see "Modifying Repository Data in the Maintain Repository Data Window" on page 12-9.

■ **Repository Authoring**. Maintain named relations to base dictionary terms in the Repository Authoring window; "Using the Repository Authoring Window" on page 12-18.

From both windows you can click the **Informative Notes** button to navigate to the Maintain Informative Notes forms, where you can maintain the SMQ algorithm as well as other Informative Notes; see "Creating Informative Notes for Terms and Relations" on page 12-24.

Searching Filter Dictionaries Using the TMS HTML Browser

You can use the HTML Browser to do the following kinds of searches based on a filter dictionary term:

■ Search for VTAs related to the filter dictionary term or its child, grandchild, etc. terms.

- Search for source terms related to the filter dictionary term and its related VTAs. This functionality is also available using the API; see the *Oracle Thesaurus Management System Technical Reference Manual*.

- Use the filter dictionary term's named relations, such as Broad or Narrow, to search for base dictionary terms.

- Use the filter dictionary term's algorithm to search for source terms; see "Performing a Simple Source Data Filter Search" on page 14-14.

See "Searching for Terms Using Filter Dictionaries" on page 14-11 for information on all the above searches.

Nonunique Coding Level Dictionaries

In **WHO-Drug Format C**, many drug names occur more than once in the coding level. They are distinguished from one another by auxiliary information including the country where they are distributed, their pharmaceutical form (for example, tablets or coated tablets), and their strength (for example, the number of milligrams per tablet). In the TMS dictionary definition, each type of auxiliary information (country or strength, for example) is defined as a sublevel of the classification level.

Auxiliary information must be imported from the source data system for each source term as a definition-value pair consisting of a dictionary level and data value; for example, Strength and 500mg. Updates to auxiliary information should be treated as source term updates.

If you are using Oracle Clinical, you can set up question sets to handle multiple input question responses for each source term. In addition to providing auxiliary information about the drug, you can send information about the indication for which the drug is being taken. This is required in cases where the term must be classified differently depending on the indication, or where a different higher level term must be derived depending on the indication. See "Classifying High-Level Terms Based on Indication" on page 6-9,"Defining Question Set Variables" on page 4-9, and "Classifying Indication Omissions" on page 10-10.

The classification of a verbatim term with a specific set of auxiliary information values to a nonunique dictionary term is called a **verbatim term individual (VTI)**, similar to a verbatim term assignment (VTA) for classifications to unique dictionary terms. Both VTAs and VTIs are supported in the same dictionary. TMS handles VTIs similarly to VTAs, but with some differences. VTIs are always created as approved and are always specific to a domain—never global. Both VTAs and VTIs can be designated either misspelled or accepted.

TMS captures and stores sets of auxiliary information values, assigning an ID to each set and reusing it for terms in all dictionaries and domains that have the same combination of auxiliary information values. For example, the first time the source term Carbamazepine is classified to the dictionary term Carbamazepine with auxiliary values Netherlands, coated tablet, and 200mg, TMS creates an ID such as 1000 for the value collection. When a different verbatim term that shares the same values—Netherlands, coated tablet, 200mg—is classified to any drug, TMS uses the same auxiliary information ID for the new VTI. The ID is stored with its values in the Auxiliary Information table, and the ID is included in the Omissions and Source Terms tables.

Autoclassification cannot match a source term to a nonunique dictionary term until a user manually classifies it. After that, if the dictionary is defined to allow autoclassification with auxiliary information, it classifies all occurrences of the same verbatim term with the same auxiliary information to the same dictionary term.

MedDRA also supports classifying the same source term to different dictionary terms. For example, the source term "Migraine Headache" might sometimes be classified as "Headache" and other times as "Migraine."

Classifying High-Level Terms Based on Indication

Different instances of the same dictionary coding-level term may be classified to different high level terms—for example, Anatomical - Therapeutic - Chemical (ATC) terms in WHO-Drug Format C—based on the **indication** for which the drug was taken by a patient. In this case, multiple high-level terms are related to a single lower-level dictionary term with no primary path defined. TMS cannot derive the high-level term and creates an *indication omission*. After a user has manually classified the indication omission, TMS derives all current and future occurrences of the same term for the same verbatim term/indication combination.

In Oracle Clinical you can create a question set variable of type Indication to collect the indication; see "Defining Question Set Variables" on page 4-9. If you are using a different external source data system you must also import "Indication" and the indication value as a value pair to support this feature.

> **Notes:**
>
> - Users cannot classify an indication omission without an indication, and after a verbatim term is classified, there is no way to send an action message to the source system requesting information. So you must set up indication collection before classifying indication omissions.
>
> - Indication omissions are not highlighted in Oracle Clinical Data Entry as verbatim term omissions.
>
> - Indication omissions are not replicated. If TMS runs in a distributed environment, indication assignments may vary across instances. Users can run the Indication Assignments Report on each instance and compare the results.

Planning Strong Dictionary Structure

This section includes:

- Defining Relationships Between Dictionary Levels on page 6-10

- Grouping Levels on page 6-13

- Deriving Terms on page 6-16

- Enforcing the Structure on page 6-17

- Strong Dictionary Structure in Other Forms on page 6-17

TMS provides flexibility in designing dictionaries, including those supplied by vendors. You can design the dictionary structure, ranking levels hierarchically and/or creating levels at the same rank. You can define the cardinality and optionality of level relations, and specify a Primary Link or Primary Path Link between levels.

Basic strategies for organizing dictionary structure include:

- Ranking levels in a vertical hierarchy so that terms become more general and fewer in each higher level of the hierarchy, effectively organizing lower-level terms into categories.

- Adding levels horizontally, at the same rank, to provide supplementary information about terms, such as preferred names or indications.

- Assigning levels to group levels to promote efficiency and enforce Primary Links to more than one level (see "Requiring a Primary Link to a Group Level" on page 6-14).

- Defining the classification level as a group level that includes both the conventional classification level and a synonym level with a many-to-one relation to the classification level. In this case, TMS searches in both sublevels for a match to a verbatim term. If it finds a match in the synonym level, it creates the VTA to link directly to the related conventional classification level term. Because of this behavior, using group levels as classification levels greatly increases the likelihood that a match will be found without increasing the number of terms in the conventional classification level.

Review each of the remaining sections in this chapter to consider your dictionary needs before defining a TMS dictionary or writing load scripts.

Uniqueness of Terms

In addition to the flexibility with which many internal and vendor-supplied dictionaries define their hierarchies and relationships, dictionaries can also differ in the manner in which they handle a term's uniqueness. You can specify that terms be unique throughout a level or group level in a dictionary, throughout all levels of a dictionary, or choose not to enforce uniqueness of terms at all for a particular dictionary.

If you plan to use a dictionary for classification of verbatim terms, you must enforce uniqueness on the classification level. For other internal or vendor-supplied dictionaries, choose the degree of uniqueness based on how duplicate terms exist within the dictionary.

TMS evaluates uniqueness based on the "term upper" version of a dictionary term, in which TMS converts the dictionary term to all uppercase letters and removes all extraneous spaces.

Defining Relationships Between Dictionary Levels

To create strong dictionary relations in TMS, so that you can link terms in one level to terms in another, you must define relations between the levels in the dictionary structure.

During Activation, TMS checks all terms and relations against the relations defined for their dictionary levels. Terms and relations whose definitions conflict with the defined level relations cannot be activated. For example, if you have defined relations to more than one term in a level with a Single cardinality relation, the Activation job activates the first relation it processes and rejects the rest.

You must define the following attributes for relations between levels within a strong dictionary: Mandatory Relations and Cardinality. For relations with Many cardinality defined you can also define a Primary Link or, if the dictionary includes a group level, a primary path.

The relations you define between levels create the derivation path for verbatim terms associated with the dictionary. A valid derivation path enables TMS to derive one and only one term from each level above the verbatim term level, and send the derived terms to an external system such as Oracle Clinical (see "Derivation Path" on page 6-16). If Single cardinality is defined from the lower to the higher level, no

Primary Link is required because a term in the lower level can be related to only one term in the higher level, and TMS can derive that term. If Many cardinality is defined, so that a term in the lower level can be related to multiple terms in the higher level, you must require a Primary Link to identify which related term in the higher level is derived. See "Primary Links and Primary Path Links" on page 6-11.

Mandatory Relations

For each level relation you can require that a term in each level have a relation with a term in the other level, or define the relation as optional in either direction. For example:

The Relation Is Mandatory in One Direction but Optional in the Other Terms in Level A must have a relation to a term in Level B, but terms in Level B may or may not have a relation to a term in Level A.

Both Relations Are Mandatory Terms in Level B must have a relation to a term in Level A, and terms in Level A must have a relation to a term in Level B.

Both Relations Are Optional Terms in Level B may or may not have a relation to a term in Level A, and terms in Level A may or may not have a relation to a term in Level B.

Cardinality

The **cardinality** of a level relation controls the ratio allowed between terms in two levels and must be defined in both directions:

■ Can a single term in Level A be linked to more than one term in Level B?

■ Can a single term in Level B be linked to more than one term in Level A?

You can define a level relation as many-to-many, one-to-many in either direction, or one-to-one.

Primary Links and Primary Path Links

If you define a relation where a term in Lower Level B can be linked to many terms in Higher Level A (Many cardinality), you can specify that each term in Lower Level B must have a **Primary Link** to a single term in Higher Level A. If the relation is on the **derivation path**—levels from which TMS can derive terms to be sent to an external system—you must require a Primary Link. Otherwise, TMS cannot activate the dictionary.

When a Primary Link is required, it is possible to create a Domain Primary Link for a particular term to override its Global Primary Link within a particular domain.

When you define a group level, you can define primary relations from terms outside (and below) the group level to the group level in two different ways:

■ **Primary Link**. If you define a Primary Link to the group level, and Many cardinality is defined on the higher side of level relations, the lower-level term can have links to multiple terms on multiple levels in the group level, but only one link can be the Primary Link. The Primary Link can be between the lower-level term and a single term on any level within the group level.

■ **Primary Path Link**. If you define a Primary Path Link to the group level, a term in the lower level can have links to multiple terms in multiple levels within the group level, but it must have a Primary Link to a term in the lowest level within the group, and to a related term in the next highest level, all the way to the top level.

You must load or define the Primary Link for each lower-level term to a term in each of the group levels from the bottom up. TMS restricts your choices at each level to terms that are related to the Primary Link term in the immediately lower level. Thus, different lower-level terms can have a Primary Link to the same term in the lowest group level but different terms in the higher group levels (see Figure 6–2).

Primary Path Links are designed to accommodate MedDRA's structure.

See "Grouping Levels" on page 6-13 for background information on group levels.

Figure 6–2 shows derivation paths using Primary Link (on the left) and a primary path (on the right). In these examples, all relations between levels are many-to-many, so Primary Links are required for each relation in the derivation path.

■ The example on the left shows a dictionary structure without a group level, where a Primary Link is required between each level and the next higher level. Terms PT1 and PT2 are both defined with a Primary Link to the same term—HLT1—in the next higher level. Their derivation path is the same because the rest of the derivation path depends on the Primary Links defined for the higher-level terms: HLT1 has a single Primary Link to HLGT1, which has a single Primary Link to SOC1.

■ The example on the right shows a dictionary structure where the HLT, HLGT, and SOC levels are all included in a group level, but a primary path is defined between the PT level and the group level. In this case, terms PT1 and PT2 can be linked to the same term in the HLT level and yet be linked to different terms in the two higher levels. This model conforms to MedDRA's structure.

> **Note:** Note that there is a third possible structure. If you defined the same group level as in the example on the right—containing the SOC, HLGT, and HLT levels—and defined a Primary Link from the PT level to the group level instead of a Primary Path Link, then the terms PT1 and PT2 would each link to HLT1 but, if Many cardinality were defined from the HLT to the HLGT and from the HLGT to the SOC, their derivation path would end there. No higher level terms could be derived because no other Primary Links or primary paths would exist.
>
> If Single cardinality were defined from the HLT to the HLGT and from the HLGT to the SOC, and those levels were defined as reportable and their relations as derivable, the derivation path would continue all the way to the SOC level.

Figure 6–2   Derivation Path Examples for MedDRA
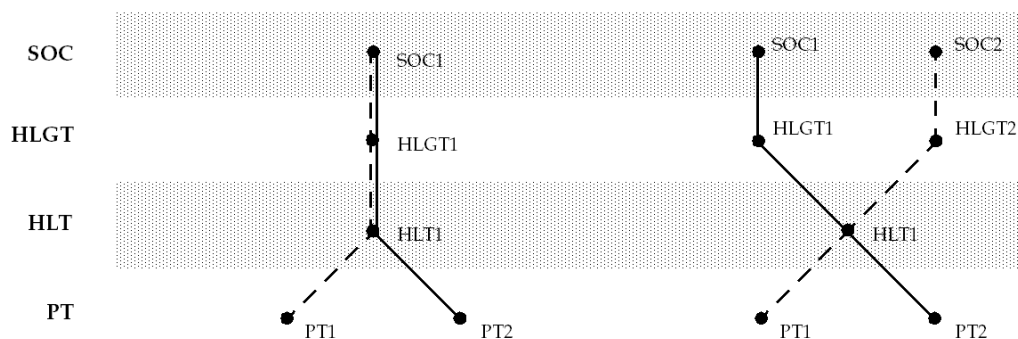
Grouping Levels

You can arrange levels of a strong dictionary into a **group level**, and define other levels as having a relation with the group level rather than one or more of its sublevels. You can then define individual terms as having relations with terms in sublevels within the group level. This enhances efficiency and functionality by:

■   Eliminating unnecessary duplication of terms in more than one level. By enforcing uniqueness of terms throughout the VTC group level in Figure 6–3, for example, you can ensure that no duplicate terms exist between the PN and SYN levels.

■   Allowing you to define a Primary Link requirement to the group level as a whole; when you define specific Primary Links for terms from outside the group level to the group level, you can choose to create the Primary Link to a term in any one of the group's sublevels.

For example, you could define WHO-Drug as shown in Figure 6–3, with two group levels, ATC and classification level VTC:

Figure 6–3   Group Levels in a Sample WHO-Drug Dictionary
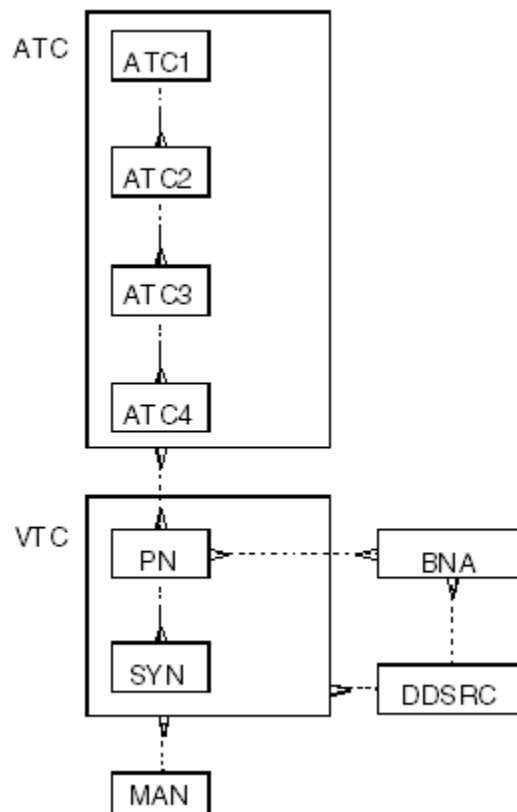
This definition of WHO-Drug makes use of the Group Level feature as described in the following two sections.

Classifying Verbatim Terms to a Group Level

By defining the **classification level** (the level in which TMS looks for a match to verbatim terms) as a group level, you can include more than one sublevel within the group for TMS to search. This increases the number of terms TMS can search without forcing the same terms to exist on multiple levels, which reduces the size of the dictionary. If you define a group level to be the classification level, you must specify that a term can exist in only one of the group's sublevels at a time. To enforce this structure, choose **Level** from the **Term Uniqueness** list when you define the group level.

In the WHO-Drug structure shown above, the classification level is defined as the group level called VTC, which contains as sublevels the Preferred Name level and the Synonym level. When TMS attempts to classify a verbatim term, it searches both sublevels, one after the other. Because TMS searches both levels to find a match, it is not necessary to include every preferred term in the Synonym level because TMS searches both levels to find a match.

Requiring a Primary Link to a Group Level

Grouping the dictionary's higher levels and requiring a Primary Link to the group level enables you to define a Primary Link for a specific term to any one of the group's sublevels. TMS derives the Primary Link term and related terms in any higher levels within the group if the following conditions apply:

- There is a many-to-one or one-to-one relation between each lower and higher level (Single cardinality is required at the higher end).

- The group level and all its sublevels are defined as reportable.

- The relations between the levels are defined as derivable.

Alternatively, you can define a primary path between the group level and a lower level. A primary path enables you to specify which terms to derive for each lower level term from each of the group sublevels (see "Primary Links and Primary Path Links" on page 6-11).

> **Note:** In addition, you must create structures in the external system to receive the derived values; in Oracle Clinical you need a question set variable and derived question for each sublevel that might contain the Primary Link term (see "Setting Up Data Collection in Oracle Clinical" on page 4-7).

Group Level Constraints

TMS enforces the following additional constraints for group levels.

- A level can be linked to a group level or a sublevel within it, but not to both.

- If two group levels are linked, none of the sublevels can be linked directly to sublevels in the other group.

- A group level cannot be a child level because that might cause problems during derivation. However, a group level can be lower in the hierarchy than other levels if the top sublevel (not the group level itself) has the child relation with the parent level.

Tree Structure Display of Group Levels

Group levels do not fit easily into a standard tree structure display. TMS displays them as follows:

- If a group level exists below the top of the hierarchy, its sublevels are displayed twice: once below the group level, which is displayed as a sibling level to the level above (because it cannot be defined as a child level), and once directly under the higher level with the relation between the top sublevel and the higher level displayed.

- The **Group Level** icon is displayed to the left of the group level.

- The relation icons (lines, dotted lines, and branches) appear in color for all sublevels within a group except the topmost, which by definition cannot have a conventional relation defined to the group level.

This display is for a sample WHO-Drug dictionary as shown in Figure 6–4.

Figure 6–4   Sample WHO-Drug Dictionary in the TMS Navigator Tree
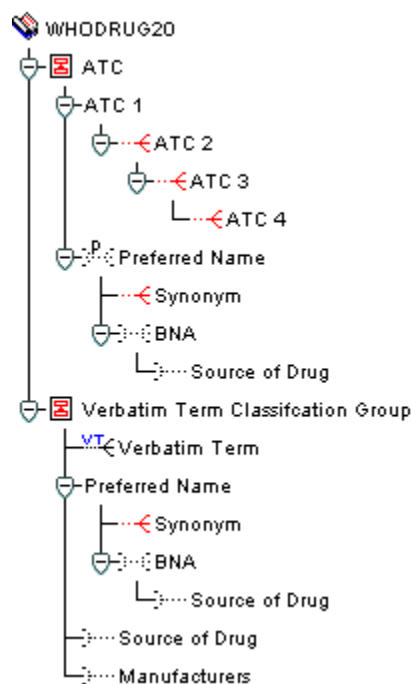
Deriving Terms

When TMS successfully maps a verbatim term to a dictionary term, it derives related terms or other user-defined information from the dictionary levels you specify when you set up the derived questions in Oracle Clinical question sets. See "Defining Questions" on page 4-11. When you use TMS with another external system, you must create the objects to receive the derived values and the mechanism to call for them. If TMS is fully integrated with the external system, it derives the values to the external system on demand.

In MedDRA, for example, you can classify the verbatim term *secondary anemia* to the classification level term *secondary anaemia* (LLT). In the dictionary definition, you can specify that the Preferred Term (PT) and System Organ Class (SOC) levels be derivable and reportable. If you have specified that TMS should derive a term, TMS returns the PT level term *secondary anaemia* and the SOC level term *blood and lymphatic system disorders*.

See the *Oracle Thesaurus Management System Technical Reference Manual* for more information on integrating TMS with external systems.

Derivation Path

In the TMS dictionary structure, you must define one and only one derivation path. A derivation path is a series of consecutively related levels defined as derivable that begins at the classification level and includes all levels from which you want to derive terms. There must be one and only one path from the verbatim term level to each derivable level.

Setting Up Derivation Within a Dictionary

Deriving terms from TMS into an external system such as Oracle Clinical requires setup activity in both TMS and the external system.

To set up derivation:

1. Select **Report Value?** in the Define Dictionary Level window to be able to derive terms from the level you are defining. The **Report Value?** setting is optionally used by the external system to display the levels available for reporting.

2. Select the **Derivable?** box in the Define Level Relations window for each level in the path of derivable levels. There must be one and only one pathway to derivable terms in any level from the classification level. All levels you define as reportable must also be derivable, but derivable levels do not necessarily have to be reportable.

3. Create objects to receive the derived values and the mechanism to call for them. This process varies according to the external system and the level of integration. For information on deriving terms for Oracle Clinical, see Chapter 4, "Integrating TMS with Oracle Clinical." For information on levels of integration, see Chapter 1, "Overview."

Enforcing the Structure

Each time TMS tries to activate a term, it runs several types of validation tests to ensure that the term and its relations conform to the dictionary structure you have defined. During this process, TMS:

- Checks if the term and its relations to terms in other levels do not violate the cardinality and optionality specifications you have defined between levels.

- Runs any additional validation code to enforce further rules and relations (see Step 8, "Add Validation Code (Optional)" on page 6-32).

- Prevents deletion of terms linked to a verbatim term assignment.

Strong Dictionary Structure in Other Forms

TMS reflects the structure you define in the way it displays the dictionary in the Maintain Repository Data and Browse Repository Data windows. See "Using the Tree Structure" on page 2-4 for details on how TMS reflects dictionary structure in these windows.

Defining a Dictionary

To use a dictionary in TMS you must define it in the TMS user interface. This section includes:

- Defining the Basic Dictionary Settings on page 6-17

- Defining the Dictionary Levels on page 6-20

- Defining Relations Between Levels (Strong Dictionaries Only) on page 6-22

- Defining Level Details on page 6-25

- Setting the Dictionary's Status to Active on page 6-27

Defining the Basic Dictionary Settings

The Base Dictionary tab of the Define Dictionaries window enables you to define the high-level settings of a dictionary and to set a dictionary's status from Provisional to Active. This section discusses definition only, because you must keep the dictionary status set to Provisional while you complete the definition tasks. You use the Base

Dictionary tab to set a dictionary to Active status later on. See "Setting the Dictionary's Status to Active" on page 6-27.

The high-level dictionary definition includes:

- The dictionary's name, short name, and description.

- Whether the dictionary is a base dictionary, virtual dictionary, or filter dictionary; see "Virtual Dictionaries" on page 6-3 and "Filter Dictionaries" on page 6-4.

- The dictionary structure (strong or weak); "Strong Dictionaries" on page 6-1.

- The language of the dictionary's terms and relations.

- The Release Label prefix.

- Other settings that drive aspects of dictionary structure and the dictionary's availability to other components in TMS.

Appendix A, "Sample Dictionary Definitions" contains the definitions you must use in order to successfully load and activate practice dictionary CLS.

To define the basic dictionary settings:

1. From the **Definition** menu, select **Define Dictionaries**. The Define Dictionaries window opens, with the Base Dictionary tab selected.

2. Choose whether you want to create a base, filter, or virtual dictionary.

   - For a base or filter dictionary, begin by highlighting the **Dictionaries** line in the tree structure and select **Insert Record**.

   - For a virtual dictionary, select an active base dictionary folder in the tree and click the **Virtual Dictionary** tab. Select **Insert Record**. See "Virtual Dictionaries" on page 6-3 for more information.

3. Define the dictionary itself (Name, Short Name, Description).

   Notes:

   - Special characters are not allowed in the Short Name definition.

   - The short name must be exactly the same in the GUI definition and in the load script. Sample load scripts are available on My Oracle Support. See Appendix A, "Sample Dictionary Definitions."

   - To delete a dictionary, you must place the cursor on the right side of the window in the **Dictionary** field (not the **Dictionary Name** in the tree). Before you can delete the dictionary itself, you must delete all the dictionary data, its levels, and set it to Provisional status.

The following steps apply to base dictionaries only. Virtual dictionaries inherit all settings from their base dictionaries.

1. From the **Language** list, select the language for this dictionary. You can link terms in dictionaries that have different languages by using named relations of type Translation.

2. From the **Dict. Type** list, select one of the following:

   - **Base**. Select Base to define a dictionary published by an external organization (such as MedDRA or WHO-Drug, or a legacy dictionary developed by your company.

- **Filter**. Select Filter to use MedDRA SMQs or a similar standardized query dictionary; see "Filter Dictionaries" on page 6-4 for more information.

3. From the **Folder Type** list, select **Strong** to create a strongly defined dictionary or **Weak** to create a dictionary folder.

4. Leave the **Status** set to **Provisional** until you are ready to activate and use the dictionary; see "Setting the Dictionary's Status to Active" on page 6-27.

5. Enter a **Label Prefix**. Accept the default value of 1. or enter the prefix of your choice. TMS automatically concatenates this value with a build number that represents the number of times Activation has been run on the dictionary. If you use a numerical prefix, you may want to end it with a period/full stop (.) to separate the prefix from the build number.

   You may want to use additional decimal places to accommodate point releases of the same dictionary, and you may want to use the dictionary's official release number.

   For example, if you define a label prefix of 3.0. when you initially load WHO-Drug 3.0, TMS assigns a Release Label of 3.0.1 to the successfully activated terms and relations the first time you run Activation on the Activation Group, 3.0.2 the second time, and so on. You can use Release Labels, including the label prefix, to define named relations of type Release Label (RL); see "Defining Named Relations" on page 7-16.

   You can also use a text prefix such as MEDDRA, or a prefix that combines text and numbers, such as MEDDRA7.0.

6. Assess the following options for your dictionary, selecting the appropriate boxes:

   - **VT Level Required?** For strong dictionaries only. If selected, you must define a classification level or classification group level in your dictionary—which, in turn, creates a Verbatim Term level—before you can activate this dictionary. In the TMS modules that deal with verbatim terms (every module under the VTA Maintenance and Omission Management nodes, as well as Browse VT Classification Data), TMS populates the list of values in the **Dictionary** field from the set of dictionaries that have a **VT Level Required?** selected.

   - **Web Search Accessible?** This setting is obsolete.

   - **Accessible to Light Browser?** Can the dictionary be used at all in the HTML Browser? If selected, TMS users will be able to browse and search this dictionary's terms, relations, and VTAs in the HTML Browser.

     Before activating this option, consider this dictionary' usage and the sensitivity of its data. The HTML Browser may be set for Auto-login, meaning that users do not have to enter a user name and password to browse accessible data. See "Customizing the TMS HTML Browser URL to Support Additional Databases or Automatic Login" on page 3-49.

     Do check this flag for filter dictionaries to enable filter dictionary searches using MedDRA SMQs.

   - **Term Uniqueness Enforced?** If selected, terms in this dictionary must be unique over the scope of the entire dictionary. When you choose to enforce uniqueness for an entire dictionary, you in turn enforce term uniqueness on all levels and group levels within that dictionary.

   - **VTI Allowed?** Check to enable creating Verbatim Term Individual VTI) classifications. This is required if nonunique terms are allowed in the coding

level, either with auxiliary information as in WHO-Drug C, or with different higher level relations, as you may have in MedDRA.

If checked, VTIs can be allowed or not allowed for this dictionary in dictionary domains.

- **Autocode with Aux?** Check to allow the TMS autoclassification process to create VTIs based on existing VTIs or dictionary terms with the same auxiliary information. Use this option if you are using a dictionary with nonunique terms in the classification level and auxiliary information for each term, such as WHO-Drug C.

   If checked, Autocoding with auxiliary information can be allowed or not allowed for this dictionary in dictionary domains.

- **Autoqueried in Light Browser?** Is the dictionary available for browsing in the Exploration/Hierarchies tab of the HTML Browser? If selected, the HTML Browser automatically queries for all terms in the top level of the dictionary when you select the dictionary in the main HTML Browser window. This setting is appropriate only for strong dictionaries (dictionaries with hierarchical levels defined).

   Not recommended for dictionaries in which most or all of the terms are stored in the highest or only level, because autoqueries in such dictionaries can be very time- and resource-consuming.

   > **Note:** The Terminologies tab in the HTML Browser appears only if both **Autoqueried in Light Browser?** and **Accessible to Light Browser?** are checked for at least one of the dictionaries linked to the current domain.

7. **Dictionary Term Display Procedure** If desired, specify a procedure in this field to modify the behavior of the **Dictionary Term** field in VT classification and reclassification. By default, TMS displays the dictionary term to which the selected verbatim term is classified, if any. By writing a procedure and specifying it for a dictionary, you can display a different term in these fields.

   For details about writing a Dictionary Term Display Procedure, see "Writing a Procedure to Change the Dictionary Display" on page 3-39. Because this feature is only relevant for verbatim term classification, enter a procedure in this field only for strong dictionaries with verbatim term levels.

8. Save. For a base dictionary, TMS inserts the new dictionary into the tree structure. Virtual dictionaries do not appear in the navigation tree within the Define Dictionaries module. In either case, TMS populates the audit information fields.

The Virtual Dictionary tab window can only display definition information for a single virtual dictionary. If you define multiple virtual dictionaries against a base dictionary, you can scroll through the records to find the virtual dictionary you want to modify. From the **Navigate** menu, select **Previous Record** or **Navigate**, then choose **Next Record** to navigate to your choice. Alternatively, use the arrows on your keyboard.

Defining the Dictionary Levels

The Level tab of the Define Dictionaries window enables you to define a dictionary's levels. For strong dictionaries, begin by defining the topmost level in the dictionary hierarchy and work your way down. For weak dictionary folders, define all of the dictionaries you want to add to this dictionary folder. To delete a level, see "Deleting a

Level" on page 6-22.

To define dictionary levels:

1. Click the Base Dictionary tab. (You cannot add levels to virtual dictionaries.)

2. Click the name of your selected dictionary in the navigator tree.

3. Select **Record**, then **Insert**.

   The Level tabs replace the Dictionary tabs, and the new level definition appears in the Level tab window and in the navigator tree. If you are defining a weak dictionary folder, the fields in the Level Relations tab are grayed out.

4. Define the short name and name of the level.

   > **Note:** The level names you define in the GUI must exactly match those you specify in the load script. See Appendix A for the level names of the practice dictionary CLS.

5. Select **Classification Level?** if this is the level against which you want TMS to map verbatim terms. You can select classification level for only one level. If you use a group level for the classification level, you must also choose **Level** from the **Term Uniqueness** list.

   If you define this level as the classification level, TMS automatically creates a level under it called Verbatim Term Level where it will store verbatim term assignments (VTAs). TMS displays a blue "vt" above the relation line in the tree structure to denote that the level is the Verbatim Term Level.

   You can define classification levels for strong dictionaries only.

6. Select **Report Value?** if you want TMS to report terms of this level to the external system as derived terms when they are called for by the external system, as long as the relation between this level and the next lower level is on the derivation path. See "Derivation Path" on page 6-16 for more information.

   You can only create reportable levels in strong dictionaries.

7. In the **Level Order** field, enter a number only if this level is one of two or more levels at the same rank (children of the same parent level) in the dictionary hierarchy. Enter 1,2…*n* to specify the order in which TMS should display the levels vertically in the dictionary hierarchy tree structure. Levels with lower Level Order numbers appear higher in the tree structure. This has no functional effect; the setting is necessary only because you cannot display two levels in the same place in the tree structure. The field is enterable for all levels, but TMS only uses it to determine the presentation order of peer levels.

8. In the **Level Type** list, choose **Level** or **Group Level**. A **group level** is a level that contains more than one level within it. You must define a group level before you define the levels within it. See "Group Level Constraints" on page 6-15. If you are defining a group level, leave the **Group Short Name** field blank. To define other levels as sublevels within the group level:

   - Level Definition – Give sublevels a Level Type of Level and enter the group level's short name in the Group Short Name field.

   - Relation Definition – In the Level Relations window, define the top sublevel's relation with the parent group level as a Relation Type of Top Sublevel. See "Defining Relations Between Levels (Strong Dictionaries Only)" on page 6-22.

Define subsequent sublevels as having a **Relation Type** of **Level** in relation to the next higher sublevel.

You can only define group levels in strong dictionaries. See "Group Level Constraints" on page 6-15 and "Tree Structure Display of Group Levels" on page 6-15.

> **Note:** The **Verbatim Term Level** setting is for system use only; TMS enters this value automatically when it creates the VT Level. Levels that are neither group nor verbatim term levels should be called simply *level* here, including sublevels within a group.

9. **Term Uniqueness**: If set to Level, terms within this level must be unique. If set to None, there can be multiple occurrences of the same term in this level. Each occurrence may have different auxiliary information.

10. If the level you are defining is one of the sublevels within a group, enter the short name of the group level in the **Group Short Name** field. Leave this field blank for all other levels, including the group level itself.

11. Save. TMS inserts the level into the tree structure, and populates the audit information fields.

12. To define subsequent levels for a strong dictionary, you can either:

    ■ Highlight the parent level in the tree structure and select **Record** and then **Insert**. Then choose **New Level** from the **Level** or **Relation Creation** dialog box.

    ■ Highlight any field in the Levels window and select **Insert Record**. TMS creates a child level of the level you highlight.

    To define subsequent dictionaries for a weak dictionary folder, highlight either the dictionary folder name or any of its dictionaries, and add a record.

Deleting a Level

To delete a level, highlight the level short name in the Levels window and select **Delete Record**. You cannot directly delete a verbatim term level, but you can remove one indirectly by either deleting the classification level above it or clearing the **Classification Level?** box for the classification level above it. In either case, TMS automatically deletes the verbatim term level.

Defining Relations Between Levels (Strong Dictionaries Only)

You define relations between levels in the Level Relations window, one pair of levels at a time. The child level is highlighted in the tree structure. See "Group Level Constraints" on page 6-15 and "Tree Structure Display of Group Levels" on page 6-15.

> **Note:** You cannot define relationships between the levels, or microthesauri, in a weak dictionary folder, although you can define links between their terms. In such cases, the cardinality and optionality of the relation is defined by the named relation itself; see "Defining Named Relations" on page 7-16 for more information.

Level Selection in the Define Dictionaries Window

This window provides subtle visual cues to show you the currently selected dictionary level. Before defining your dictionary structure any further, be sure that you understand this user interface behavior and its consequences for dictionary definition.

There are two ways that the user interface displays levels as appearing to be selected:

■ The level name appears highlighted, as in the left example in Figure 6–5. TMS highlights levels when you click them for the first time.

■ The name appears highlighted and dashed lines are present above and below the level name, as in the right example in Figure 6–5. You can select a level by double-clicking any level, or single-clicking a highlighted level.

Figure 6–5   Highlighted and Selected Levels in Define Dictionaries
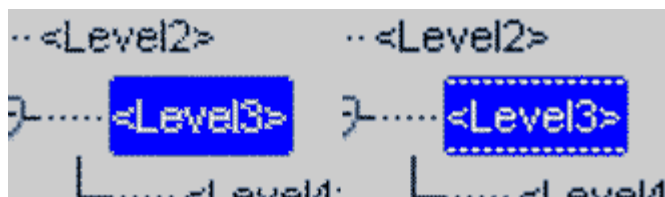
The difference between these selection states appears when you attempt to enter a new level into a dictionary. If you select **Insert** and then **New Record** when a level is highlighted, TMS inserts a child level under it.

However, if you insert a new record when a level is selected, TMS prompts you to choose between adding a new level under the selected level, or creating a new relation between the selected level and an existing level in the dictionary.

Defining Relations Between Levels

To define relations between levels of a strong dictionary:

1. Open the Level Relations window using one of the following methods, depending on:

   ■ If you are defining the relation between the level you have just created and its parent, click the **Level Relations** tab. TMS populates the parent and child level information.

   ■ If you are defining a relation between levels that already exist:

2. Highlight or select the parent level in the tree structure, click the **Level Relations** tab, select **Record**, and then **Insert**.

3. If you had selected (instead of highlighted) the level, TMS opens the **Level** or **Relation Creation** box. Select **New Relation to an Existing Level?** and click **OK**. TMS displays a list that includes all levels that do not currently have a relation with the parent level.

4. Select the level to which you want to define a relation and click **OK**. TMS enters the new relation in the tree structure: it displays the child level a second time with a relation line to the new parent level.

5. Click the new display of the child level in the tree structure and click the **Level Relations** tab. TMS populates the parent and child level information.

   For example, to create the standard definition of MedDRA you define a branch of related levels beginning with System Organ Class (SOC) and continuing to High

Level Group Term (HLGT), High Level Term (HLT) and Preferred Term (PT), each the parent of the next. However, you can link the PT level directly to the SOC level. To create this relation, after you have created the PT level under the HLT level, select (do not merely highlight) the SOC level in the tree structure, select **Insert Record**, select **New Relation to an Existing Level**, select the PT level from the **Child** LOV, and define the relation.

6. From the **Relation Type** list, choose either **Level** or **Group Level** (**Verbatim Level** is for TMS system use only).

■ Choosing **Top Sublevel** instructs TMS to create the new level as the top sublevel within the parent level, which must have a **Level Type** of **Group Level** in the Level window. See "Group Level Constraints" on page 6-15. Top sublevels cannot have any other relation type specified; you cannot select any of the other flags in this window.

■ Choosing **Level** instructs TMS to create the new level as a child of the parent level. If the parent level is the top sublevel within a group, TMS creates the child as a lower sublevel within the group. If the parent level is a group level, the child level will be linked to the group level as a whole and cannot be linked directly to any of the sublevels within it.

7. Define the parent side of the relation. Select or leave deselected each box as it relates to the parent side of the relation.

It may help to diagram the relation, indicating whether each end is either multi or single, and either mandatory or optional.

For example, to define the relation shown below, leave both the **Mandatory?** and **Many Cardinality?** boxes clear for the parent level, and select both those boxes for the child level.
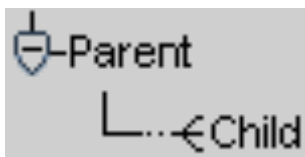
See Appendix A, "Sample Dictionary Definitions" for sample dictionary diagrams and the corresponding definitions.

■ **Mandatory?** Select if terms in the parent level must have a link to a term in the child level.

■ **Many Cardinality?** Select if terms in the child level can have links to more than one term in the parent level.

■ **Primary Link?** Select if **Many Cardinality?** is selected for the parent level and you want to require that one of the terms in the parent level be defined as the child level term's Primary Link. If **Derivable?** is also selected for the parent level, you must select either the **Primary Link?** or **Primary Path Link?** boxes. Otherwise, TMS will not activate the dictionary.

■ **Primary Path Link?** Select this to define a Primary Path Link between the selected parent and child levels.

■ **Derivable?** There must be one and only one pathway to derivable levels from the classification level. Select if you want this relation to be part of that pathway.

> **Notes:**
>
> - To derive terms from a dictionary level, the level must be on the derivation path (see "Derivation Path" on page 6-16), the level's **Report Value?** box must be selected, and the external system must be set up to request derived values from that level. For example, see "Setting Up Data Collection in Oracle Clinical" on page 4-7.
>
> - You cannot modify the automatically defined relation between the verbatim term level and the classification level. On the child side, **Mandatory?** and **Many Cardinality?** are selected; on the parent side, only **Derivable?** is selected.

8. Define the child side of the relation. Select or leave clear each box as it relates to the child side of the relation.

    a. **Mandatory?** Select if terms in the child level must have a link to a term in the parent level.

    b. **Many Cardinality?** Select if terms in the parent level can have links to more than one term in the child level.

> **Note:** It is possible to define relations between levels of different dictionaries as well as levels within a dictionary. For more information on linking terms between dictionaries, see "Setting Up Cross-Dictionary Relations" on page 12-22.

9. Save. TMS populates the audit information fields.

Defining Level Details

This section describes what level details are and how to define them.

- Understanding Level Details on page 6-25
- Defining a Dictionary's Level Details on page 6-26

Understanding Level Details

Level details are optional. They store and display additional, customized information about terms in a particular dictionary level. Details reference seven columns in the tms_dict_contents table. Level details enable you to customize these columns for your company's needs.

> **Note:** Level details and the validation configured within the Define Dictionaries module are only enforced and displayed from within the TMS Graphical User Interface. Batch loading of data does not adhere to these configurations.

TMS displays fields corresponding to Level Details in the Maintain Repository Data and Browse Repository Data windows. You define the field label, content type, and other characteristics for up to seven fields. These fields are:

Label The column heading that you want TMS windows to use for this column in this dictionary.

Category You can use this field to define a category for the term, such as 'Provisional' for terms as they go through the MSSO Dictionary Maintenance process.

Dict Content Code Corresponding to the dict_content_code column, this field is designed to serve as the primary key within this dictionary. This column is indexed.

Dict Content Alt Code By default, the **Dict Content Alt Code** has a function different from **Values 1…4**. It corresponds to the dict_contents_alt_code column, which is indexed, unlike the Value 1…4 columns. It is designed to allow you a cross-reference to a legacy thesaurus system.

Value 1, 2, 3, and 4 You can use **Value 1…4** to customize other fields in the Maintain and Browse Repository Data windows. For example, you could define that **Value 1** for the WHO-Drug Dictionary's Preferred Term level was "number of ingredients." See Step 3 below to define a list of values for the field.

> **Note:** You can modify level details even in an active dictionary.

Defining a Dictionary's Level Details

This section describes how to define a dictionary's level details using the Define Dictionaries window.

To define a dictionary's level details:

1. Click the **Level Details** tab. The Level Details window opens.

2. In the **Level Detail** field, choose a value from the list. See the explanations of these fields on page 6-26 for more information.

3. Specify the label of the field to appear on the Maintain Repository Data form.

4. Write a description of the purpose and/or content of the field.

5. Specify the following:

   - **Enterable?** Select if you want the user to be able to enter a value in the field to appear on the Maintain Repository Data form.

   - **Updateable?** Select if you want the user to be able to update a value in the field, following the initial insert of the term.

   - **LOV Validation?** Select if you want TMS to validate that one of the values from the LOV is entered in the field. If **Validation?** is selected and the user enters an invalid value in the Maintain Repository Data window, TMS does not allow the user to save the record.

   - **Mandatory?** Select if you want the user to be required to enter a value in the field. If **Mandatory?** is selected, and you do not enter a value in the Maintain Repository Data window, TMS does not allow you to save the record.

   - **Entry Length**. Enter the maximum number of characters the field can contain. TMS does not allow the user to save a longer value in the Maintain Repository Data window.

   - **Data Type**. From the list, select the type of data to be contained in the field: char, date, or numbers. In the Maintain Repository Data window, TMS does not allow the user to save the wrong data type.

6. In the **Description** field, define the purpose or content of the field.

7. In the **LOV Statement** field, write a select statement to determine the items to appear in an LOV for the field (optional). The statement must select two columns from the database, one of which is the value that will be stored in the database, and the other of which is the value displayed in the LOV.

   For example:

   ```
   select 'A', 'Approved' from dual

   union

   select 'P', 'Provisional' from dual
   ```

   > **Note:** Do not put a semicolon (;) at the end of the statement.

8. Save. TMS commits the level details to the database, and populates the audit information fields.

   To define another detail, put your cursor in the existing detail and select **Insert Record**.

Setting the Dictionary's Status to Active

When you are satisfied with all the dictionary's definitions, set its status to Active in the **Status** field of the Define Dictionaries window and save.

Once a dictionary is set to Active status, you may set its status back to Provisional only as long as it has no terms, TMS domains or VTAs associated with it, or if the dictionary has any Translation Derivation Links. However, you can still add levels to a weak dictionary folder even when it is set to Active status.

When changing the Active/Provisional status for a base or virtual dictionary, keep the following rules in mind:

- An Active base dictionary cannot be set to Provisional if an Active virtual dictionary based on it exists.

- A base dictionary set to VT Level Required cannot be set to Active unless it has one and only one verbatim term level.

- A Provisional virtual dictionary can be deleted at any time.

- A virtual dictionary with status Active cannot be deleted.

- The status of a virtual dictionary cannot be set from Active to Provisional if it is assigned to a domain.

- A base dictionary cannot be deleted if there are search objects associated with it.

- Changing the status of a base dictionary that has a virtual dictionary defined to provisional deletes the Provisional virtual dictionary.

The following table describes which activities are allowed when the dictionary is Provisional and when you set it to Active.

Table 6–1   Activities Allowed in Provisional and Active Dictionaries

Action	Provisional	Active
Modifying the dictionary short name	Yes	No
Changing the folder type (before any levels exist)	Yes	No

Table 6–1   (Cont.) Activities Allowed in Provisional and Active Dictionaries

Action	Provisional	Active
Changing enforcement of uniqueness of terms	Yes	No
Changing the **VT Level Reqd?** setting	Yes	No
Modifying the level hierarchy	Yes	For weak dict. folders only
Adding new level details	Yes	Yes
Modifying details	Yes	Yes
Modifying the dictionary name	Yes	Yes
Modifying the dictionary description	Yes	Yes
Modifying the dictionary Release Label prefix	Yes	Yes
Changing the level order	Yes	Yes

Defining Dictionary-wide Informative Notes Before Activation

See "Overview of Informative Notes and Attributes" on page 7-22 for general information about Informative Notes. You must define Informative Note Attributes and associate them with a dictionary before you can define dictionary-wide Informative Notes for the dictionary; see "Defining Informative Note Attributes" on page 7-22 and "Making Informative Note Attributes Available for Dictionaries and Record Types" on page 7-24.

This section describes adding dictionary-wide Informative Notes to the pre-dictionary tables, using the TMS API. This approach enables you to validate the notes you add by running Activation. You can also add dictionary-wide Informative Notes directly to the production tables by using the Define Dictionaries window (see "Defining Dictionary-wide Informative Notes" on page 7-26). The direct approach skips the Activation process.

Dictionary-wide Informative Notes serve one of two purposes in TMS, depending on the note type you choose:

- Notes of type Standard and URL cascade down to that dictionary's data. By defining a dictionary-wide Informative Note, you also assign that note to every data record within the dictionary.

- Dictionary version numbers enable you to specify which version of a dictionary you have loaded into the database. Dictionary versions are based upon the attribute Dictionary Version, which is derivable. See "Defining a Dictionary Version Informative Note" on page 7-28.

You cannot use the Workflow note type for dictionary-wide Informative Notes.

You can create dictionary-wide Informative Notes for either base or virtual dictionaries, but the dictionary must be active.

To add a dictionary-wide Informative Note to the pre-dictionary tables:

1. Connect to SQL*Plus as the load user.

2. Run the API procedure tms_user_mt_info.insertInfo.

   You must specify a pInfoNoteType of 'D' to create a dictionary-wide Informative Note.

Loading a Dictionary

This section includes:

- Required Setup on page 6-29
- About Load Scripts on page 6-29
- Downloading Sample Scripts on page 6-30

> **Note:** These instructions apply both to loading a dictionary for the first time and to upgrading a loaded dictionary to a new version. For more information on upgrading a dictionary, see Chapter 8, "Upgrading to a New Dictionary Version."

Required Setup

Before you load a dictionary, check that you have the necessary setup:

- Loading data into the temporary tables requires Oracle SQL Loader, which is part of your basic Oracle software package, and is located in the bin directory.
- Create a new user for loading data into TMS. In the TMS Define Users window, create a superuser and set the **Load User?** flag to **Yes**. For more information, see "Defining Users" on page 3-5.
- Define the dictionary structure, domains, activation groups, dictionary named relationships and dictionary informative notes attributes (instructions in this chapter and in Chapter 7, "Defining Other TMS Elements").
- Download the appropriate sample scripts from My Oracle Support and edit them for your installation. See "Downloading Sample Scripts" on page 6-30.

About Load Scripts

If you are loading data into TMS, a load script is necessary to move the data from temporary tables to the **predict tables** (pre-dictionary tables). Your load script must be consistent with your dictionary level and level relation definitions.

Oracle supplies sample load scripts for various dictionaries including MedDRA, MedDRA Primary Path, MedDRA SMQ, WHO-Drug B, B2, and C, and SNOMED on My Oracle Support. The corresponding dictionary structures are described in Appendix A, "Sample Dictionary Definitions."

> **Note:** This documentation is not intended to provide complete and specific instructions for the loading process, which may vary depending on the dictionary and on the individual needs of your location. Rather, these instructions are a guideline by which you can create load scripts that are compatible with your dictionaries. Oracle provides only sample load scripts that are not intended to dictate the usage or maintenance strategy for individual customers. These scripts are supported only to the extent that the API functions are specified.

Load scripts must do the following for each defined element that you are using:

> **Note:** Using named relations and Informative Notes is optional.

- Load dictionary terms into tms_predict_contents.

- Load relations between terms, including named relations, into tms_predict_relations.

- Load named relation definitions into tms_def_named_rels.

- Load mappings between named relationships and dictionaries into table tms_def_dict_nrls.

- Load Informative Notes into tables tms_predict_info_hdrs, tms_predict_info_strs, and tms_predict_info_clobs.

- Load Informative Note attributes into tms_def_details.

- Load mappings between Informative Notes and the items they describe or complement—dictionaries, terms, or relations—into table tms_def_dict_info_dets.

Load Script Requirements

Your load script must:

- Consist mainly of a series of select statements to transfer data from the temporary tables into the predict tables.

- Set the DML column value in the predict tables to indicate the transaction to be performed as Insert, Update or Delete for each term and relation.

- Make use of the constants in the tms_def_dict_cons package to avoid hardcoding constants.

- Call the necessary packages for all appropriate transactions. The load script for loading a dictionary for the first time only needs to allow for insert transactions, but the load script for updating a dictionary must allow for Insert, Update, and Delete transactions. You can use the same script for loading and updating if it allows for all three transactions. The information on the transaction required for each term and relation is contained in the ASCII files supplied by the vendor.

API Packages

All inserts, updates and deletes must be done using API calls rather than direct DML statements. Some of these packages are explained in "The API" on page 1-4, and full API descriptions are available in the *Oracle Thesaurus Management System Technical Reference Manual*, which is available from Oracle Support.

Load Scripts and Activation Groups

The sample load scripts work on one Activation Group at a time, but you can write one that loads more than one Activation Group at a time. By default, when you run the load script, all of the terms in the temporary tables are inserted into the Activation Group you specify. See "Creating or Assigning an Activation Group" on page 6-31.

Downloading Sample Scripts

Download instructions and sample scripts from My Oracle Support at https://support.oracle.com.

1. Sign in to My Oracle Support.

2. Search for Article ID: 258975.1; see "Searching by Article ID" on page xxii. The article is titled *Sample Maintenance Scripts for MedDRA, MedDRA Primary Path, MedDRA SMQ, WHO-Drug Formats B, B2, and C, and SNOMED*.

3. Scroll down to the Attachments section and download the file for the dictionary you want to load.

Activating Loaded Terms

The TMS Activation process runs against terms and relations in the predict tables that are associated with the Activation Group you choose. Activation includes two stages:

- TMS validates terms and relations against dictionary definitions. During this process, TMS validates pre-dictionary Informative Notes as well.

- TMS moves records that do not violate dictionary definitions to the production tables. The system leaves any records that do violate these rules in the pre-dictionary tables and populates the error message field for each record with the reason that the record failed activation.

You can run Activation in Check or Transfer mode. Check mode stops short of transferring data to the production tables while enabling you to see the results of the data validation. See "About Activation" on page 6-33.

You can run Activation from the GUI or from SQL*Plus. You can monitor the process and, if you run in Check mode, you can view data that would fail activation in the Maintain Repository Data window and make the necessary changes before transferring the data to the production tables.

Creating or Assigning an Activation Group

You must assign all data to an Activation Group for TMS to load it into predict tables and, later, to activate it. TMS activates all data in the group in the same batch job. You can use an existing group or define a new one. To define a new Activation Group:

1. From the **Repository Maintenance** menu, select **Maintain Repository Data**. The Maintain Repository Data window appears.

2. Highlight **Activation Groups** in the tree structure and select **Insert Record**.

3. Enter a name and description. (If you are setting up the practice drug dictionary CLS, name the Activation Group CLS_AG.)

4. Under **Dictionaries** within the Activation Group, click in the **Short Name** field and press F9 to see the LOV. Select the dictionaries you want to include in the Activation Group.

5. Save. TMS populates the **Name** and **In Domain?** fields. When you run Activation on this Activation Group, TMS processes all dictionaries associated with this group. However, if a dictionary is not assigned to the current domain, you cannot see it in the Maintain Repository Data window and therefore cannot use the GUI to modify it. To add a dictionary to a domain, from the **Definition** menu, select **Define Domains**, and click the **Dictionaries** button.

> **Note:** TMS populates the audit information fields.

Running the Activate Preliminary Data Batch Job from the GUI

To invoke Activation from the TMS Graphical User Interface:

1. From the **Repository Maintenance** menu, select **Activate Preliminary Data**. The Batch Job window appears.

    > **Note:** You can also invoke Activation (immediately, in Transfer mode only, on the current Activation Group) from the Maintain Repository Data window by clicking the **Transfer Data** button.

2. In the Job Specific section, click the **Activation Group** field. A list arrow appears.

3. Select an Activation Group from the list. (For information about Activation Groups, see "Creating or Assigning an Activation Group" on page 6-31.)

4. Click the **Activation Mode** field. A list arrow appears.

5. Select an activation mode from the list. The choices are:

    ■ **Check**. TMS invokes the Activation process, but stops short of transferring valid data to the production tables. You can view data that would fail activation in the Maintain Repository Data window and make the necessary changes before activating the data.

    ■ **Transfer**. TMS runs the Activation process to completion. Valid data is transferred to the production tables. Invalid data remains in the predict tables.

6. In the Schedule section, enter the name of the report server. Other schedule options appear.

7. Select **Submit** from the **Job** menu or click the **Submit** icon.

Running Activation from SQL*Plus

To run Activation in Transfer mode, type:

```
exec tms_user_activation.activateterms (Predict_Group_Id, 'T')
```

To run Activation in Check mode, type:

```
exec tms_user_activation.activateterms (Predict_Group_Id, 'C')
```

You can monitor the Activation process and analyze the tables to optimize execution speed. To do so, from SQL*Plus, type:

```
select count (*) from tms_dict_contents
```

In previous versions, it was necessary at this stage to compute statistics to speed the job, then reanalyze tables after job completion. The Activation batch job performs these steps for TMS 4.0 and later, so these steps are no longer necessary.

Add Validation Code (Optional)

TMS provides an empty package, **tms_ud_activation_rules**, that runs immediately after TMS has completed its own activation process. You can modify the package to include additional validation code.

For example, you can specify using a SQL statement that in MedDRA, a preferred term must always also exist as a lowest level term. If TMS finds a preferred term without a corresponding lower level term, the preferred term stays in the predictionary table with an error message saying it failed activation due to user-defined validation code.

When you write code in TMS, refer to the package **tms_def_dict_cons** for constants including the dictionary ID, dictionary levels, and any other constants relevant to all active dictionaries.

Refreshing the Context Server Index

After activation refresh the context server index; see "Refreshing the Context Server Index" on page 3-42.

Gathering Statistics to Improve Performance

Refresh schema statistics to improve performance. This is especially important after activating WHO-Drug C. See "Running Scripts to Gather Schema Statistics for the 11g Optimizer" on page 3-43.

About Activation

During Activation, TMS processes terms and relations in one Activation Group at a time, enforcing the integrity of their relations against the level relations defined for the dictionary. TMS gathers threads of data—terms related directly and indirectly to each other—and checks all the links in the thread.

> **Note:** During Activation, TMS checks for cardinality violations. If you have defined relations to more than one term in a level with only a single cardinality relation defined, TMS rejects all relations.

Rules Enforced

In addition to user-defined dictionary level relations and optional validation rules, TMS enforces the following internal rules about external, company, and TMS domain terms:

- A domain or company term cannot be linked to more than one external term, but an external term can be linked to more than one domain or company term.

- A domain term cannot be linked to more than one company term, but a company term can be linked to more than one domain term.

Database Tables

TMS stores terms awaiting activation in the predict tables (tms_predict_contents and tms_predict_relations) and moves them to the production tables (tms_dict_contents and tms_dict_relations) when they are successfully activated. Terms that fail the Activation process remain in the predict tables associated with an error message.

Activation Failure Messages

You can view the error messages associated with failed data in the **Error** field of the Maintain Repository Data and Repository Authoring windows, or by running the Preliminary Repository Report.

DML Transactions

TMS uses a series of Insert, Update and Delete transactions to move the data from the predict tables to the production tables. During the load process—when you are loading a dictionary for the first time—all transactions will be of type Insert.

More Information on Activation

The Activation process uses objects and relations defined in many parts of the system and documented as follows:

- **Activation Groups**: See "Creating or Assigning an Activation Group" on page 6-31.

  From the **Repository Maintenance** menu, select **Maintain Repository Data**, then choose **Activation Groups**.

- **Dictionary Level Relations**: See "Defining Relationships Between Dictionary Levels" on page 6-10 and"Defining Relations Between Levels (Strong Dictionaries Only)" on page 6-22.

  From the **Definition** menu, select **Define Dictionaries**, then choose **Level Relations**.

- **Level Details (Optional)**: See "Defining Level Details" on page 6-25.

  From the **Definition** menu, select **Define Dictionaries**, then choose **Level Details**.

- **Validation Code (Optional)**: See "Add Validation Code (Optional)" on page 6-32.

- **Terms and Relations**: See Chapter 12, "Repository Maintenance."

  From the **Repository Maintenance** menu, select **Maintain Repository Data**.

Evaluating Dictionary Contents

If you have the tms_maintain_priv role, you can see and evaluate data that failed activation in two places: the Maintain Data Repository Window and the Preliminary Repository Report.

If you are upgrading to a new dictionary version, additional tools are available; see Chapter 8, "Upgrading to a New Dictionary Version."

Maintain Data Repository Window

From the **Repository Maintenance** menu, select **Maintain Repository Data**, and select **All Data** from the **Data Source** list to see data in the production and predict tables. Terms and relations that failed activation have error messages explaining the reason for the failure in the **Error Message** field. See "About the Maintain Repository Data Window" on page 12-2 for more information.

Preliminary Repository Report

To run the Preliminary Repository Report, do the following:

1. From the **Repository Maintenance** menu, select **Preliminary Repository Report**.

2. Enter a **Destination Type** and any dependent fields (see "Setting Up and Running Reports and Other Batch Jobs" on page 2-13).

3. Under **Job Specific**, enter values for the following parameters:

   - **Activation Group**. From the list, choose the Activation Group whose data you want to check.

   - **Activation Data Scope**. From the list, choose **All** to see all records in the Activation, whether or not they were successfully activated; or **Errors Only** to see only data that failed activation.

4. Set schedule parameters, to run the job at a future time (optional; see "Setting Up and Running Reports and Other Batch Jobs" on page 2-13).

5. Click the **Submit Job** icon.

Defining Links Between Dictionaries

You cannot relate terms in separate TMS dictionaries until you establish linkage between the dictionaries. TMS allows you to define the following types of linkage between dictionaries:

- A **cross-dictionary link** allows you to define a dictionary named relation between two dictionaries; see "Defining Dictionary NRLs" on page 7-20. After you create the dictionary named relation, you can create relations between terms in the linked dictionaries.

- A **Translation Derivation Link** connects two dictionaries of different languages, enabling you to derive translations between corresponding terms in the dictionaries based on the dictionary code. These links are only permissible between dictionaries with identical structure.

- Define an **Is Filter Dictionary Of** link from a filter dictionary to its base dictionary. See "Defining Filter Dictionary Links" on page 6-37.

- Define a **Has Filter Dictionary** link from a base dictionary to its filter dictionary. See "Defining Filter Dictionary Links" on page 6-37.

Defining a Cross-Dictionary Link

Using cross-dictionary links can help you migrate a study from one base dictionary to another more easily. It also provides you more flexibility in creating relations in the repository.

Before you can define relations between terms in different dictionaries, you must link the dictionaries using the Dictionary Link tab window. TMS refers to the links you define in this window when it lists the candidate dictionaries for a reference term in the Repository Authoring window. After you select the first term in a relation, TMS restricts the list of candidate dictionaries for the reference term in the relation by using the cross-dictionary links defined for the first term's dictionary.

> **Note:** Links between dictionaries are bi-directional, so you can define relationships from terms in either dictionary to terms in the other dictionary once you create the link.

After you define the cross-dictionary links you need for a particular dictionary, see "Setting Up Cross-Dictionary Relations" on page 12-22 to define relations between terms in the linked dictionaries.

Rule

You must choose two Active base dictionaries. Term mappings can span dictionary levels, although group and VT levels do not support term mappings.

Creating a Cross-dictionary Link

To create a cross-dictionary link in the TMS user interface:

1. From the Define Dictionaries window, choose either dictionary in the relation by clicking its name in the structured part of the window.

2. Click the **Dictionary Link** tab. The Define Dictionaries window displays the **Dictionary Link** tab window.

3. Choose **Record**, then **Insert**.

4. From the **Link Type** field, choose **Cross Dictionary Link**.

5. From the **To Dictionary** list, choose the dictionary to which you want to link.

6. Save. If both dictionaries satisfy the linkage criteria listed above, TMS commits this cross-dictionary link to the database.

You can create links from one dictionary to multiple target dictionaries. Select links to other target dictionaries by selecting the next lines and inserting records.

You can break a cross-dictionary link by deleting its row from the **Dictionary Link** tab window. TMS allows you to remove a cross-dictionary link only if no dictionary named relations have been defined between the linked dictionaries.

Defining a Translation Derivation Link

Translation Derivation Link enable you to connect a **global language** dictionary (usually one with terms and relations in English) to an identically structured **local language** dictionary. Once you define the link, the reporting levels of the two dictionaries are linked. You can then automatically derive a translation for a reported level term in either direction: you can determine the global language equivalent of a term in the local language dictionary, or vice versa.

TMS derives a term's translation by searching in the non-coding dictionary for a term on the same level that shares the same dictionary code. Note that TMS still derives values in the coding dictionary by using the hierarchy of that dictionary.

Because TMS cannot derive translations correctly unless each code appears exactly once in each of the linked dictionaries, you should run the Translation Reports to resolve differences between the data in these dictionaries after you link them. See "Translation Reports to Identify Inconsistent Data" on page 12-36 for details.

In this section, you only define the Translation Derivation Link between two dictionaries. During the process of Linking an Oracle Clinical Study to TMS in the TMS Domain Elements window, you identify the global language dictionary and local language dictionary in this pairing, and define which one is the coding dictionary for this Oracle Clinical project or study.

Rules

The dictionaries must satisfy the following criteria:

- You must select two Active base dictionaries with identical structure. The short names of the dictionary levels must match exactly.

- The dictionaries must be in different languages. You set a dictionary's language from the **Language** field in the **Base Dictionary** tab.

- Both dictionaries must have a coding level.

- The linked dictionaries must have dictionary codes that use the same number of bytes to store each digit. Single-byte and multibyte (or "narrow" and "wide") representations of the same characters are not considered equivalent by TMS.

You can only delete Translation Derivation Links when neither of the linked dictionaries contains any data. You also cannot set a dictionary to Provisional status if it has a Translation Derivation Link.

Defining a Translation Derivation Link

To define a Translation Derivation Link:

1. From the Define Dictionaries window, choose either dictionary in the relation.

2. Click the **Dictionary Link** tab. The Define Dictionaries window displays the **Dictionary Link** tab window.

3. Choose **Record**, then **Insert**.

4. From the **Link Type** field, choose **Translation Derivation Link**.

5. From the **To Dictionary** list, choose the dictionary to which you want to link.

6. Save. If both dictionaries satisfy the linkage criteria listed above, TMS commits this cross-dictionary link to the database.

To delete a Translation Derivation Link, select its row in the **Dictionary Link** tab window, choose **Record**, and then **Delete**. As stated in the Rules, you cannot delete a Translation Derivation Link if data exists in either of the linked dictionaries.

Defining Filter Dictionary Links

When you define a filter dictionary, you must create a dictionary link between the filter dictionary and its base dictionary. You can define the link in either direction; TMS then displays the link from each dictionary. You must define these links before you can activate the filter dictionary.

Each filter dictionary can have only one base dictionary, and each base dictionary can have only one filter dictionary.

Filter to Base   To create a link from the filter dictionary to its base dictionary, do the following:

1. With the filter dictionary selected in the **Define Dictionaries** window, click the **Dictionary Link** tab. The **Dictionary Link** tab opens.

2. From the **Link Type** list, select **Filter Dictionary of**.

3. Click in the **To Dictionary** field. The **Cross Dictionary Link Dictionaries** list of values appears.

4. Select the base dictionary. For the MedDRA SMQ filter dictionary you must select MedDRA as the base dictionary.

5. Save.

Base to Filter   To create a link from the base dictionary to a filter dictionary, do the following:

1. With the base dictionary selected in the **Define Dictionaries** window, click the **Dictionary Link** tab. The **Dictionary Link** tab opens.

2. From the **Link Type** list, select **has Filter Dictionary**.

3. Click in the **To Dictionary** field. The **Cross Dictionary Link Dictionaries** list of values appears.

4. Select the filter dictionary.

5. Save.

Creating Domains and Assigning Dictionaries to Domains

This section includes:

-
-

A domain is a logical group of dictionary terms and VTAs. You must explicitly associate a dictionary with a domain in order to use it in that domain. In addition, you can define terms, relations, and verbatim term assignments (VTAs) for use only in a particular domain.

When you use TMS, you usually see a domain view of repository data; that is, you see only data associated with the current domain, including global and domain-specific terms and relations, for the dictionary or dictionaries you have linked to the domain.

TMS domains enable you to use the TMS repository differently in different situations as necessary. For example, you can use domains in the following ways:

- You can tailor dictionary modifications and VTAs for a particular use by defining them as domain-specific. For example, you could create two domains for reference by two studies that needed to track one drug's effect on different body systems.

- You can associate different dictionary combinations with different domains. For example, you might need to reference one adverse event dictionary in one country, and a different one in a different country. Your sites in the two countries can use different TMS domains.

- If you choose, you can allow classifying terms to nonapproved dictionary terms in some domains but not others. For example, in a domain used for an active study you can prevent classification to nonapproved terms, but in a domain used for post-marketing surveillance, you can allow classification to nonapproved terms to prevent unnecessary reclassification after a dictionary upgrade.

Creating a Domain

To create a domain:

1. From the **Definition** menu, select **Define Domains** to reach the Define Domains window. The Define Domains window has two tabs: **Domains** displays a single domain and its audit information, and **Multi Display Domains** displays the names and descriptions of several domains at once.

2. Insert a new record in either tab window, and enter a name and description for the new TMS domain.

Note: Do not use special characters in the **Domain Name**. Characters such as (& @ *) may cause problems, including preventing using the Disconnected System Integration feature for studies associated with the domain.

---

3. Save. TMS populates the audit fields.

4. Click the **Dictionaries** button to reach the Define Domain Dictionaries window. See "Assigning a Dictionary to a Domain" on page 6-38.

Assigning a Dictionary to a Domain

You must assign a base dictionary to a TMS domain before you can use it (filter dictionaries inherit their domain assignments from their base dictionary). A dictionary may be assigned to more than one domain, and a domain may have more than one dictionary, and any number of non-external terms, assigned to it.

> **Note:** All dictionaries associated with a single domain must be either base dictionaries or virtual dictionaries with an identical cut-off date and time. Do not use a combination of base and virtual dictionaries or virtual dictionaries with different cut-off dates (including the complete timestamp, down to the second) with a single domain.
>
> In addition, a domain can be associated with only one rendition—base or virtual—of any one base dictionary.

To assign a dictionary to an existing TMS domain:

1. From the **Definition** menu, select **Define Domains** to reach the Define Domains window and query for the domain to which you want to assign one or more dictionaries.

2. Click the **Dictionaries** button to reach the Define Domain Dictionaries window.

3. Click the ellipsis (**…**) or press F9 to launch the list of values for available dictionaries.

4. Select a dictionary and click **OK**.

5. Select **VTA Appr Reqd?** to require that in this dictionary/TMS domain combination, all VTAs are created as Unapproved. There is one exception to this rule: if the VTA has been created via Synchronization because of a direct match to a dictionary term, the VTA is created as approved.

    This may be useful at the beginning stages of a study, or use of the TMS domain, because it gives you the opportunity to manually check TMS's classifications. When you are satisfied that a classification is correct, you can manually approve the VTA in the Approve VTAs window under the **Omission Management** menu. After you do, TMS classifies all future occurrences of the verbatim term automatically.

    The default setting of the VTA Appr Reqd? flag is determined by the setting of a reference codelist value that appears in the TMS_CONFIGURATION codelist; see "DOMVTAAPPRREQD" on page 3-27.

    You can change the VTA Appr Reqd? flag setting at any time, including during an ongoing study.

6. Select the **Action Appr Reqd?** box to require that in this dictionary/TMS domain combination, all Answerable Action assignments must be manually approved. Internal Actions are used for this purpose.

    The default setting of the Action Appr Reqd? flag is determined by the setting of a reference codelist value that appears in the TMS_CONFIGURATION codelist; see "DOMACTAPPRREQD" on page 3-29.

    You can change the Action Appr Reqd? flag setting at any time, including during an ongoing study.

7. **VTI Allowed?** Check to make it possible to create Verbatim Term Individual (VTI) classifications. This is required if non-unique terms are allowed. It is possible to select this setting only if VTIs are allowed in the dictionary definition.

8. **Autocode with Aux?** Check to enable the TMS autoclassification process to create VTIs after the same term with the same auxiliary information has been manually classified. It is possible to select this setting only if it is allowed in the dictionary definition. Changing this setting affects only future executions of Autoclassification.

9. In the **Non Appr DT** drop-down:

   ■ Select **None** to prevent classification to nonapproved dictionary terms.

   ■ Select **VTA** to allow classification to nonapproved dictionary terms. This setting may be appropriate, for example:

       – during post-marketing surveillance if you do not want to have to reclassify verbatim terms whose dictionary terms become invalid (nonapproved) after a dictionary upgrade

       – if you are using MedDRA-J and want to be able to classify terms to nonpreferred Japanese terms when multiple Japanese terms correspond to the same English term.

   When you select this option, in this domain the Autoclassification job treats nonapproved terms exactly the same as approved terms in this dictionary.

   > **Note:** You can change this value from VTA to None only if no current VTAs are linked to a nonapproved dictionary term in the dictionary/domain combination.

10. To add another dictionary, click in the next row and repeat the process.

11. Click **OK**. TMS saves the domain/dictionary links you have defined.

Unassigning a Dictionary from a Domain

To unlink a dictionary (to prevent its use in this TMS domain), highlight it and select **Record**, then **Delete**.

> **Note:** You cannot delete a domain/dictionary assignment if a filter dictionary is defined on the base dictionary and contains dictionary terms. Filter dictionaries inherit the domain assignments of their base dictionary. See "Filter Dictionaries" on page 6-4 for more information.

7

Defining Other TMS Elements

Oracle Thesaurus Management System (TMS) enables you to define elements other than dictionaries and domains in the user interface (see Chapter 6, "Defining and Loading Dictionaries" for instructions about defining dictionaries and domains). To define any of the following objects, you must have the tms_define_priv role.

If you have a distributed environment, you can perform the following tasks at the master instance only:

- Defining HTML Layouts on page 7-2
- Customizing Report Templates on page 7-4
- Defining and Using Actions on page 7-6
- Defining Search Objects on page 7-10
- Defining Named Relations on page 7-16
- Defining Informative Note Attributes on page 7-22
- Defining Dictionary-wide Informative Notes on page 7-26

See Chapter 3, "Administration" for information on the following tasks that appear in the **Definition** navigation path:

- Customizing Defaults in TMS Windows Using TMS Settings on page 3-17
- Setting TMS Preferences with Reference Codelist Values on page 3-25
- Synchronizing Reference Codelist Settings on page 3-32 (local instances only)
- Controlling the Use of Materialized Views on page 3-39
- Refreshing the Context Server Index on page 3-42
- Synchronization on page 3-36
- Analyzing Tables on page 3-42
- Purging Classified Omissions from the Omissions Table on page 3-42

See Chapter 5, "Integrating TMS with Other Systems" for information on the following tasks that appear in the **Definition** navigation path:

- Defining External System Information in TMS on page 5-28
- Register Remote Databases on page 5-8
- Running the DSI Initialization Job on page 5-7
- Delete DSI X Areas on page 5-27

Defining HTML Layouts

HTML layouts enable you to customize which data columns you present for Terminology Searches in the HTML Browser. Layouts are dictionary-specific, and rely on user-defined functions that derive TMS data from the production tables.

There is no security placed on HTML Layouts. You can edit or delete any HTML Layouts in the database.

This section includes the following subsections for creating HTML Layouts:

- Defining HTML Layout Functions on page 7-2
- Defining the Layout Name and Function on page 7-3
- Defining the Layout Columns on page 7-3

To delete a layout in this database, see "Deleting a Layout" on page 7-4.

Defining HTML Layout Functions

Before you can define an HTML Layout in the TMS application, you must define a function that you will use to derive data for that layout.

This section describes how to structure data in a package specification and package body so that the function can be used by an HTML Layout. For full samples, see the TMS Knowledge page on the My Oracle Support website.

Sample Package Specification

The package specification must include each function you want to use in an HTML Layout. This example for the package dmo_package includes just two functions, LayoutOne and LayoutTwo. You can define as many packages as you want, each containing as many functions as you want.

Example 7–1   Sample Package Specification for HTML Layouts

```
CREATE OR REPLACE PACKAGE dmo_package IS

  FUNCTION LayoutOne(
          pRec      IN sys_refcursor
          )
  RETURN Tms_qry_result_TypeSet PIPELINED
  ;

  FUNCTION LayoutTwo(
          pRec      IN sys_refcursor
          )
  RETURN Tms_qry_result_TypeSet PIPELINED
  ;


END dmo_package;
/
sho err
grant execute on dmo_package to rxclin_read;
exec opa_ddl.CreateDropPublicSynonym( 'dmo_package');
```

Defining an HTML Layout Function

HTML Layout functions handle data using reference cursors of the type Tms_qry_result_Type. You can use the columns in this cursor to specify which data are

presented in HTML Layouts that use this function, and how this data should be presented. The column types are:

text_n columns: The fifteen text columns (text_1, text_2, …) describe which data columns you return to the layout. You can point directly to a specific data element such as the term name, or use TMS' data model to derive a related term record, like the term's parent.

description_n columns: The fifteen description columns (description_1, description_2, …), in a one-to-one relationship with the corresponding text_n columns, specify how each data element should be presented in the layout.

Defining the Layout Name and Function

An HTML Layout uses one TMS dictionary and one layout function. This unique linkage is necessary because layout functions are written to be very dictionary-specific. In this section, you specify the dictionary that will use this layout, and the layout function that will deliver data to the HTML Layout.

To define a layout name and its function:

1. From the **Definition** menu, select **Define HTML Layouts**. The Define HTML Layouts window opens.

2. Choose **Dictionary** as the **Ref Type**. This is currently the only choice available.

3. From the **Ref Name** field, choose the dictionary to which you want to link this layout. You will only be able to use this layout in the HTML Browser when this dictionary is selected.

4. Enter a **Display Name** for this layout. This name appears in the **Layout** list in the HTML Browser when the linked dictionary is selected.

5. Enter the name of the function you want to use for this layout, in `package_name.function_name` format. To use LayoutOne from Example 7–1, enter `dmo_package.LayoutOne`.

6. Save.

Proceed to "Defining the Layout Columns" on page 7-3 to complete the layout definition.

Defining the Layout Columns

Defining the columns for a layout means specifying which columns will display in the layout, in which order, and the headings that will display for those columns. You can choose the columns that are made available in the layout function: if you want a particular layout column to display the term name, and the layout function handles this data in the term_6 column, enter "6" as the **Col ID** for this column. See "Defining HTML Layout Functions" on page 7-2 for information on writing an HTML Layout function.

You should define the columns in the order that you want them to display in the layout. The first column you enter in the **Columns** block will be the leftmost column in the layout.

The names that you enter for the columns are the names that display as headings when you use the layout in the HTML Browser. You can change column display names at any time.

To define the columns for a layout, choose the layout you want to modify from the upper block of the window, then repeat the following steps for each column in the layout.

1. In the **Columns** block, click in an empty row or add a record.

2. In the **Col ID** field, enter the number that corresponds to the data column you want to use in the layout function. If you want this column to display the term name, and the layout function handles this data in the term_6 column, enter "6" for this column.

3. Enter the **Display Name** for this column. The HTML Browser displays this name in the column heading when it displays results in this layout.

4. Repeat for each additional column, then save. TMS makes this layout available to the HTML Browser.

Deleting a Layout

You can remove HTML Layouts from the database by first deleting all of the layout's columns, then deleting the layout itself.

Customizing Report Templates

This section includes:

- Downloading a Report Template on page 7-4
- Modifying a Report Template on page 7-5
- Uploading a Custom Report Template on page 7-5
- Updating or Deleting a Custom Report Template on page 7-5

You can customize the layout of TMS reports using Oracle Business Intelligence (BI) Publisher. You can download the default template and, for example, change the background color, add conditional statements, insert logos and other images, change the order of the fields displayed. You then upload this customized template back into TMS with its own name. Each time a user runs the report, he or she can choose to use the default template or a customized template.

TMS allows you to upload multiple templates for each report.

You cannot overwrite or delete the default Oracle template.

The TMS_DEFINE_PRIV privilege is required to customize templates.

Downloading a Report Template

To download a report template, follow the instructions below:

1. Log into TMS.

2. Click the HTML Browser link from the TMS opening screen. The HTML Browser opens.

3. Click the Reports tab.

4. Select the appropriate report from the **Report** field. The form refreshes and lists the selected report's templates in the table below.

5. From the list, click the appropriate template name. (The predefined default template is called the Oracle Template.) A standard File Download pop-up box opens.

6. Click **Save**. A standard Browse pop-up box opens.

7. Navigate to the location where you want to work on the template and click **Save**. The system saves a copy of the template as an .rtf file on your local system.

8. Save the file.

Modifying a Report Template

To create your first custom template for any report, begin by downloading the default Oracle template. You can create additional custom templates or modify existing custom templates at any time.

1. Open the downloaded template file locally in MS Word.

2. Make the changes you require. For information about using Oracle BI Publisher, go to the following website:

   http://docs.oracle.com/cd/E23943_01/bi.1111/e22254/toc.htm

3. Save your changes.

Uploading a Custom Report Template

To upload the template after customization, follow the steps below:

1. Log into TMS.

2. Click the HTML Browser link from the TMS opening screen. The HTML Browser opens.

3. Click the Reports tab.

4. Enter a name for the new customized template in the **Template Name** field.

5. Using the **Browse…** button, select the .rtf template file from your system.

6. Click the **Create New Template** button. The form refreshes and adds the customized template to the list of templates for this report.

Updating or Deleting a Custom Report Template

To update or delete an existing template, follow the steps below:

1. Log into TMS.

2. Click the HTML Browser link from the TMS opening screen. The HTML Browser opens.

3. Click the Reports tab.

4. Select the appropriate report from the **Report** field. The form refreshes and lists the selected report's templates in the table below.

5. Choose the appropriate template by clicking its corresponding button in the **Select** column. You may select only one template at a time.

6. Click **Checkout Template**. Your user ID appears in the Checkout column.

Only one person can check out a template at a time. The template version that was current at the time of checkout remains available for use in TMS until the updated version is uploaded. The new version is then available.

> **Note:** If you decide not to update or delete the template, click **Undo Checkout**.

7. If you are updating the template, download the template and modify it locally; see "Downloading a Report Template" on page 7-4 and "Modifying a Report Template" on page 7-5.

8. Select the report you are updating or deleting, if it is not already selected.

9. If you are updating the report, enter a template name and browse for the modified local .rtf template file, then click **Upload Template**. The system updates the template and automatically checks it in.

If you are deleting the template, click **Delete Template**.

TMS does not allow you to overwrite or delete the default Oracle template.

Defining and Using Actions

This section includes:

- Action Types on page 7-6
- Using Actions on page 7-8
- Defining an Action on page 7-9

You can use actions for communication with personnel using an external source data system, or for communication within TMS, about an omission.

Action Types

There are three types of actions: Answerable Actions and Unaswerable Actions, both of which TMS sends to an external system, and Internal Actions, which are used within TMS.

After you apply an Action of any type to a term, TMS automatically applies the same Action to new occurrences of the same term in the same dictionary/domain combination.

> **Note:** Before TMS Release 4.6, Answerable actions were known as Conditional actions, and Unanswerable actions were known as Never Expire actions. Internal actions were new in TMS Release 4.6.

Answerable Actions

Sometimes a verbatim term cannot be either automatically or manually because the verbatim term itself is flawed. In this case, the person trying to manually classify the term instead associates an Action with the omission and sends it back to the external system requesting that the term be amended and resubmitted to TMS.

For example, TMS does not support combined verbatim terms. TMS can process only one symptom per term in a medical dictionary; either *bad headache* or *nausea*, for example, but not *bad headache and nausea*. In this example, you could apply an Action

with the text "Split the term" to the verbatim term *bad headache and nausea*. A verbatim term may also be ambiguous—too vague; for example, *head*—or completely garbled due to a typing error. In this case you could apply an Action with the text "Clarify the term" to the verbatim term.

Answerable actions allow a user in the external system to send a comment back to TMS. When that happens, TMS resumes ownership of the omission and a TMS user must take the next step: either classifying the term or sending a discrepancy message back to the external system to respond to the external comment. For example, you assign an Action to the term "Pain" with the text, "This term is too vague. Please be more specific." An external user responds, "The patient did not provide any more information." You see this response in the Classify VT Omissions window, All VT tab, Action Text field. You can use a discrepancy message to respond, for that source term only, "TMS cannot process this term without information about where in the body the pain occurred. Please get additional information."

See "Applying a Discrepancy Message" on page 10-8 for more information about Discrepancy Messages.

Unanswerable Actions

If you assign an Action of type Unanswerable to a term, TMS does not register comments sent back to TMS from the external system. If an external system user replies to the Action text for a particular source term without changing the term itself, TMS sends the same Action text back during the next data exchange job. For example, if you assign an Action definition with the text "This term is too vague. Please be more specific" to the term "Pain" and an external system user responds, "The patient did not provide any more information," this comment never appears in TMS, the omission ownership remains set to the external system, and TMS sends back the original Action text ("This term is too vague. Please be more specific") in the next data exchange with the external system.

> **Note:** The external system does not "see" and so does not display the Action type. Therefore the external system user does not know whether TMS can read his or her response or not.

Use unAnswerable Actions in cases where you know a particular response is always wrong, or if you do not have the resources to support a dialog.

Internal Actions

Internal Action definitions are used for two purposes, Action Approval and Internal Communication:

Action Approval Use Internal Actions with a base Answerable Action as a mechanism for approving the assignment of a particular Answerable Action to a particular verbatim term. After a TMS user approves the Action assignment, TMS sends the Answerable Action to the external system.

To use Internal Actions as a mechanism for approving the assignment of a particular Answerable Action to a particular verbatim term, do the following:

1. Set up Action approval:
   – In the dictionary/domain combinations where you want to require the approval of Action assignments, check **Action Appr Reqd?**; see "Assigning a Dictionary to a Domain" on page 6-38.

– For each Answerable Action that you may use in a dictionary/domain combination where Action approval is required, define an Internal Action with the Answerable Action as the base Answerable Action. Use a naming convention so that coders can select the correct Internal Action to assign to an omission; see "Defining an Action" on page 7-9. When you specify the base Answerable Action, TMS supplies its text as the text for the Internal Action. It makes sense to keep them the same.

2. Assign actions to terms. In the Classify VT Omissions window under Omission Management, if you are working in a dictionary/domain combination where Action approval is required, TMS displays only active internal and unAnswerable Actions in the Action drop-down list; see "Classifying Terms Manually" on page 10-3.

3. In the Approve Action Assignments window under Omission Management, a user with the TMS_APPROVE_PRIV role can approve an Internal Action assignment. TMS then applies the Internal Action's base Answerable Action to the term and sends the term with its Action to the external source data system during the next data exchange (Batch Validation for Oracle Clinical).

   If the user instead rejects the assignment, or assigns a different Action, TMS removes the Internal Action assignment. A record of the Internal Action assignment is available in the status history of the term.

Internal Communication Use Internal Actions without a base Answerable Action to indicate to other coders that someone is already working on a particular verbatim term.

For example, a coder may need to check with his or her supervisor before classifying a verbatim term. To avoid a duplication of effort, the coder first applies an Internal Action with no base Answerable Action, and with text such as "Inquiry in progress" to the term. When the coder collects the required information, he or she can either classify the term or apply another Action to it.

Any user with the privileges normally required can either classify or apply another Action to the term. TMS then removes the Internal Action without a base Answerable Action from the term.

To use Internal Actions for internal communication only, do the following:

1. Define one or more Internal Actions without a base Answerable Action.

2. Assign actions to terms. In the Classify VT Omissions window under Omission Management, apply an Internal Action without a base Answerable Action to a term; see "Classifying Terms Manually" on page 10-3.

3. When you are ready, classify the term or assign a different Action to it.

Using Actions

Using actions involves many parts of TMS described elsewhere:

■ **Defining Actions** is described in this section on page 7-9.

■ **Requiring Approval for Action Assignments**. You can require that actions assigned to omissions be approved before they are sent to the external system. You require this approval for a particular dictionary/domain combination; see "Assigning a Dictionary to a Domain" on page 6-38. Internal Actions are used for this purpose; see "Internal Actions" on page 7-7 for further information.

- **Applying Actions to Omissions**. You assign actions to omissions in the Classify VT Omissions window; see "Apply an Action to a Term" on page 10-7 for details.

- **Applying Actions to VTs Before They Occur**. You can apply actions to verbatim terms proactively, even before a verbatim term enters the system. In the Maintain Actions window, you can enter any verbatim term you choose (or any term that you think may be entered as a verbatim term in the future) and assign an Action to it. See "Maintaining Action Assignments" on page 11-4.

- **Viewing Action Assignment History**. You can see actions that have been assigned to a term as follows:

  - You can see the actions that have been assigned to a verbatim term in the Distinct VT Actions tab of the Classify VT Omissions window; see Chapter 10, "Omission Management."

  - You can see the actions that have been assigned to a specific source term in the All VT Actions tab in Classify VT Omissions form; see Chapter 10, "Omission Management."

  - The Status/Notes window, called from Classify VT Omissions, Approve VTAs, Approve VTA Assignments, or VT History window, shows all term status history information, including actions; see "Using the Status/Notes Pop-up Window" on page 10-21.

  - The HTML Browser displays terms' complete Action history; see "Viewing a Term's Action History" on page 14-24

Defining an Action

To create an Action Definition, do the following:

1. From the **Definition** menu, select **Create Action Definitions**. The Create Action Definitions window opens.

2. In the **Action** field, enter a short (no more than 15 characters), descriptive name for the Action. In the *bad headache and nausea* example above, this might be "Split term."

   > **Note:** If you are defining an Internal Action with a base Answerable Action, use a naming convention such as *base Answerable Action name _* INT to name the Internal Action, so that coders can select the correct Internal Action to assign to an omission. For example, if the base Answerable Action is named SPLIT TERM, name the corresponding Internal Action SPLIT TERM_INT. (Note that if you follow this naming convention, which requires 4 characters for the Internal Action name, base Answerable Actions can have a maximum of 11 characters in their name.)

3. In the **Action Type** field, select **Answerable**, **Unanswerable** or **Internal**. See "Action Types" on page 7-6 for information.

   When the choice is Internal for the purpose of requiring an Action assignment to be approved before sending to the external system, you must select an option from the Base Answerable Action LOV. This LOV contains all defined Answerable Actions. Each Internal Action can be based on only one Answerable Action. Likewise, an Answerable Action can be associated with only one Internal Action.

4. Leave the **Status** field value set to **Active**. If you need to stop using the Action in the future, you can set the status to **Retired**. The system then removes all current assignments of the Action.

> **Note:** If an Action has never been applied to a term, you can delete it. After it has been applied, you can only retire it.

5. **Base Answerable Action**. Use this field only if you are defining an Internal Action with a base Answerable Action for use in one or more dictionary/domain combinations where Action approval is required. Specify the Answerable Action to be sent to the external system if the assignment of the Internal Action you are defining is approved.

   An Internal Action can have only one base Answerable Action (and should have a similar name to the Base Answerable Action's name or text; see above). An Answerable Action can be associated with only one Internal Action as a base Answerable Action.

> **Note:** In dictionary/domain combinations where Action approval is not required, you can apply an Answerable Action directly to an omission, even if it is associated with an Internal Action.

6. In the **Text** field, enter the default text to send to external system users, explaining what they need to change to allow the system to process the term. In the same example, this text might be "Please split this term."

   If you are defining an Internal Action with a base Answerable Action, click in the **Text** field. TMS populates it with the base Answerable Action's text. This text will be sent to the external system associated with a source term after the Action assignment is approved. The person who approves the assignment can modify the text.

7. In the **External System** field, specify one or more external systems collecting source terms (Oracle Clinical, for example) to which you may want to send this Action.

8. In the **Omission Status** field, enter the omission status you want the external system to assign to the omission when it returns from TMS. In Oracle Clinical, this status is normally Inv Review (Investigator Review) to indicate that the investigator must take the next step in resolving the discrepancy, but you can enter any valid status.

   An external system-specific LOV is available for this field; you can define this LOV in the **Multiple Term Action Omission Statuses** LOV field of the Define External Systems window.

   For each external system, select one omission status.

9. Save.

Defining Search Objects

Search objects contain algorithms used to search the TMS repository in three situations:

■ To find matches for verbatim terms during Autoclassification.

- To find Candidate Terms during manual classification.

- To find repository terms in the Extended Search window, which can be opened from the Browse Repository Data window and the Classify VT Omissions window.

In the Define Search Objects window you activate each search object in the dictionaries and domains you want, and specify how you want to use the search object (see "Defining a Search Object" on page 7-11).

TMS runs the Nonapproved VTA search before other search objects during Autoclassification, manual classification and extended searches. It searches in the current domain for an exact match that is a Nonapproved VTA. You cannot see the Nonapproved VTA search object in the Define Search Objects window because you cannot modify it. However, you see references to it in the **Search Type** and **Origin** fields in several windows.

Sample Domain Match Search Object   TMS includes a pre-defined search object called Domain Match that is available for you to use as you wish. It searches for an exact match of a VTA in another domain. If it finds more than one match, it returns multiple Candidate Terms. If it finds one and only one match, it either autoclassifies the term or returns a Candidate Term, depending on the Approval Type setting. In addition, if it finds one and only one match:

- If the verbatim term's dictionary/domain combination allows classification to nonapproved dictionary terms, the Domain Match search object includes VTAs linked to nonapproved dictionary terms in its search.

- If the verbatim term's dictionary/domain combination does not allow classification to nonapproved dictionary terms, the Domain Match search object does not include VTAs linked to nonapproved dictionary terms in its search, even if they exist in other domains.

In addition, you can write your own search algorithms in the database and use the Define Search Objects window to identify them to TMS and define their usage.

Defining a Search Object

This section discusses defining a search object using the graphical user interface. For information about defining a search object using the API, see the *Oracle Thesaurus Management System Technical Reference Manual*.

To use either the pre-packaged search object Domain Match or a user-defined search object, you must activate it and specify how you want to use it for each dictionary. You can use different settings in different domains.

To define a search object:

1. From the **Definition** menu, select **Define Search Objects**. The window opens with the **Define Search Objects** tab selected.

2. Specify the following information about your search object, then save:

Name   The name of the search object.

Inherit?   If selected, the search object you create and assign to this base dictionary will also be created for all virtual dictionaries that use the base dictionary.

Description   Enter an explanation of what the search object searches for. Up to 2000 alphanumeric characters.

Use VTA? Applies to Autoclassification (Autocode Search Objects) only. Select if you want the system to search for VTAs as well as dictionary terms during Autoclassification.

Stop 1:m? Exit criteria. This applies only to Autoclassification (Autocode Search Objects). Select if you want the system to stop the search if it finds more than one match using any one search algorithm. (If the system finds more than one match during a Candidate Term search or an extended search, it will return all the matches it finds.)

Approval Type Select one of the following to specify the behavior you want if the search finds one and only one match:

- **Approved**. TMS automatically creates the VTA and uses it to derive information.

- **Omission**. TMS does not create a VTA, but displays the omission with the match it finds as the one and only Candidate Term, with the name of the search object that found it, in the Classify VT Omissions window. No values can be derived for the verbatim term until you classify it. When you classify the omission, you can choose to create either an approved or a Nonapproved VTA.

Autocode Object Enter the name of the function you have written for TMS to run during Autoclassification. For the Domain Match search object this is predefined. For user-defined search objects, enter the function name here or, if the function is included in a package, enter *package_name.function_name*. See "Creating Custom Search Algorithms" on page 7-14.

Candidate Object Enter the name of the function you have written for TMS to run to generate Candidate Terms during manual classification. For the Domain Match search object this is predefined. For user-defined search objects, enter the function name here or, if the function is included in a package, enter *package_name.function_name*. See "Creating Custom Search Algorithms" on page 7-14.

Candidate Type Select **Package**. TMS does not currently support the **Form** option.

Extended Search Object Enter the name of the function you have written for TMS to run to browse the repository during an extended search. For the Domain Match search object this is predefined. For user-defined search objects, enter the function name here or, if the function is included in a package, enter *package_name.function_name.* See "Creating Custom Search Algorithms" on page 7-14.

Setting the Search Objects for a Dictionary

This section includes:

- Specifying the Use of a Search Object with a Particular Dictionary on page 7-12
- Removing a Search Object's Association with a Dictionary on page 7-13

Specifying the Use of a Search Object with a Particular Dictionary

You must specify the dictionaries with which each Search Object is available for use, as follows:

1. Click the **Dictionary Mappings to Search Objects** tab.

2. From the **Dictionary** list, choose the dictionary to which you want to apply the search object.

3. For each search object you want to use against this dictionary, fill in the fields in the following order, then save:

Search Object   Select the search object from the scrolling list.

Order   Specify the order in which you want TMS to execute this search algorithm in relation to the others listed here. Numbers do not have to be consecutive. Search objects with lower numbers will be executed before those with higher numbers.

Domain   The list includes all TMS domains with which the dictionary is linked. The domain-specific settings in this lower portion of the window override the corresponding dictionary-wide settings in the upper portion of the window. To use the search objects the same way in all domains, do not enter a domain name here.

Disabled?   Select to disable the search object in this domain/dictionary combination.

> **Note:**   If you have used a search object in a dictionary/domain combination and then disabled it, the disabling applies only to future occurrences of terms. Users can still query for and find omissions created using the search object in the Classify VT Omissions window.

Approval Type   From the list, choose the result you want TMS to return if it finds one and only one match during the search in this domain/dictionary combination. See "Approval Type" on page 7-12.

Removing a Search Object's Association with a Dictionary

If you decide that you no longer want to use a particular search object with a dictionary in any domain, you can remove the search object by using the **Dictionary Mappings to Search Objects** tab.

Omissions previously created using the search object still exist and are queryable and viewable in the Classify VT Omissions window. However, the Search Object field for these omissions is blank, and the system does not use the search object to create any additional omissions associated with the dictionary.

To remove a search object's association with a dictionary:

1. From the **Definition** menu, choose **Define Search Objects**.

2. Query for the search object you want to remove from a dictionary.

3. Click the **Dictionary Mappings to Search Objects** tab.

4. From the **Dictionary** list, choose the dictionary from which you want to remove a search object. TMS lists the search objects defined for this dictionary.

5. Scroll to, or query for, the search object you want to remove, and select its row.

6. With the search object's row highlighted, click in one of the three detail fields (**Domain**, **Disabled?**, and **Approval Type**), and delete the record. This deletes the detail records for this association, which you must do before deleting the association itself.

7. Select the search object again, and delete the record. TMS deletes the search object's association with this dictionary.

8. Save. TMS deletes the search object's association with this dictionary.

Deleting a Search Object

If you decide that you no longer want to use a search object with any dictionary, you can delete it entirely from TMS. Previously created omissions still exist and are queryable and viewable in the Classify VT Omissions window. However, the Search Object field for these omissions is blank, and the system does not use the search object to create any additional omissions.

To delete a search object:

1. From the **Definition** menu, choose **Define Search Objects**.

2. Query for the search object you want to remove from a dictionary and select it.

3. From the **Record** menu, choose **Delete Record**.

4. Save. TMS deletes the search object.

Creating Custom Search Algorithms

Each of the search algorithms provided with TMS searches only for direct matches to the verbatim term in the TMS repository. You may want to take advantage of the text-retrieval capabilities available in the Oracle database. The *inter*Media Text software, formerly known as the Context Server Cartridge, is integrated in the RDBMS. See "Entering a Context Query" on page 2-10.

To create a custom search algorithm, create a PL/SQL function in the database and enter its name in the Search Object Definition window in TMS in the appropriate field—**Autocode Object**, **Candidate Object**, or **Extended Search Object**—and set the candidate_type='package'. For an overview, see "Defining a Search Object" on page 7-11).

If the function is part of a package, follow the naming convention *package_name.function_name* for functions. When TMS runs Autoclassification, a Candidate Term search, or an extended search, it calls the function in the order you specify in the **Exec Order** field.

You will probably want to make each search algorithm available in all three situations, but this is optional.

The input and output parameters vary in each situation as follows:

Writing a Search Algorithm for Autoclassification

You can create one or more custom search algorithm to supplement TMS's Autoclassification process. Create a PL/SQL function in the database and enter its name in the Search Object Definition window in TMS in the **Autocode Object** field (see "Defining a Search Object" on page 7-11). Each Autoclassification search algorithm must have the following specifications:

```
FUNCTION function_name (
  pDefDictionaryId IN tms_def_dictionaries.def_dictionary_
id%TYPE
  , pDefDomainId IN tms_def_domains.def_domain_id%TYPE
  , pVtLevelId IN tms_def_levels.def_level_id%TYPE
  , pVtCodeLevelId IN tms_def_levels.def_level_id%TYPE
  , pVTASearchFlag IN VARCHAR2
  , pTermUpper IN tms_dict_contents.term_upper%TYPE
  , pVDefDictionaryID IN tms_def_dictionaries.def_dictionary_
id%TYPE
  , pCutoffDate IN tms_def_dictionaries.cutoff_date%TYPE
  , pDictContentid IN OUT tms_dict_contents.dict_content_
```

```
id%TYPE
) RETURN PLS_INTEGER;
```
The *function_name* may be part of *package_name.function_name* and must correspond to the value entered in the **Autocode Object** field of the Define Search Objects window.

The function has one output parameter that identifies the dictionary term or VTA match it finds. The function returns an integer as follows:

Value	Returned if Function Finds	Next, TMS
0	no matches	runs next search object, if any
1	one and only one match	stops Autoclassification, returns match
2	more than one match	if the exit criterion Stop 1:m is set to Y, Autoclassification stops and returns the search object that found the matches; if the exit criterion is set to N, TMS runs the next search object, if any.

Writing a Search Algorithm for Candidate Term Generation

You can create one or more custom search algorithm to supplement TMS's Candidate Term search process. Create a PL/SQL function in the database and enter its name in the Search Object Definition window in TMS in the **Candidate Object** field (see "Defining a Search Object" on page 7-11). Each candidate search algorithm must have the following specifications:

```
FUNCTION function_name (
  pDefSearchId IN tms_vt_omissions.def_search_id%TYPE
  , pDefDictionaryId IN tms_vt_omissions.def_dictionary_id%TYPE
  , DefDomainId IN tms_vt_omissions.def_domain_id%TYPE
  , pVtLevelId IN tms_def_levels.def_level_id%TYPE
  , pVtCodeLevelId IN tms_def_levels.def_level_id%TYPE
  , pTermUpper IN tms_dict_contents.term_upper%TYPE
  , pVDefDictionaryID IN tms_def_dictionaries.def_dictionary_
id%TYPE
  , pCutoffDate IN tms_def_dictionaries.cutoff_date%TYPE
  , pCutoffDate IN tms_def_dictionaries.cutoff_date%TYPE
) RETURN VARCHAR2;
```
The *function_name* may be part of *package_name.function_name* and must correspond to the value entered in the **Candidate Object** field of the Define Search Objects window.

The function returns a string of characters that populates the Where clause, which populates the lower block of the Classify VT Omissions window. The function can be invoked by the user during manual classification (see "Candidate Terms and Search Objects" on page 10-2).

In addition, you can restrict an algorithm's search to only current terms by adding the following to the populated where clause:

```
AND end_ts = to_date(3000000, 'J')
```

Writing a Search Algorithm for Extended Searches

You can create one or more custom search algorithm to supplement TMS's extended search process, which can be invoked from the Browse Repository window or the Classify VT Omissions window. Create a PL/SQL function in the database and enter its name in the Search Object Definition window in TMS in the **Extended Search Object** field (see "Defining a Search Object" on page 7-11). Each extended search

algorithm must have the following specifications:

```
FUNCTION function_name (
  pDefSearchId IN tms_vt_omissions.def_search_id%TYPE
  , pDefDictionaryId IN tms_vt_omissions.def_dictionary_id%TYPE
  , DefDomainId IN tms_vt_omissions.def_domain_id%TYPE
  , pVtLevelId IN tms_def_levels.def_level_id%TYPE
  , pVtCodeLevelId IN tms_def_levels.def_level_id%TYPE
  , pTermUpper IN tms_dict_contents.term_upper%TYPE
  , pVDefDictionaryID IN tms_def_dictionaries.def_dictionary_
id%TYPE
  , pCutoffDate IN tms_def_dictionaries.cutoff_date%TYPE
) RETURN VARCHAR2;
```

The *function_name* may be part of *package_name.function_name* and must correspond to the value entered in the **Extended Search Object** field in the Define Search Objects window.

The function returns a string of characters that populates the Where clause, which populates the Extended Search window.

Defining Named Relations

This section includes:

About Named Relations

Standard-type named relations create a structure for weak dictionary folders by describing the nature of relations between terms. You can define named relations as being one-directional or bi-directional, as shown in Figure 7–1.

Figure 7–1   One-directional and Bi-directional Relationships
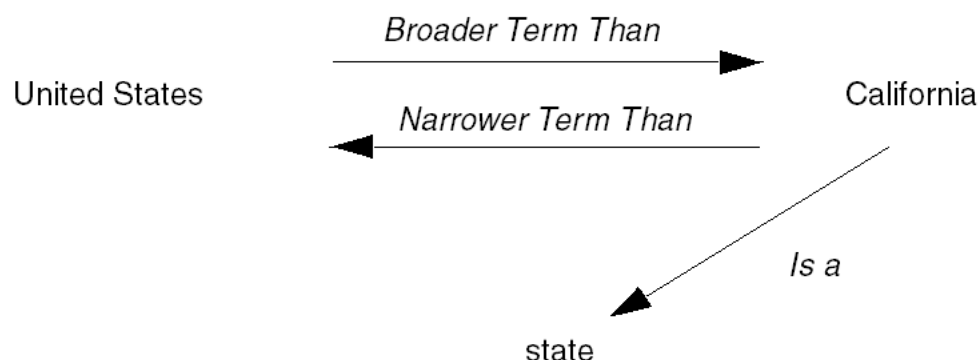
Bi-directional relationships like the one above between "United States" and "California" have two components, an Indicator relationship and a Reciprocal Indicator relationship. Indicator relationships describe the link from the source term to the reference term, and Reciprocal Indicators describe the reverse relation. If "United

States" is the source term for the above relation, "Broader Term Than" is the Indicator for this relationship and "Narrower Term Than" is the Reciprocal Indicator.

One-directional named relations have Indicator relationships only. Thus, in the example one-directional relation above, no relationship is defined from "state" to "California."

Named relations also define the cardinality of the relation between terms. The cardinality of a relationship determines the maximum number of reference terms (one or many) to which a term may be linked. If you specify Many Cardinality for the Indicator relationship but Single Cardinality for the Reciprocal Indicator, then source terms can be linked to many reference terms, and each reference term must link to only one source term.

Because named relations create a dynamic dictionary structure, you cannot enforce that named relations be mandatory. You can, however, enforce mandatory relations between levels in a strong dictionary; see "Mandatory Relations" on page 6-11.

TMS does not allow circular named relations of the same type among terms. That is, if you define an Indicator relationship of a particular type (such as Related Term) between Term A and Term B, and another Indicator relationship of the same type between Term B and Term C, you cannot define an Indicator relationship of the same type between Term C and Term A. Activation of the final relationship in the circle fails. (The reason for this restriction is that the ANSI/NISO Z39.19-1993 standards for thesauri specify that you must be able to represent a thesaurus structure using a flat display, which is impossible with circular relations.)

You can introduce a named relationship into your installation using the Define Named Relationships window, or by loading the relationship directly into the repository with a load script (see "About Load Scripts" on page 6-29). TMS stores named relations in the tms_def_named_rels table, and enables you to use any named relation to link terms in any dictionaries throughout the TMS installation.

See "Defining Named Relations" on page 7-19 and "Creating Named Relations Between Terms" on page 12-20 for more information, including information on other types of named relations.

Types of Named Relations

You can use named relations to create browsable relations between dictionary terms.

There are three types of named relations:

- **Standard**. Use standard named relations to create relations among terms in weak dictionary folders or to supplement existing relations among terms in strong dictionaries.

  After you define a standard named relation, you can create standard named relations between terms in the Repository Authoring window under the Repository Maintenance menu.

- **Translation**. Use named relations of type Translation to define a relationship between a term in one dictionary and a term with the same meaning in another dictionary in a different language. TMS allows named relations of type Translation only between terms in different dictionaries, each of which must have a different language defined.

  After you define a Translation named relation, you can create Translation named relations between terms in the Repository Authoring window under the Repository Maintenance menu.

- **Release Label**. Use named relations of type Release Label to create relations between terms in the same or different dictionaries at different points in time.

  This is especially useful for maintaining an audit trail of complex changes to a term from one dictionary version to another; for example, where two terms are merged, or one term is split into two or more terms, or one term is substituted for another.

  After you define a Release Label named relation, you can create Release Label named relations between terms in the Release Label Authoring window under the Repository Maintenance menu; see "Using the Release Label Authoring Window" on page 12-30.

Release Label Named Relation Example 1   The MedDRA 7 high level terms "Liver and spleen infections" and "Gallbladder infections" were merged in MedDRA 8 into a single term, "Hepatobiliary and spleen infections."

If you are currently using MedDRA 7 in TMS, do the following:

1. After upgrading to TMS 4.6, define a label prefix for MedDRA 7: 7.0.

2. Create a virtual dictionary of MedDRA 7. TMS creates a Release Label for the virtual dictionary.

3. Define a named relation called "is merged into;" see "Defining Named Relations" on page 7-19.

4. Update the label prefix to 8.0.

5. Load data for MedDRA 8 into the predictionary tables.

6. Define two named relations in the Release Label Authoring window, one from "Liver and spleen infections" to "Hepatobiliary and spleen infections" and one from "Gallbladder infections" to "Hepatobiliary and spleen infections." In both cases, given the definition "is merged into," "Hepatobiliary and spleen infections" must appear in the Reference section of the screen. See "Creating Named Relations Between Terms" on page 12-20. Use the NEXT setting for "Hepatobiliary and spleen infections," which is in the predictionary tables and does not yet have a Release Label.

7. Activate the data (MedDRA 8).

8. View the data in the HTML Browser; see "Examining a Term's Details and History" on page 14-32.

Release Label Named Relation Example 2   If you have already loaded and activated MedDRA 8, do the following to trace the relationship between MedDRA 7 high level terms "Liver and spleen infections" and "Gallbladder infections" the single MedDRA 8 high level term, "Hepatobiliary and spleen infections":

1. Create label prefixes for MedDRA 7 and MedDRA 8.

2. Create virtual dictionaries for MedDRA 7 and MedDRA 8. TMS creates a Release Label for each.

3. Define the "is merged into" named relation as above.

4. Define named relations between the two terms in MedDRA 7 and the term they are merged into in MedDRA 8.

5. View the data in the HTML Browser; see "Examining a Term's Details and History" on page 14-32.

Defining Named Relations

To create a new named relation:

1. From the **Definition** menu, select **Define Named Relations**. The Define Named Relations window opens, with the **Named Relation** tab displayed.

> **Note:** By clicking the **Multi Display Named Relations** tab, you can view all of the named relations defined in this TMS installation. Examine the existing named relations before creating a new named relation to avoid creating similar but extraneous relationships.

2. Define the Indicator relationship:

   a. In the **Name** field, enter the relationship name that will appear for relations from a term to its referenced term.

   b. Select the **Many Cardinality** box to allow multiple source terms to relate to one (or more) reference terms using the Indicator relationship.

3. Define the Reciprocal Indicator:

   a. In the **Name** field, enter the relationship name that will appear for relations from the reference term back to the source term. If the relationship does not have a reciprocal, leave the **Name** field blank.

   b. Select the **Many Cardinality** box to allow multiple reference terms to relate to one (or more) source terms using the Reciprocal Indicator relationship.

4. Define the relationship details. All the fields in the **Details** block are optional except for the **Type** field.

   a. In the **Relationship Code** field, you can enter an optional code about this named relation. For example, when loading named relations into TMS via the API, you could populate the **Relationship Code** with the vendor-supplied dictionary from which you loaded the relationship, or the relationship's ID in that dictionary.

   b. Select a **Type**.

   Standard. Use for normal named relations between terms. TMS does not enforce any special behavior.

   Translation. TMS validates that the two terms in the relation are from dictionaries that have different languages specified. Because translation relationships must be between dictionaries of different languages, you can only instantiate such relations as cross-dictionary relations. See "Creating Cross-Dictionary Relations Between Terms" on page 12-23 for instructions on using these relationships in the Repository Authoring window.

   Release Label. Release Label named relations create links between terms in the same or different dictionaries at different points in time; see "Types of Named Relations" on page 7-17.

   c. In the **Short Name** field, enter a unique value for the short name of this relationship. TMS allows names containing up to ten bytes of alphanumeric text. If you leave this field blank and save the record, TMS supplies a default, unique value.

   d. In the **Activation Rule** field, you can specify a function that you want to use as the Activation Rule for this relationship. "Writing Named Relation Activation

Rules" on page 7-21.

    e. In the **Category** field, you can enter a category to define further this named relation.

    f. In the **Description** field, you can provide more information about this named relation. Information entered here could help other TMS users.

5. Save. TMS records the named relation in the repository, and updates the audit fields in the bottom of the window.

You can modify many attributes of named relations in the same manner. Because TMS only uses the internal ID of the relationship to identify which named relation is associated with a particular relation, you can change any information about an existing named relation except its ID, cardinality, and type.

Defining Dictionary NRLs

For each named relation you define, you must specify the dictionaries and dictionary combinations where it should be available for creating named relations between terms.

To specify the dictionaries where a named relation is available for use, do the following:

1. In the Define Named Relations window, with the named relation for which you want to specify dictionaries selected, click **Dictionary NRL**s. The Define Dictionary Named Relationships window opens.

2. Click in the first empty row. The fields become enterable.

3. In the **From Dictionary** column, from the list, select a dictionary in which you want the named relation to be available.

4. In the **To Dictionary** column, from the list, select a dictionary:

- To make the named relation available to define relations between terms in a single dictionary, choose the same dictionary you specified in the **From Dictionary** column.

- To make the named relation available to define relations between terms in the dictionary you specified in the **From Dictionary** column and a second dictionary, choose the name of the second dictionary.

Note: In the **To Dictionary** list of values, TMS displays:

- The From Dictionary; you can always define relations between terms in the same dictionary.

- Dictionaries that have a filter dictionary link with the From Dictionary; one is a filter of the other.

- Dictionaries with a cross-dictionary link defined to the From Dictionary. To define a dictionary link, go to the Define Dictionaries window, select a dictionary, and click the Dictionary Link tab. See "Defining Links Between Dictionaries" on page 6-35.

---

5. Leave **Status** set to **Active**.

You can set the status to **Retired** in the future to prevent this named relation definition's being used to create additional relations in this dictionary or dictionary combination. This is the only modification possible.

You can set the status back to **Active** only if the To and From Dictionaries have one of the relationships described in the above Note.

6. Save.

Writing Named Relation Activation Rules

Named relation Activation Rules can perform further validation of the terms and their relationship(s) before Activation. For example, if you create a named relation "Abbreviation for," you could write a function to verify that the reference term you plan to use as the abbreviation is, in fact, shorter in length than the original term.

Named relation Activation Rules must be part of database packages, and the proper format for definition is package_name.function_name.

The function must use the parameters described in the following table.

Table 7–1   Parameters Required for Named Relation Activation Rules

Parameter (In/Out)	Datatype (Size)	Description
pRelId (in)	NUMBER(10)	ID number of the relation.
pGroupId (in)	NUMBER(10)	ID number of the Activation Group.
pNRLId (in)	NUMBER(10)	ID number of the named relation you are using for this relation.
pConId (in)	NUMBER(10)	ID number of the first term in the relation.
pConRefId (in)	NUMBER(10)	ID number of the reference term in the relation.
pDomId (in)	NUMBER(10)	ID number of the domain in which you are creating this relation.
pCreatedBy (in)	VARCHAR2(30)	The user name of the TMS user who created this relation.
pLevelId (in)	NUMBER(10)	ID number of the level in which the first term resides.
pLevelRefId (in)	NUMBER(10)	ID number of the level in which the reference term resides.
pDML (in)	VARCHAR2(15)	The requested Action you want to perform on the TMS preproduction repository. Values are 'I' (Insert), 'U' (Update), and 'D' (Delete).
pDefDictId (in)	NUMBER(10)	ID number of the dictionary in which the first term resides.
pDefDictRefId (in)	NUMBER(10)	ID number of the dictionary in which the reference term resides.
pRelType (in)	VARCHAR2(15)	The type of relation you are loading: DT – Dictionary Term DPL – Domain Primary Link PPL – Primary Path Link VTA – Verbatim Term Assignment
pRelSubtype (in)	VARCHAR2(15)	If the RelType is VTA, the subtype can either be AC (Accepted) or MS (Misspelled). For any other relation type, possible subtype values are E (External), C (Company), and D (Domain).
pActivationId (in)	NUMBER(10)	Internal ID used to track the Activation process.
pTransId (in)	NUMBER(10)	Not used in this release.
pStatus (in)	VARCHAR2(15)	An optional description of the status of this term.
pCommentText (in)	VARCHAR2(200)	Comment about this Activation Rule.
pErrorMsg (in/out)	VARCHAR2(200)	Error message associated with this record. This is the only part of a production record that you can change with this function.

Defining Informative Note Attributes

This section describes the following topics relating to Informative Notes and Informative Note Attributes:

- Overview of Informative Notes and Attributes on page 7-22

- Defining an Informative Note Attribute on page 7-22

Overview of Informative Notes and Attributes

Informative notes are data structures that you can create to provide more information about a term, relation, VTA, VTI, or dictionary in the database. The advantage that Informative Notes offer over the other detail items—the code, alternate code, category, and the value_1 through value_4 columns—is their flexibility. Using Informative Notes, you can attach a URL, text in a Character Large Object (CLOB), or a note of predefined length containing character, number, mixed, or date data.

Each time you create an Informative Note, you must base it upon an Informative Note Attribute, which gives the note its type, data type, maximum length, and other properties. Using attributes as the basis for Informative Notes saves time during note definition and provides more consistency among the Informative Notes defined in your database.

You can define dictionary-wide Informative Notes that are inherited by all terms in a dictionary. Users can create an Informative Note for a specific term, relation, VTA/VTI, action assignment, or an SMQ algorithm in a filter dictionary such as MedDRA SMQs. See:

- "Defining Dictionary-wide Informative Notes" on page 7-26

- "Creating Informative Notes for Terms and Relations" on page 12-24

- "Using the Status/Notes Pop-up Window" on page 10-21 for creating Informative Notes for VTAs, VTIs, and Action Assignments

- "Defining Informative Notes for SMQ Algorithms" on page 6-7.

In distributed environments, activated attribute definitions are included in replication; once they are replicated, attributes are available for users in all databases in the TMS installation.

Defining an Informative Note Attribute

This section includes:

- Defining Informative Note Attribute Properties on page 7-22

- Defining the Details for the Informative Note Attribute on page 7-24

- Making Informative Note Attributes Available for Dictionaries and Record Types on page 7-24

Defining Informative Note Attribute Properties

Launch the Define Informative Note Attributes window by navigating to **Definition**, then **Define Informative Note Attributes**. The Define Informative Note Attributes window opens, with the **Informative Note Attributes** tab selected. This tab shows detailed information about one Informative Note Attribute, and is the window for defining new attributes. Click the **Multi Display Informative Note Attributes** tab to view several attribute records at once.

The **Properties** block contains the essential information about the attribute: its name, type, and other settings that control its use in TMS.

To define the attribute's properties, from either tab of the Define Informative Note Attributes window:

1. In the **Label** field, enter text to describe the sort of information you store in Informative Notes based on this attribute. Users defining an informative note for a particular object use the label to choose an informative note attribute on which to base the note.The label text appears as a header wherever the Informative Note appears in TMS.

2. From the **Type** list, select:

   ■ **Standard**. For Informative Notes that contain text, a number, or date, to provide supplementary information about a term, relation, VTA, VTI, or dictionary.

   ■ **URL**. For Informative Notes that are character strings representing URLs.

   ■ **Workflow**. For Informative Notes used in workflow processes, such as approval of VTAs. If you are creating an Informative Note during classification, VTA approval, or reclassification, TMS requires that you base the note on an attribute of type Workflow.

   ■ **Algorithm**. For use with filter dictionaries such as MedDRA SMQs. See "Defining Informative Notes for SMQ Algorithms" on page 6-7 for more information.

3. The **Derivable** list choice determines whether external systems will be able to derive Informative Notes that you define based on this attribute. Choose **Yes** to enable external systems to derive this Informative Note data, or **No** to prevent this derivation.

4. Select the **Updateable?** box if you want users to be able to update Informative Notes based on this attribute.

5. Select the **LOV Validation** box if you want TMS to limit the values for a particular Informative Note to those defined in the SQL statement you enter in the **LOV Statement** field in the **Details** panel below. This setting is not applicable if no LOV is generated from the LOV statement, and is not relevant for Informative Note Attributes of data type Memo or of attribute type URL.

6. From the **Data Type** list, choose one of the four available Informative Note data types: **Char** (character), **Date**, **Memo** (for character large objects or CLOBs up to 32K), or **Number**.

   Attribute types Algorithm and URL should always have a data type of Char.

   You can use the Memo data type for either STANDARD or WORKFLOW Informative Note types.

7. In the **Entry Length** field, enter a maximum length for Informative Notes based on this attribute. Entry length is irrelevant for Informative Notes of data type Memo

or Date. If you do not enter a value in this field for a Char or Number Informative Note, TMS supplies the default value of 300 characters.

8. Save.

Defining the Details for the Informative Note Attribute

Attribute details are optional, but you can create a more specific and useful Informative Note Attribute by specifying them for some applications.

To define attribute details:

1. Navigate to the **Details** block on the **Informative Note Attributes** tab to enter attribute details.

2. In the **Attribute Code** field, enter a mixed-case code to use to link this Informative Note Attribute to an attribute defined in a vendor's dictionary.

3. In the **Short Name** field, you can enter a unique value for the short name of this Informative Note Attribute. TMS allows short names containing up to ten bytes of alphanumeric text. If you leave this field blank and save the record, TMS supplies a default, unique value.

4. In the **Description** field, you can enter user-defined text about this Informative Note Attribute.

5. In the **LOV Statement** field, you can enter the SQL statement you want to use to populate a list of values for Informative Notes based on this attribute. The system limits values to those in the LOV only if you select the **LOV Validation?** box, described in the above section, "Defining Informative Note Attribute Properties" on page 7-22.

6. Save. TMS commits this Informative Note Attribute to the database, which enables you to create Informative Notes based on this attribute.

Making Informative Note Attributes Available for Dictionaries and Record Types

Specify the dictionary or dictionaries with which an Informative Note is appropriate for use, and the item—term, relation, term history, dictionary, or filter dictionary algorithm—with which it is appropriate. When users define Informative Notes, the system allows them to select only Informative Note Attributes that you define as appropriate in this window.

1. From the Define Informative Note Attributes window, with the Informative Note attribute displayed (or, in the multi-display tab, selected), click **Dictionary Attrs**. The Define Dictionary Informative Note Attributes Dictionary (APPROVAL) window opens.

2. Click in the topmost empty row. The fields in the row become enterable. If you need more rows than are displayed, select **Insert** from the **Record** menu.

3. In the **Base/Filter Dictionary From** column, click the ellipsis (…) to display the list of values, which includes all base and filter dictionaries defined. Select a dictionary in which you want to be able to use the selected Informative Note attribute.

If the Informative Note attribute is of type Algorithm, TMS displays only active filter dictionaries.

4. In the **Base/Filter Dictionary To** column, click the ellipsis (**…**) to display the list of values, which includes all base and filter dictionaries defined.

- In most cases, select the same dictionary you selected in the **Base/Filter Dictionary From** column. If the Informative Note attribute is of type Algorithm, you must select the same dictionary you selected in the **Base/Filter Dictionary From** column.

- To use the attribute for cross-dictionary relations, select the reference dictionary. You can select a different dictionary only if there is a cross-dictionary link or a filter dictionary link defined for the two dictionaries and only if **Applies To** is set to **Relation** (see next step); see "Defining a Cross-Dictionary Link" on page 6-35 and "Defining Filter Dictionary Links" on page 6-37.

5. In the **Applies To** column, select the type of item for which the user can create Informative Notes from this attribute in this dictionary or dictionary combination.

> **Note:** If an attribute is appropriate for use with more than one item, create a record (add a line) for each one.

- **Content**. If an attribute is appropriate for use in creating an Informative Note on a term, select **Content**. If the Informative Note attribute is of type Algorithm, you must select **Content**.

- **Relation**. If an attribute is appropriate for use in creating an Informative Note for a relation between terms, including terms in different dictionaries, select **Relation**.

- **Term History**. If an attribute is appropriate for applying to term history data; for example, to comment on a change in term status; select **Term History**. See "Term Statuses" on page 10-19 for more information.

  Only Informative Note attributes of type Workflow and data type Memo can be used for term history Informative Notes.

- **Dictionary**. If an attribute is appropriate for use in creating a dictionary-wide Informative Note, select **Dictionary**.

6. In the Status column, leave **Active** selected.

Change to **Retired** later, to prevent any further Informative Notes from being created in this dictionary or dictionary combination with this Informative Note attribute.

7. Save.

Defining Dictionary-wide Informative Notes

When you define an Informative Note for an entire dictionary, each term in the dictionary inherits the Informative Note as well. This inheritance is useful if, for example, you want to associate each term in a dictionary with a URL that contains its name. By defining a single, dictionary-wide Informative Note that includes a variable for the term name, you can create context-sensitive Informative Notes for each term in the dictionary and maintain these notes with a single definition.

> **Note:** Relation records do not inherit Informative Notes from their dictionaries.

You can define dictionary-wide Informative Notes only for strong dictionaries of Active status.

1. Open the Define Dictionaries window and select a dictionary.

2. Click the **Informative Notes** button. The Maintain Dictionary Informative Notes window opens.

3. In the **Informative Note** field, click the ellipsis (…). The list of values opens.

4. Choose an attribute from the list of values and click **OK**. TMS populates the **Informative Note** and **Type** fields with your attribute selection.

   In addition to user-defined Attributes, the list of values includes RDC Action Text. Select this Attribute to create a label for the Special Listings tab in Oracle Clinical Remote Data Capture Onsite Release 4.5.3.10 and above. See "RDC Action Informative Notes" on page 7-27.

5. Enter a **Value**:

   ■ For most **Standard** type notes, enter the text, date, or numeric value to display in the note.

   ■ For **Standard** type notes with a data type Memo, TMS displays the word Memo in blue type. To enter the text:

   a. Double-click the word Memo or select the drill-down option. TMS opens the Memo (*Attribute Name*) window.

   b. Enter or paste the text you want to store in this field.

   c. Click **OK**. The text is stored as a CLOB (character large object).

   ■ For **URL** type notes, enter the URL beginning with a protocol (http:// or ftp://). To use variables, see "Entering Variables in Dictionary-wide URL Informative Notes" on page 7-26.

6. Save.

> **Note:** For information on Informative Notes of type Algorithm, see "Defining Informative Notes for SMQ Algorithms" on page 6-7.

Entering Variables in Dictionary-wide URL Informative Notes

You can use variables to add information in the text of the URL. For example, if you include the variable %TERM% in a dictionary-wide URL Informative Note, TMS substitutes the term name in the text of the URL. Thus, when you define the dictionary-wide URL Informative Note `https://www.google.com/#q=%TERM%`, the term

`rubeola` inherits `https://www.google.com/#q=rubeola` as an Informative Note. You can connect to this URL from the Browse Informative Notes window.

You can insert the following variables into URL values. They correspond to columns in two internal TMS tables.

- These variables correspond to columns in the TMS_DICT_CONTENTS table and apply to term records:
  - TERM
  - TERM_UPPER
  - VALUE_1, VALUE_2, VALUE_3, VALUE_4
  - COMMENT_TEXT
  - CATEGORY
  - DICT_CONTENT_ID
  - DICT_CONTENT_CODE
  - DICT_CONTENT_ALT_CODE

- These variables correspond to columns in the TMS_DEF_DETAILS table and apply to Informative Notes:
  - LABEL_TEXT
  - LABEL_TEXT_UPPER
  - DEF_DETAIL_ID
  - DICT_INFO_HDR_ID

To add a variable to a URL Informative Note:

1. Click in the **Value** field and enter the basic URL.

2. While the cursor is in the **Value** field, press F9 or select **Edit** to see the list of values.

3. Select a variable and click **OK**. TMS appends *%variable%* to any text in the **Value** field.

You can also cut and paste variables in the **Value** field to create a valid URL.

RDC Action Informative Notes

If you have TMS integrated with Oracle Clinical and are using Remote Data Capture (RDC) Release 4.5.3.10 or above, you can use the RDC Special Listings page to display information about a patient's adverse events, concomitant medications, or other data in TMS.

You use a special type of Informative Note to specify the text of the item in the drop-down list on the Home and Casebooks pages for a particular dictionary. By default, this text is: `Review TMS_Dictionary_name / DCM_name`. On the Special Listings page itself the same value (without "Review") appears in the Listing Type drop-down.

You can replace the dictionary name with text that would be more meaningful to RDC users, such as Adverse Events or Concomitant Medications, so that the item in the Home and Casebooks pages' drop-down list would be "Review Adverse Events / *DCM_Name*," for example, rather than "Review MedDRA / *DCM_Name*."

To do this:

1. In the **Informative Notes** field, select **RDC Action Text** from the list of values.

2. In the **Value** field, enter the text you want to appear in the RDC drop-down list, such as Adverse Events or Concomitant Medications.

3. Save.

> **Note:** You can define only one Informative Note of Attribute RDC Action Text for each dictionary. The value text is used in RDC for every study and domain where the dictionary is used.

Defining a Dictionary Version Informative Note

You can use a specialized, predefined Informative Note Attribute called Dictionary Version to assign a version number to a dictionary. To define the version for a dictionary, using an Informative Note:

1. Open the Define Dictionaries window and select a dictionary. You can define a version number for base or virtual dictionaries.

2. Click the **Informative Notes** button. The Maintain Dictionary Informative Notes window opens.

3. In the **Informative Note** field, click the ellipsis (**…**) button. TMS opens the Informative Note Attributes list of values window.

4. Choose **Dictionary Version** from the list of values, and click **OK**. TMS populates the **Informative Note** and **Type** fields with your attribute selection.

5. Enter the version number in the **Value** field.

6. (Optional) Enter a status to define this version number further.

7. Save. TMS validates the entry and saves the Informative Note, returning you to the Define Dictionaries window.

8

Upgrading to a New Dictionary Version

This section includes:

- Upgrading TMS Dictionaries on page 8-1
- Running the Autoprocess VTAs/VTIs Job on page 8-2
- Running Dictionary Upgrade Reports on page 8-3
- Maintaining Predictionary Terms and Relationships on page 8-8

To upgrade to a new version of a vendor-supplied dictionary that you have already defined and loaded, you need to load the new version into the Oracle Thesaurus Management System (TMS) database.

You can use TMS reports to analyze the impact that the new version will have on your VTAs and omissions. You can then make changes—primarily reclassifications—while the new dictionary version is still in the predictionary tables, avoiding problems during and after Activation. A TMS job, Autoprocess VTAs, can correct many problems automatically. You can also correct problems manually.

To run impact reports and make changes in the predict tables, you must have the role TMS_DICTUPG_PRIV. When you have this role, you see a new menu item called Dictionary Upgrade.

This functionality is available only on a master or single TMS instance.

Upgrading TMS Dictionaries

The general steps involved in upgrading a dictionary in TMS include:

1. Load the new dictionary version into temporary tables and then into the predictionary tables. Use the instructions in Chapter 6, "Defining and Loading Dictionaries."

2. Run the **Dictionary Impacts Report** showing current VTAs and omissions that will be impacted by updated, added, or deleted dictionary terms in the new dictionary version. See "Setting Up and Running Reports and Other Batch Jobs" on page 2-13 for more information.

3. Run the **Autoprocess VTAs** job to automatically load the affected verbatim terms into the predict tables if they are not already there and perform the required classifications, reclassifications, and declassifications.

4. Manipulate the data manually if necessary; see "Maintaining Predictionary Terms and Relationships" on page 8-8.

5. Run the specialized dictionary upgrade reports. Return to the **Maintain Predictionary VTAs/VTIs** window to fix problems, rerun the reports, and fix additional problems as necessary.

   The dictionary upgrade reports include:

   - **Predict VTAs/VTIs Report**. This report displays VTAs in the Activation Group you specify in the following groups: classifications, reclassifications, and declassifications. The same data is displayed in the first three tabs of the Maintain Predictionary VTAs/VTIs window. If the Activation Group includes terms and relations in more than one dictionary, you can choose to see them all or specify a single base dictionary.

   - **Site Impacts Report**. This report lists the external system databases from which affected verbatim term occurrences came, with their associated X Area. For Oracle Clinical, this is the study in which the verbatim term occurrence was collected.

   - **Omission Impacts Report**. This report lists omissions that will be resolved (by the addition of a dictionary term that is a direct match) as well as new omissions that will be created (by the deletion of a dictionary term to which verbatim terms are classified) by Activation and Synchronization. This report also shows the actions applied to the current omissions, if any.

   - **Cross-Dictionary Impacts Report**. This report lists terms in other dictionaries that have links defined to affected terms in the new dictionary version.

6. Activate the Activation Group in Check or Transfer mode. See "Activating Loaded Terms."

7. Run the **Preliminary Repository Report** to show errors resulting from the Activation. See "Preliminary Repository Report" on page 6-34 for more information.

8. If necessary, make changes manually using the instructions in "Maintaining Predictionary Terms and Relationships" on page 8-8 and rerun Activation. Repeat this process as necessary.

9. Run a new **Site Impacts Report** to determine which X Areas on which external system sites are impacted. See "Site Impacts Report" on page 8-6 for more information.

10. Run the data exchange job between TMS and the external system for the affected X Areas. If the external system is Oracle Clinical, run Batch Validation for the affected studies on the affected sites.

Running the Autoprocess VTAs/VTIs Job

The Autoprocess VTAs/VTIs job automatically handles many of the changes required by the new dictionary version's impact on your VTAs and VTIs. The job inserts affected VTAs and VTIs into the predictionary tables if they are not already there, so that it can operate on them and you can too, in the Maintain Predictionary VTAs/VTIs window. (The VTAs and VTIs remain in production as well.) The job makes the following changes:

- The job reclassifies VTAs and VTIs whose dictionary term has been moved from one coding level to another.

- The job declassifies VTAs and VTIs whose dictionary term has been deleted

- If you set the **Reclassify Direct Matches?** parameter to Yes, the job reclassifies verbatim terms to new dictionary terms that are a direct match.

To run the Autoprocess VTAs/VTIs job, do the following:

1. In the Navigator window, select **Dictionary Upgrade**, then **Autoprocess VTAs/VTIs**.

2. Enter values for the following Job Specific parameters:

   - **Activation Group**. From the drop-down list, select the name of the Activation Group that contains the new dictionary version.

   - **Reclassify Direct Matches?**. If set to Yes, the job reclassifies VTAs currently to dictionary terms that are not a direct match for which there is a new dictionary term that is a direct match.

3. Submit the job. See "Running a Job" on page 2-16.

Running Dictionary Upgrade Reports

TMS includes the following dictionary upgrade reports:

- Dictionary Impacts Report on page 8-3
- Predict VTA/VTI Report on page 8-5
- Site Impacts Report on page 8-6
- Omission Impacts Report on page 8-7
- Cross Dictionary Impacts Report on page 8-8

Dictionary Impacts Report

The Dictionary Impacts Report displays all the global VTAs, VTIs, and current omissions that will be affected by a base dictionary upgrade. You can run the Dictionary Impacts Report before or after the Autoprocess VTAs/VTIs job. You can run this report only on a base dictionary and in the global domain.

The report includes the following sections:

- **The Assigned Dictionary Term Will Be Deleted for the Following VTAs/VTIs** If the new dictionary version deletes dictionary terms to which VTAs or VTIs are currently classified, you must reclassify the affected terms.

  Each row of the report displays one verbatim term that will be declassified, with the following information:

  - The affected dictionary
  - The affected domain
  - The verbatim term
  - The classification type: VTA or VTI
  - The VTA or VTI ID
  - The dictionary term and its ID

- **The Assigned Dictionary Term Will Be Updated for the Following VTAs/VTIs** If the new dictionary version modifies some dictionary terms to which VTAs or VTIs are currently classified, you may want to reclassify the affected terms.

Each row of the report displays one affected verbatim term, with the following information:

- The affected dictionary

- The affected domain

- The verbatim term

- The classification type: VTA or VTI

- The VTA or VTI ID

- The approval status of the existing VTA or VTI

- The dictionary term and its ID

- **There Will Be a Direct Match for the Following VTAs** If the new dictionary version includes new terms that are a direct match to VTAs that are currently to nonmatching dictionary terms, you may want to reclassify the verbatim terms to the new, matching dictionary terms. (You can do this automatically by running the Autoprocess VTAs/VTIs job with Reclassify Direct Matches set to Yes.)

  > **Note:** If any dictionary/domain combination allows classification to nonapproved dictionary terms, this report includes verbatim terms in that dictionary domain that are a direct match to both approved and nonapproved dictionary terms. This could be the case in MedDRA-J, for example. See "Assigning a Dictionary to a Domain" on page 6-38 for more information.

  Each row of the report displays one affected verbatim term, with the following information:

  - The affected dictionary

  - The affected domain

  - The verbatim term

  - The approval status of the existing VTA

  - The new dictionary term that is a direct match

- **There Will Be a Direct Match for the Following VTIs** If the new dictionary version includes new terms that are a direct match to VTIs that are currently to nonmatching dictionary terms, you may want to reclassify the verbatim terms to the new, matching dictionary terms. (You can do this automatically by running the Autoprocess VTAs/VTIs job with Reclassify Direct Matches set to Yes.)

  If the new, matching dictionary term is unique, it may be a candidate for a VTA. If it is not unique, any auxiliary information is not processed. You must manually classify the VTI if you are using auxiliary information.

  Each row of the report displays one affected verbatim term, with the following information:

  - The affected dictionary

  - The affected domain

  - The verbatim term

  - The approval status of the existing VTI

- The new dictionary term that is a direct match, and its ID

- Whether or not the dictionary term is unique in its level

To generate a **Dictionary Impacts Report**:

1. Select **Dictionary Upgrade** in the navigator window.

2. Select **Dictionary Impacts Report**.

3. Fill in the general report specifications. See "Setting Parameters" on page 2-13.

4. Enter a value for each **Job Specific** parameter:

   - **Activation Group**: Select the Activation Group that contains the new version of the dictionary.

   - **Base Dictionary**: Select the base dictionary being upgraded. If unselected, all base dictionaries in the Activation Group are processed.

   - **Template**: Select the report template. If your company has created a custom template, it appears in the list of values. The Oracle Template is the default template.

5. Submit the job. See "Running a Job" on page 2-16.

Predict VTA/VTI Report

The **Predict VTA/VTI Report** shows the current state of VTAs and VTIs in the predictionary tables for a particular Activation Group and dictionary specified at runtime and indicated in the report header.

Accordingly, the report groups the VTAs and VTIs into the following sections:

- **Classifications**: Displays verbatim terms in the predictionary classification table.

- **Reclassifications**: Displays the terms in the predictionary reclassifications table. These reclassifications may have been done by the Autoprocess VTAs/VTIs job or manually.

- **Declassifications**: Displays terms in the predictionary declassifications table. These declassifications may have been done manually or by the Autoprocess VTAs/VTIs job.

The following columns are displayed in the above report sections:

- Dictionary

- Domain

- Verbatim Term

- Type: VTA or VTI

- VTA or VTI ID

- Dictionary Term and its ID (This is blank for declassifications)

- Old Dictionary Term and its ID (only for Reclassifications)

Note: Only Superusers can run this report

To generate a Predict VTA/VTI Report:

1. Select the **Predict VTA/VTI Report** from the **Dictionary Upgrade** menu.

2. Fill in the general report specifications. See "Setting Parameters" on page 2-13.

3. Enter a value for **Job Specific** parameters:

   - **Activation Group:** Select the Activation Group that contains the new version of the dictionary.

   - **Base Dictionary:** Select the appropriate base dictionary to define the scope of the report. If not selected, all base dictionaries within the selected Activation Group are processed by default.

   - **Template:** Select the template for the report. If your company has created a custom template, it appears in the list of values. The Oracle Template is the default.

4. Submit the job. See "Running a Job" on page 2-16.

Site Impacts Report

The Site Impacts Report displays the X Areas affected by a dictionary upgrade on a particular site.

In Oracle Clinical, an X Area is a study. If you select Oracle Clinical as the external system, the report displays actual study names. If you are using a different external systems, the report displays either the X Area ID or a recognizable name you specify using a function. See "Defining External System Information in TMS" on page 5-28.

Run the Site Impacts Report after the Autoprocess VTA job.

The Site Impacts Report can be run only on the master or single database.

The Site Impacts Report displays the following:

- The external system in use

- The list of X Areas affected by the upgrade

- The corresponding X Area names

- The external system instance

> **Note:** This report includes sites affected by an upgrade because at least one dictionary/domain combination allows classification to nonapproved dictionary terms, and the upgrade includes nonapproved dictionary terms that are a direct match to verbatim terms currently to a different dictionary term. This could be the case in MedDRA-J, for example. See "Assigning a Dictionary to a Domain" on page 6-38 for more information.

To generate a **Site Impacts Report**:

1. Select **Dictionary Upgrade** in the navigator window.

2. Select **Site Impacts Report**.

3. Fill in the general report specifications. See "Setting Parameters" on page 2-13.

4. Enter a value for **Job Specific** parameters:

   - **Activation Group**: Select the Activation Group that contains the new version of the dictionary.

   - **External system**: Select an external system.

- **Remote Instance:** Select a site. The default instance is the current site.

- **Template**: Select the report template. If your company has created a custom template, it appears in the list of values. The Oracle Template is the default template.

5. Submit the job. See "Running a Job" on page 2-16.

Omission Impacts Report

A dictionary upgrade resolves omissions if a new dictionary term is added that is a direct match, and creates new omissions if the dictionary term to which a VTA or VTI is mapped is deleted.

For both omissions that will be resolved and omissions that will be created, the Omission Impacts Report displays the following:

- Domain

- Dictionary

- External system instance

- External system name

- X Area

- X Area name (if available)

- Term

- Number of records affected

> **Note:** For omissions that will be resolved, this report includes omissions that are a direct match to a nonapproved dictionary term, if their dictionary domain allows classification to nonapproved dictionary terms. This could be the case in MedDRA-J, for example. See "Assigning a Dictionary to a Domain" on page 6-38 for more information.

To generate an Omission Impacts Report, do the following:

1. Select the **Omissions Impacts Report** from the **Dictionary Upgrade** menu.

2. Fill in the general report specifications. See "Setting Parameters" on page 2-13.

3. Enter a value for each job-specific parameter:

   - **Activation Group**: Select the Activation Group that contains the new version of the dictionary.

   - **External System**: Omissions may be created from any of the external systems linked with TMS. Select the required system from the list provided here.

   - **Remote Instance**: You can select a remote database if there is one, from the drop-down list. The report runs on the current instance by default.

   - **Base Dictionary**: Select the appropriate base dictionary to define the scope of the report. All base dictionaries within the selected Activation Group are processed by default.

- **Template**: Select the template for the report. If your company has created a custom template, it appears in the list of values. The Oracle Template is the default.

4. Submit the job. See "Running a Job" on page 2-16.

Cross Dictionary Impacts Report

If you have mapped terms across dictionaries using named relationships (NRLs), these mappings will be affected if either of the related terms is deleted. The Cross Dictionary Impacts Report lists all affected relations.

Run the Cross Dictionary Impacts Report before or after the autoprocess VTA job.

It can be run only from the master or single database.

For each affected named relation, the Cross Dictionary Impacts Report displays the following:

- Domain

- The dictionary and dictionary level on the From side of the relation

- The dictionary and dictionary level on the To side of the relation

- The term on the From side of the relation

- The named relationship definition used

- The term on the To side of the relation

- The DML

- The error message (gives the ID of the missing term)

To generate a **Cross Dictionary Impacts Report**:

1. Select **Dictionary Upgrade** in the navigator window.

2. Select **Cross Dictionary Impacts**.

3. Fill in the general report specifications. See "Setting Parameters" on page 2-13.

4. Enter a value for **Job Specific** parameters:

- **Activation Group**: Select the Activation Group that contains the new version of the dictionary.

- **Template**: Select the report template. If your company has created a custom template, it appears in the list of values. The Oracle Template is the default template.

5. Submit the job. See "Running a Job" on page 2-16.

Maintaining Predictionary Terms and Relationships

This section includes:

- About the Maintain Predictionary VTAs/VTIs Tabs on page 8-9

- Common Procedure for Using All Tabs on page 8-10

- Common Fields in the Maintain Predictionary VTAs/VTIs Window on page 8-12

The Maintain Predictionary VTAs/VTIs window shows verbatim terms and their relations in a particular Activation Group, domain, and dictionary that have been modified but not yet activated. They may have been modified manually in this

window or by the Autoprocess VTAs/VTIs job or, in some cases, in the Maintain Repository window or using Maintain Repository APIs.

You do not see data loaded into the predictionary tables (as for a dictionary upgrade) but not modified after being loaded, and you do not see dictionary terms in levels higher than the coding level.

In this window you can work on virtual dictionaries as well as base dictionaries.

All the work you do in this window is in the predictionary tables. You must run Activation to move the terms and relations into production. You can run Activation by running the Activate Preliminary Data job from the Repository Maintenance window or the Repository Authoring window.

About the Maintain Predictionary VTAs/VTIs Tabs

In the Maintain Predictionary VTAs/VTIs window, terms and relations are grouped into tabs according to the type of modification they have undergone in the predictionary tables:

- **Classification**. In the Classifications tab you see verbatim terms that have been in the predictionary tables. You can manually create verbatim terms and classify them to dictionary terms in the new dictionary version. You cannot query for unclassified verbatim terms in this tab; they are in the Invalid tab.

- **Reclassification**. In the Reclassifications tab you can see terms that have already been reclassified in the predictionary tables in the Activation Group, domain, and dictionary you specify. These reclassifications may have been done by the Autoprocess VTAs/VTIs job, manually in this tab, manually in the Maintain Repository window, or using Maintain Repository APIs.

- **Declassification**. In the Declassifications tab you can see terms that have already been declassified in the predictionary tables in the Activation Group, domain, and dictionary you specify. These declassifications may have been done manually by a user in this tab, manually in the Maintain Repository window, or by the Autoprocess VTAs/VTIs job; for example, if a new dictionary version does not include a dictionary term that had been used for classification.

- **VTA Update**. In the VTA Update tab you can see VTs, VTRs, and their dictionary terms (DTs) that have been manually updated in the Maintain Repository window or using Maintain Repository APIs in the Activation Group, domain, and dictionary you specify. A VTA update is a change to the VTA such as a change in the setting for VTA Global? or VTA Approved? or Subtype.

- **Invalid**. The Invalid tab displays verbatim term and relations (VTRs) that are currently invalid and cannot be activated.

  The following verbatim terms and relations are invalid:

  - Relations to dictionary terms that no longer exist

  - Relations to unapproved dictionary terms

  - Multiple relations inserted for the same verbatim term

  - Relations whose VT has a different subtype

  - VT with no parent (either the VTR was deleted without deleting the verbatim term, or a verbatim term was added without creating a VTR)

  Use this tab to reclassify or declassify terms whose relations are currently invalid.

Common Procedure for Using All Tabs

You can use the same general procedures in all tabs of the Maintain Predictionary VTAs/VTIs window. In all tabs, you set the filter and query for one or more terms and relations. You can then, depending on the tab, do one or more of the following:

- Setting the Filter and Querying on page 8-10
- Declassifying Terms on page 8-11
- Reclassifying Terms on page 8-11
- Classifying Terms on page 8-12
- Deleting Terms and Relations on page 8-12

Setting the Filter and Querying

When you begin to work in the Maintain Predictionary VTAs/VTIs window, specify the data you want to view in the fields across the top of the window, as follows. These filter settings apply to all tabs in the window. You can change them at any time.

1. From the **Dictionary Upgrade** menu, select **Maintain Predictionary VTAs/VTIs**, then select the tab you want.

2. **Group**. From the drop-down list, select the name of the Activation Group that includes the new dictionary version.

3. **Domain**. The drop-down list displays the domains associated with the Activation Group you selected. Select the domain you want to work on.

4. **Dictionary**. The drop-down list displays the dictionaries associated with the Activation Group you selected. Select the dictionary you want to work on.

5. Click on the tab you want and enter a query in the upper portion of the window. See "Querying in Windows" on page 2-9.

   TMS returns all the terms and relations that satisfy your query criteria, with information appropriate for each tab; see "Common Fields in the Maintain Predictionary VTAs/VTIs Window" on page 8-12.

6. To make a change, select the term or relation you want to work on by clicking it or tabbing to it.

   To declassify the selected term, see "Declassifying Terms" on page 8-11. To delete the selected term or relation, see "Deleting Terms and Relations" on page 8-12.

   To reclassify or classify the selected term, proceed to the next step.

7. In the **Repository Terms** section at the bottom of the window, query for dictionary terms and VTAs or VTIs to which you may want to reclassify or classify the verbatim term. Use any of the fields as query criteria. For a description of these fields, see"Common Fields in the Maintain Predictionary VTAs/VTIs Window" on page 8-12.

   In addition, for each term type—dictionary terms (DTs), VTAs, and VTIs—select one of the following to limit the query:

   - **No**. If selected for DT, TMS will not search for dictionary terms. If selected for VTA or VTI, TMS will not search for VTAs or VTIs.
   - **All**. If selected for DT, TMS will search for all dictionary terms that match the other query criteria. If selected for VTA or VTI, TMS will search for all VTAs or VTIs that match the other query criteria.

- **Approved**. If selected for DT, TMS will search for only approved dictionary terms that match the other query criteria. If selected for VTA or VTI, TMS will search for only approved VTAs or VTIs that match the other query criteria.

- **Not Approved**. If selected for DT, TMS will search for only unapproved dictionary terms that match the other query criteria. If selected for VTA or VTI, TMS will search for only unapproved VTAs or VTIs that match the other query criteria.

  Depending on the Non Appr DT setting for the current dictionary/domain combination, you may or may not be able to classify a verbatim term to a nonapproved dictionary term. See "Assigning a Dictionary to a Domain" on page 6-38 for more information. You can never create a Global VTA with a nonapproved dictionary term.

8. To reclassify the term, see "Reclassifying Terms" on page 8-11.

   To classify the term, see "Classifying Terms" on page 8-12.

   Note: It is not necessary to Save except for deletions. For classifying, reclassifying, and declassifying, TMS saves your work when you click the **Classify**, **Reclassify**, or **Declassify** button.

Declassifying Terms

When you declassify in this window, TMS deletes the verbatim term (VT) and the verbatim term relation (VTR) from the predictionary tables. You can declassify terms in the Reclassification, VTA Update, and Invalid tabs.

To declassify a term, select it and click

1. Select the term in the upper tab.

2. Click **Declassify**. TMS deletes the values from the New Dictionary Term and New DT ID fields, as well as all the New DT fields in the lower portion of the Reclassification tab.

After you declassify a term and leave the tab, you can find the same term only in the Declassification tab.

Reclassifying Terms

When you reclassify, TMS deletes the current verbatim term relation (VTR) in the predictionary tables and creates the new one in the predictionary tables. You can reclassify terms in any tab except Classifications.

To reclassify a term, do the following:

1. Select the term in the upper tab.

2. In the Repository Terms tab, query for the repository term to which you want to reclassify the verbatim term.

3. Select the term to which you want to reclassify.

4. You can set values for enterable fields. See "Common Fields in the Maintain Predictionary VTAs/VTIs Window" on page 8-12.

5. Click **Reclassify**. TMS populates the New Dictionary Term fields in the upper tab with values that pertain to the new classification.

Classifying Terms

You can classify terms only in the Classification tab, and you can classify only verbatim terms that you enter manually. You cannot query for unclassified verbatim terms.

To classify a term:

1. Enter a verbatim term in the upper tab.

2. In the Repository Terms tab, query for the repository term to which you want to reclassify the verbatim term.

3. Select the term to which you want to reclassify.

4. You can set values for enterable fields. See "Common Fields in the Maintain Predictionary VTAs/VTIs Window" on page 8-12.

5. Click **Classify**. TMS populates the fields in the upper tab with values that pertain to the new classification.

Deleting Terms and Relations

In every tab, you can delete VTs and VTRs from the predictionary tables. The VT and VTR are completely removed from the predictionary tables and therefore are never activated and have no impact on production data.

To delete a term or relation:

1. In the upper tab, select the term or relation you want to delete.

2. From the Record menu at the top of the screen, select Delete.

3. TMS deletes the row.

4. Save.

Common Fields in the Maintain Predictionary VTAs/VTIs Window

This section includes descriptions of common fields in multiple tabs of the Maintain Predictionary VTAs/VTIs window:

- Common Fields in the Upper Tabs on page 8-12

- Common Fields in the Repository Tab on page 8-14

> **Note:** When a value is displayed in blue, you can click it to see additional information.

Common Fields in the Upper Tabs

The following fields appear in one or more of the upper tabs: Classification, Reclassification, Declassification, VTA Update, and Invalid.

Old Dictionary Term   This field displays the dictionary term to which the verbatim term is currently in the production tables.

New Dictionary Term   This field displays the dictionary term to which the verbatim term is currently in the predictionary tables.

Type   In the Classification tab, select the type of classification to be created. **VTIs** are valid only in dictionaries with nonunique terms allowed in the classification level. See

"Nonunique Coding Level Dictionaries" on page 6-8. VTIs cannot be global or nonapproved.

VTA Global?   In the Classification tab, select **Yes** to create the VTA as Global when it is activated or **No** to create the VTA as a Domain VTA in the current domain when it is activated. In the Reclassification and Declassification tabs, this is a read-only field that indicates the nature of the selected VTA.

VTA/VTI Subtype   This is a read-only field that indicates the subtype of the selected VTA: either Accepted or Misspelled.

VTA Approved?   In the Classification tab, select **Yes** to create the VTA as Approved when it is activated or **No** to create the VTA as Unapproved when it is activated. In the Reclassification and Declassification tabs, this is a read-only field that indicates the status of the selected VTA.

VTA/VTI ID   This field displays the unique system-generated ID of the selected verbatim term.

DT ID   This field displays the ID of the dictionary term to which the verbatim term is currently in the production tables.

New DT ID   This field displays the ID of the dictionary term to which the verbatim term is currently in the predictionary tables.

VT Created By   This field displays the user name of the individual who first added the verbatim term.

VTR Created By   This field displays the user name of the individual who first defined the VTR.

DML   This field displays the DML statuses of the repository term. Values—Insert, Update, or Delete—denote the action TMS will take on the term during Activation. If the DML field does not contain a value, the term is currently in the production tables.

Error   If the verbatim term's activation generates an error, it is displayed in the Error field.

DT Content Code   This field displays the unique system-generated ID of the verbatim term.

Old DT Code   This field displays the unique system-generated ID of the term to which the verbatim term is currently in the production tables.

New DT Code   This field displays the unique system-generated ID of the term to which the verbatim term is currently in the predictionary tables.

Predict?   If set to **Yes**, the currently dictionary term is in the predictionary tables. If set to **No** in is in the production tables.

Created By

These fields display the user names of the individuals who first added the old and new dictionary terms.

Level   This field displays the dictionary level of the verbatim or dictionary term.

Subtype   These fields display the subtypes of the old and new dictionary terms.

Approved?   These fields indicate whether the old and new dictionary terms have a status of Approved.

Global?   This field indicates whether the new dictionary term is Global.

Common Fields in the Repository Tab

The Repository tab contains the same fields for Classification, Reclassification, Declassification, VTA Update, and Invalid. These fields include:

Data Source   Select a data source from which to select repository terms. If you select **Dictionary**, TMS displays only data from the production tables. If you select **Predictionary**, TMS displays only data from the predictionary tables. If you select **All Data**, TMS displays both production and predictionary data.

Comment   You may enter comments about your action in this field.

Dictionary Term   This field displays the related dictionary term for the term selected in the lower block. It is determined by the user-defined function for the dictionary, if present, or the immediately related dictionary term if the selected term is a verbatim term, or is null otherwise.

Term   Depending on the options you select in the **Dictionary** and **Data Source** lists, repository terms are displayed in these fields.

DML   This field displays the DML statuses of the repository term. Values—Insert, Update, or Delete—denote the action TMS will take on the term during Activation. If the DML field does not contain a value, the term is currently in the production tables.

Level   This field displays the dictionary level of the repository term in the **Term** field in the same row.

Code   This field is designed to contain the repository term's ID in this dictionary, but its use depends on your company's policy. This field contains a value only if the term is already in the production tables or if a value was entered in the Maintain Repository Data window.

Alt Code   This field is designed to contain a unique ID for the repository term across TMS, but its use depends on your companys policies. This field contains a value only if the term is already in the production tables or if a value was entered in the Maintain Repository Data window.

Type   This field displays the repository term's Type. This value is either **Dictionary Term** or **Verbatim Term**.

SubType   This field displays the subtype for the repository term. This is either **Company**, **Domain**, or **External** for dictionary terms and **Accepted** or **Misspelled** for verbatim terms.

Glb?
This box indicates whether or not the repository term is global.

Appr?   This box indicates whether or not the repository term is approved.

Status   This field indicates the status of the repository term.

Category   This field indicates the category of the repository term.

Id   This field indicates the unique identifier of the repository term.

Comment Text   This field contains comment text for the repository term.

Error Msg
This field displays an error message for activation of the repository term if the term is not activated; otherwise it is null.

Created By   This field displays the user name of the individual who created the repository term.

Valid From   This field displays the timestamp of the creation of the repository term.

Valid To   This field displays the timestamp of the deletion of the repository term.

Trans.id   This field displays the transaction identifier for the repository term.

Value 1   This field displays a detail value defined by your company for the first detail for the repository term, if present.

Value 2   This field displays a detail value defined by your company for the second detail for the repository term, if present.

Value 3   This field displays a detail value defined by your company for the third detail for the repository term, if present.

Value 4   This field displays a detail value defined by your company for the fourth detail for the repository term, if present.

9

Allocating Tasks to TMS Users

This section includes:

-

-

-

-

-

About Task Allocation

You can use Oracle Thesaurus Management System (TMS) to distribute coding and QA tasks—classifying omissions, approving VTAs, and approving Action assignments—to people in your organization. You can also keep track of each person's workload, and each person can track his or her own tasks.

You can set up task allocation to work in several different ways:

- **Manual Pool Allocation**. As omissions, unapproved VTAs, and Action assignments are created, they accumulate in a **pool** until you manually allocate them to other TMS users through either the Allocate Tasks by Quantity or the Allocate Tasks by Term window.

- **Automatic Pool Allocation**. Tasks—omissions, unapproved VTAs, and Action assignments—accumulate in a pool until you invoke a process to allocate them to other users. You allocate only one type of task at a time from one dictionary/domain (or, in the case of VTAs, dictionary/global) combination at a time, to the users you specify. You can select an algorithm that takes into account only the capacity you define for each worker, or an algorithm that uses capacity plus current workload. You run the allocation process from the Allocate Tasks by Quantity window.

- **Direct Allocation**. Omissions, unapproved VTAs, and unapproved Action assignments do not accumulate in a pool. TMS allocates them to users as they are created, according to the user capacities you define.

  TMS performs direct allocation in the following circumstances:

  - on omissions that enter TMS through Batch Validation

  - on terms with associated tasks that enter TMS through Disconnected System Integration (DSI)

  - on VTAs that are created by manually classifying an omission where VTA approval is required

–   on VTAs that are created proactively

–   on actions assigned to omissions where approval is required

However, TMS does not perform direct allocation on omissions or unapproved VTAs or actions that existed before direct allocation was set up, and if a term is manually unallocated, direct allocation does not reallocate it.

Direct allocation will not allocate an approval task to the same user who created the unapproved VTA or unapproved Action assignment.

You can still do automatic and manual pooled allocation, reallocation, and unallocation even when you are using Direct Allocation. TMS prevents two sessions from running automatic pooled allocation at the same time to avoid conflicts. TMS manages conflicts between direct allocation and automatic pooled allocation internally to prevent conflicts.

Using the Task Allocation feature is not required. You specify separately for each type of task—omissions, unapproved VTAs, and unapproved Action assignments—in a reference codelist whether you want to use Task Allocation at all, and if so, whether you want to use direct or pool allocation; see "Setting Task Allocation Reference Codelist Values" on page 9-2.

Approving VTAs and Action assignments are also not required. You specify whether to require these approvals for each domain/dictionary combination; see "Assigning a Dictionary to a Domain" on page 6-38. You can also require approval for VTAs and actions assigned by certain users through the users' profile settings; see "Defining Users" on page 3-5.

> **Note:** You can only require approval for, and therefore allocate tasks for, Answerable Actions. Internal Actions with base Answerable Actions are used for this purpose.

TMS assigns a status to each omission, VTA, and Action as it progresses to each new life cycle stage; see "Term Statuses" on page 10-19.

Setting Up Task Allocation

Setting up Task Allocation includes the following:

■   Setting Task Allocation Reference Codelist Values on page 9-2

■   Adding Workers to the Task Allocation System on page 9-2

Setting Task Allocation Reference Codelist Values

The following installation reference codelists settings affect your Task Allocation setup:

■   In the TMS_CONFIGURATION installation reference codelist, the DEALLOCONLY setting; see "DEALLOCONLY" on page 3-29.

■   The installation reference codelist TMS_TAL_POOL_CONFIGURATION; see "TMS_TAL_POOL_CONFIGURATION" on page 3-30.

Adding Workers to the Task Allocation System

Before you can allocate tasks to users you must add them to the system in the Maintain User Task Allocation Pools window.

Setting Each Worker's Capacity

If you are using either Direct Allocation or Automatic Pooled Allocation, you must specify a **capacity** for each user who will be doing tasks: classifying omissions or approving unapproved VTAs or actions.

For Direct Allocation, you do this in the Maintain User Task Allocation Pools window. For Automatic Pooled Allocation you do it in the Allocate By Quantity window, where you can modify users' capacities each time you run the allocation job if necessary; see "Using Automatic Pool Allocation" on page 9-5.

For direct allocation, the capacity is a number between 0 and 100 that represents a user's capacity for work relative to other workers within the same dictionary or all dictionaries. It is not a percentage, and a worker's capacities for different dictionaries do not need to add up to 100.

For example, four users might have the following capacities:

Table 9–1   Example of Worker Capacities

User	Dictionary 1	Dictionary 2	All Dictionaries
Worker A	90	100	N/A
Worker B	100	80	N/A
Worker C	100	40	N/A
Worker D	N/A	N/A	100

The table indicates that in Dictionary 1, Workers B and C have greater capacity to receive tasks than does Worker A. In Dictionary 2, Worker A has the greatest capacity to receive tasks, Worker B has the next greatest capacity, and Worker C has the least capacity. You can specify a non-dictionary-specific capacity instead, as for Worker D. A worker can have a single capacity for All Dictionaries or for any number of individual dictionaries, but not a combination of All Dictionaries and individual dictionaries. See "Direct Allocation Algorithm" on page 9-9 for more information.

> **Note:**   You can specify a worker's capacity for a dictionary but not a domain/dictionary combination. However, the user's existing Data Access Group security assignments determine the dictionary/domain combination in which tasks can be assigned to the user.

Setting Each Worker's Availability

You can set each worker to Unavailable when he or she is on holiday or in class, for example. No tasks can be assigned to a worker while he or she is Unavailable. However, tasks allocated to the user remain allocated unless they are explicitly deallocated or reallocated.

Maintaining User Tasks Allocation Pools

In the Maintain User Task Allocation Pools window you must list each worker to whom tasks—classifying omissions, approving or rejecting unapproved VTAs and actions—can be allocated.

Adding a User   Do the following to enable a TMS user to have task allocations:

1.   Click in the **Account Name** field to activate the List of Values (LOV).

2. Click the LOV ellipsis (…) on the right of the Account Name field. The Accounts LOV appears. You can enter a search in the Find field.

3. Select the user account of the person you want to add. The system enters the person's account name, first and last name, and specifies whether or not he or she is a superuser.

4. Set **Available?** to Yes when you are ready to start allocating tasks to this user.

5. **Save**.

Specifying Capacities   If you are using Direct Allocation, in the Direct Allocation Capacities section, specify the user's capacity for the dictionaries to which he or she has security access. See "Setting Each Worker's Capacity" on page 9-3 for more information.

1. Click in the **Dictionary** field to activate the List of Values (LOV).

2. Click the LOV ellipsis (…) on the right of the Dictionary field. The Dictionary LOV appears, listing all the dictionaries to which the user has security access. You can enter a search in the Find field.

3. Select a value from the LOV. You can select the value All Dictionaries or you can select the name of a single dictionary. To allow a user to work on multiple dictionaries but not all dictionaries, select a dictionary in as many additional lines as necessary.

4. In the **Capacity** field, enter a number between 1 and 100. See "Setting Each Worker's Capacity" on page 9-3 for more information.

5. **Save**.

Removing a User from the Pool   To remove a user from the task allocation system, so that tasks can no longer be allocated to him or her, do the following:

1. Click the user's **Account Name** to select it.

2. Select **Delete** from the **Record** menu.

3. **Save**.

Allocating Tasks to Workers

You can allocate tasks to workers in several different ways:

- Using Direct Allocation on page 9-4

- Using Automatic Pool Allocation on page 9-5

- Using Manual Allocation by Term on page 9-7

Using Direct Allocation

To use direct allocation, follow the instructions in "Setting Up Task Allocation" on page 9-2. Direct allocation then proceeds automatically, allocating tasks as soon as an omission, unapproved VTA, or unapproved Action assignment is created in any way.

Check the Maintain Direct Allocation Log Files window periodically to see if direct allocation has completed with errors. Each time direct allocation fails, the system enters the information in this window.

For each failed job, the system displays an error log ID, the user ID of the person who ran the job that created the omission(s), VTA(s), or Action(s), and the timestamp of that

job. The system displays a detail ID for the error and an error message in the lower portion of the window for the error log selected in the upper portion of the window. The thi_term_id may appear in the error message.

You can use a SQL*Plus tool to determine the name of the term, domain, and base dictionary from their respective IDs and then correct each problem manually.

Using Automatic Pool Allocation

In the Task Allocation by Quantity window you can invoke an algorithm to automatically allocate tasks. You have more control over the process than with Direct Allocation, but you must explicitly invoke the algorithm. You can allocate only one type of task at a time: omissions, unapproved VTAs, or unapproved Action assignments. You can allocate tasks in only one domain/dictionary combination at a time.

You can specify the workers to whom you want to allocate tasks and set workers' capacities each time you run the job. You can choose either an algorithm that allocates tasks based on workers' current workload as well as their capacities, or use an algorithm that uses only their relative capacities.

Reallocating and Unallocating Tasks   You can use the same method to reallocate tasks from one worker to one or more different workers and to unallocate tasks—that is, remove task assignments from a worker and put the tasks back into the Unallocated pool.

> **Note:**   Unlike direct allocation, automatic pool allocation allows assigning an approval task to the same user who created the VTA or Action assignment requiring approval, if the user has the required privileges.

To use automatic pool allocation, reallocation, or unallocation, do the following:

1. From the **"From" Worker** drop-down list, do one of the following:

    ■ To allocate tasks that are currently unallocated, select UNALLOCATED.

    ■ To reallocate tasks from one worker to one or more other workers or to unallocate tasks that are currently allocated, select the name of the worker from whom you want to remove the tasks.

2. From the **Domains** drop-down list, select either a single domain or All.

    ■ If you select a single domain, the system returns omissions, unapproved Action assignments, or unapproved VTAs that apply only to the domain you select.

    ■ If you select All, the system returns all omissions, unapproved Action assignments, or unapproved VTAs, including unapproved global VTAs.

    In either case, the system returns only unallocated tasks or tasks assigned to a particular user, depending on your selection in the **"From" Worker** field.

3. From the **Dictionary** drop-down list, select either a single dictionary or All (to signify that you want omissions, unapproved Action assignments, or unapproved VTAs from all dictionaries).

4. If you wish, click the **Filter** button to refine the query with information from the external system, or to check the current settings. If you have changed these

settings in another TMS window during this session, the changed settings are still in effect.

In the pop-up window, select the external source data system and database, and then select values for any or all of the fields that are specific to the source data system you select.

5. Click the tab for the type of task you wish to allocate: **Omissions**, **Verbatim Term Assignments**, or **Action Assignments**.

6. Press **F8** or select **Execute Query** from the **Query** menu.

   The system retrieves information on all omissions, VTAs, or Action assignments for the worker or unallocated pool in the domain/dictionary combination you specified at the top of the window, applying any additional external system criteria you specified in the Filter pop-up.

7. Select the domain/dictionary combination you want to work on by clicking in any field in the relevant row.

8. In the **# to give away** column, specify the number of tasks you want to allocate. By default the number in the column is the same as the number of tasks—omissions, VTAs, or actions, depending on the tab—either unallocated or allocated to the worker you selected in the **From Worker** field at the top of the window.

   If you enter a number greater than the number of tasks either unallocated or allocated to the worker you specified, the system stops allocating tasks when there are no more available to allocate. The system gives an error if you enter a negative number.

9. In the **Worker** field in the lower part of the window, the system displays all the workers who are allowed to work on the dictionary/domain combination selected in the Domain and Dictionary fields in the upper portion of the tab. Use the scroll bar to see additional workers or query for a particular worker in the Worker field; see "Querying in Windows" on page 2-9.

   For each worker to whom you want to allocate tasks, enter a number between 0-99999 in the **Capacity** field, inclusive, as the worker's capacity relative to other workers' capacities to whom you are allocating tasks; see "Setting Each Worker's Capacity" on page 9-3. Click **Clr** to reset the number to zero (0) or **Set** to set the number to 100.

   > **Note:** UNALLOCATED is the last value listed in the Worker column. To deallocate tasks, enter any number greater than zero (0) as the capacity for the UNALLOCATED row and 0 for all others.

10. From the **Distribution Algorithm** field, select the algorithm you want to use to allocate the tasks: **Pooled** or **Pooled By Workload**. The Pooled By Workload algorithm takes into account the number of tasks already assigned to a worker as well as the capacity you set for the worker. The Pooled algorithm uses only the capacity.

    Both algorithms consider only the specified dictionary/domain combination and only the type of task currently being allocated.

11. If you wish, click **Calculate Projections**. The system displays the number of tasks that will be allocated to each user if you run the allocation job with the parameters you have entered, but does not actually allocate the tasks. You can make changes and calculate projections as many times as necessary.

When the system calculates the projected allocations, it rounds up to the nearest whole task for each user. When the system actually allocates tasks, it stops when it comes to the last task. Therefore the calculated projection may be slightly different from the actual allocation total for each user, but the maximum difference should be one fewer tasks allocated than projected.

To reset values to zero (0), click the **Clr** button at the top of the column. To reset values to 100, click the **Set** button at the top of the column. To reset the value for a single row, use the **Clr** or **Set** button for that row.

12. Click **Allocate**. The system allocates tasks to the users you specified.

Using Manual Allocation by Term

Use the Task Allocation by Term window to manually allocate, deallocate, or reallocate a single omission, unapproved VTA, or unapproved Action assignment to a particular user.

> **Note:** You can manually allocate multiple omissions, unapproved VTAs, or unapproved Action assignments to a single user in the Task Allocation by Quantity window; see "Using Automatic Pool Allocation" on page 9-5.

> **Note:** Unlike direct allocation, manual allocation allows assigning an approval task to the same user who created the VTA or Action assignment requiring approval, if the user has the required privileges.

To allocate, deallocate, or reallocate a single omission, unapproved VTA, or unapproved Action assignment, do the following:

1. In the **Task Allocation by Term** window, click the tab for the type of task you want to allocate: **Omissions**, **Verbatim Term Assignments**, or **Action Assignments**.

2. Click the Filter button to limit the tasks displayed. See "Setting Filters and Creating Activity Lists" on page 10-24 for instructions.

3. Query for the particular task you want to allocate. You can use the following fields as criteria: Term, Domain, Dictionary, Substatus, and Assigned. The Assigned column includes both workers to whom a task is assigned and the value UNALLOCATED, for tasks not currently assigned to a worker.

   The system retrieves the tasks that meet the criteria you entered.

4. Select the task you want to allocate by clicking it in the **Term** column. You can use Shift+Click to select multiple consecutive tasks or Ctrl+Click to select multiple nonconsecutive tasks.

   Click **Status/Notes** to see its status history and Release Label information in the Status/Notes pop-up window. You can also add a note in the Status/Notes pop-up window. See "Using the Status/Notes Pop-up Window" on page 10-21.

5. Click the **Assign To** field. The List of Values ellipsis (…) appears.

6. Click the LOV ellipsis (…). The List of Values pop-up appears. You can enter a search for a particular user in the **Find** field, using the % wildcard.

7. Select the user to whom you want to allocate the task. Or, to deallocate the task, select UNALLOCATED.

8. Click **Assign**. TMS puts the name of the user to whom the task is now assigned in the Assigned column for the task.

9. Repeat for as many tasks as required.

10. Save.

Tracking Users' Workloads

You can see the current workload for any user in the Task Allocation by Quantity window and by creating a list of activities for any user. In either case, you can only view one task type—omissions, unapproved VTAs, or unapproved Action assignments—at a time.

Tracking Workloads in the Task Allocation by Quantity Window

To see a user's current number of tasks, do the following:

1. In the **Task Allocation by Quantity** window, **"From" Worker** field, select the user whose workload you want to see.

2. Select **All** for both the **Domain** and **Dictionary**. Do not enter a filter search.

3. Press **F8** to retrieve all tasks.

   In the upper portion of each tab, the system lists omissions, VTAs, and Action assignments for the user, and lists the total number for each task type as well.

Tracking Workloads as Activities

You can use the filter pop-up window to create a list of omissions, unapproved VTAs, or unapproved Action assignments for a single user or for all users, for any dictionary/domain combination or for all dictionaries and all domains. If you save the filter as an **activity**, you can then view it at any time from the Activity List window in the Omission Management menu.

The Activity List window always displays a summary of the current results of the query you defined in the filter window, including the task type, dictionary, domain, term status, and current total.

You can double-click the activity name to see the individual tasks retrieved by the query.

You specify filter criteria in the **Filter** pop-up from the appropriate window, as follows:

- To retrieve allocated omissions, go to the Classify VT Omissions window.

- To retrieve allocated unapproved Action assignments, go to the Approve Action Assignments window.

- To retrieve unapproved VTAs, go to the approve VTAs window.

For instructions, see "Creating and Using Activity Lists" on page 10-24.

Task Allocation Algorithms

Direct allocation and automatic pooled allocation work somewhat differently. See:

- Direct Allocation Algorithm on page 9-9

- Automatic Pooled Allocation Algorithm on page 9-9

Direct Allocation Algorithm

The Direct Allocation algorithm is based on the capacities you define for each worker and each dictionary. The algorithm allocates tasks one task at a time as it loops through the set of available workers. During a given loop, the algorithm determines which workers should receive tasks and which should not, based on the workers' capacities.

For example, given the following worker capacities for the same dictionary:

Worker A capacity: 10
Worker B capacity: 20
Worker C capacity: 40

The algorithm works out the relative capacities of the three workers available to work on the relevant dictionary and assigns Worker C twice as many tasks as Worker B and four times as many as Worker A.

The algorithm does not give the first 10 to Worker A, the next 20 to Worker B, and the next 40 to Worker C, for example. Instead, it does something like: 2 to Worker C, 1 to Worker B, 2 to Worker C, 1 to Worker B, 1 to Worker A until all the tasks are allocated.

Direct Allocation does not take current workload into account when assigning tasks.

Automatic Pooled Allocation Algorithm

When you invoke automatic allocation from the Task Allocation by Quantity window, you can select one of two algorithms:

- Pooled on page 9-9

- Pooled by Workload on page 9-9

Pooled

The Pooled algorithm works the same way as the Direct Allocation algorithm (see "Direct Allocation Algorithm" on page 9-9) except that it performs all the allocations at one time (rather than as they are created in TMS) and allocates only the number of tasks you specify to only the users you specify. (In addition, you can allocate only one type of task at a time.)

Pooled by Workload

The Pooled by Workload algorithm also works the same way except that it also considers the current workload of each user (in the current dictionary and for the current task type) so that even if a particular user has a higher capacity for the relevant dictionary, if that user has a larger current workload (number of tasks currently in his or her own pool) then the user receives fewer tasks than another user with a lower workload.

Unfulfillable Assignments Report

You can use TMS to allocate coding and quality assurance tasks among your team members. These tasks include classifying omissions, approving VTAs and approving Action assignments. However, it is possible to assign tasks to users who cannot do them for one or more of the following reasons:

- User lacks Assignee status.

- User has Assignee status but is unavailable.

- User lacks DAG access.

- User lacks database role.

The Unfulfillable Assignments Report displays the following:

- The name of the person assigned the task (assignee)

- The database role of the assignee

- The tasks allocated: VTO, VTA, Action

- The dictionary that is affected by the unfulfilled assignments

- The domain that is affected by the unfulfilled assignments

The Unfulfillable Assignments Report displays the number of counts against each item and the reasons for incompletion. It also shows you the total unfulfilled tasks against each assignee as well as the grand total.

To generate a **Unfulfillable Assignments Report**:

1. Select **Task Allocation** in the navigator window.

2. Select **Unfulfillable Assignments**.

3. Fill in the general report specifications. See "Setting Parameters" on page 2-13.

4. Enter a value for **Job Specific** parameters:

   - **Domain**: Choose the appropriate domain.

   - **Active dictionaries**: Choose the dictionary on which to run the report.

   - **Assignment type:** Choose the assignment type for the report.

   - **User**: Choose a user from the list.

   - **Template**: Select the template you want to use for this report. If your company has created a custom template, it appears in the list of values. The Oracle Template is the default template.

5. Submit the job. See "Running a Job" on page 2-16.

Part II

Managing Classifications

Part II contains the following chapters, each of which covers the tasks in the navigation path of the same name:

10

Omission Management

The windows under the Omission Management menu enable you to approve and unapprove VTAs, set up Autoclassification, and classify terms manually. Oracle Thesaurus Management System (TMS) restricts access to the Approve VTAs window to users with the tms_approve_priv role, and restricts access to all other Omission Management windows to users with the tms_classify_priv role. Omission Management includes:

- Classification Concepts on page 10-1
- Classifying Terms Manually on page 10-3
- Classifying Indication Omissions on page 10-10
- Approving and Unapproving VTAs on page 10-11
- Approving Action Assignments on page 10-14
- Term Statuses on page 10-19
- Using the Status/Notes Pop-up Window on page 10-21
- Using the VT History Window on page 10-23
- Creating and Using Activity Lists on page 10-24
- Omission Management Reports on page 10-28

Classification Concepts

The process of assigning a source term—a term collected during a clinical trial as a response to a CRF question—to a dictionary term is called *classifying* the term. After loading a group of source terms, you run the Autoclassification process, which classifies as many terms as possible automatically:

- If a dictionary term exists that is exactly the same as the source term and is unique in the coding level, the process classifies the source term to the dictionary term, creating a *Verbatim Term Assignment* (**VTA**).

- If a dictionary term exists that is exactly the same as the source term and is NOT unique in the coding level but has auxiliary information, the process can classify the source term with the same auxiliary information to the dictionary term, creating a *Verbatim Term Individual* (**VTI**); see "Nonunique Coding Level Dictionaries" on page 6-8 for more information.

- If an existing VTA—either automatically or manually created—matches the source term, the process classifies the source term to the VTA.

- If an existing VTI—either automatically or manually created— matches the source term and its auxiliary information, the process classifies the source term to the VTI.

If Autoclassification cannot classify a term, it creates an *omission*. A user must classify each omission, manually creating a VTA or VTI. Autoclassification then classifies each subsequent occurrence of the same source term to the manually created VTA or VTI.

A user can classify an omission to an existing VTA, but not to an existing VTI. When a user classifies a term to an existing VTA, behind the scenes TMS maps the verbatim term directly to the dictionary term specified by the VTA.

See "Autoclassification" on page 3-35 for more information.

Misspelled VTAs and VTIs   It is possible to designate a VTA or VTI as misspelled, and a dictionary term can have any number of misspelled VTAs and VTIs; for example, the dictionary term acetyl salicylic acid might have misspelled VTAs asperin and aspurin as well as (not misspelled) aspirin. When you classify a term, you must select either Misspelled or Accepted (not misspelled).

Approved and Unapproved VTAs   In each dictionary/domain you specify whether you want VTAs to be created as Approved or Unapproved. Unapproved VTAs require explicit manual approval for each occurrence.

- If VTAs are created as Approved, no further action is required. Autoclassification maps all occurrences of the term to the Approved VTA. However, it is possible to declassify or reclassify a verbatim term.

- If VTAs are created as Unapproved, all occurrences of the term are mapped to the Unapproved VTA until a user explicitly approves the VTA. All subsequent occurrences are mapped to the Approved VTA.

> **Note:**   VTIs—Verbatim Term Individuals, classifications to a nonunique dictionary term—are always created as Approved.

Global and Domain VTAs   A **Domain VTA** is available for use in classification only in the TMS domain in which it was created. A **Global VTA** is available across all TMS domains. Domain VTAs take precedence over Global VTAs: if a domain version of a VTA exists in the current TMS domain, you will see only it, even if a global version also exists.

You can *promote* a Domain VTA to be a Global VTA or *demote* a Global VTA to a Domain VTA in the Promote/Demote VTAs window.

If you have a distributed environment, you can classify omissions on any location but you must set reference codelist values and promote and demote VTAs only on the master instance.

Classification and Verbatim Term Levels   You can classify omissions to terms in either the classification level of the dictionary, which are dictionary terms, or to terms in the Verbatim Term level, which are previously classified terms (VTAs). If you select a VTA, behind the scenes TMS instead classifies the verbatim term directly to the dictionary term to which the VTA you selected is classified.

Candidate Terms and Search Objects   Your company may set up Search Objects, or algorithms, to supplement the basic Autoclassification process to find possible valid dictionary matches in an automated way. These are presented as *candidate* terms.

Search Objects are executed in order and run only until candidate terms are found, and the Search Object that found them is displayed in the **Search** field. You can run Search Objects during manual classification. Unless there have been changes to the database since Autoclassification, you will not find any candidate terms by using a search object whose execution order is lower than the one entered in the **Search** field. However, you may find additional matches by using a search object with a higher execution order. You can see the order of execution in the Define Search Objects window under the Definition menu.

Actions   Sometimes you cannot manually classify a term because the term itself is flawed. For example, a term might include two symptoms that cannot be mapped to a single dictionary term, such as *headache and nausea*. Or a term may be garbled because a data entry operator's fingers were on the wrong keyboard keys; for example *sd[otom* instead of *aspirin*. In this case, you can assign a predefined Action to the term and send it back to the external source data system. Actions contain a message; for example, *Please split the term* or *Please clarify*. See "Defining and Using Actions" on page 7-6.

Internal Actions for Unapproved Actions   To minimize the time spent on communication with the external system, you can require that Action assignments be approved before being sent back to the source system. In this case, you allocate an *Internal Action* as a task to a TMS user to approve an external Action before sending it to the source data system. See "Defining and Using Actions" on page 7-6.

Discrepancy Message   You can send a message to the external system regarding a single source term in the Classify VT Omissions window. A discrepancy message is not predefined; you enter text specifically for a particular source term. See "Applying a Discrepancy Message" on page 10-8.

Activities   Your organization can assign tasks, including classifying omissions and approving unapproved VTAs and actions, to you and other users. You can create lists of the tasks assigned to you and use them to organize your daily work; see "Creating and Using Activity Lists" on page 10-24.

Term Statuses   TMS automatically assigns statuses to terms—omissions and unapproved VTAs—as they go from one stage to the next and become approved VTAs or VTIs. These statuses reflect whether or not they are allocated to a user and whether or not an Action is associated with them. Most statuses have substatuses. See "Term Statuses" on page 10-19 for more information.

Classifying Terms Manually

You resolve omissions—terms that TMS could not classify automatically—by manually classifying the term. The type of classification you can create, and the system's behavior after your classification, depend on how the dictionary is set up by your organization.

VTA   If the dictionary allows only unique dictionary terms in the classification level, you classify an omission by creating a Verbatim Term Assignment (**VTA**). TMS automatically classifies all future occurrences of the same term to the same VTA. VTAs can be created as global or domain-specific, approved or not approved, misspelled or accepted.

VTI with Auxiliary Information   If the dictionary allows nonunique classification-level terms and includes auxiliary information for dictionary terms, like WHO-Drug Format C, you classify an omission by creating a Verbatim Term Individual (**VTI**). TMS

automatically classifies all future occurrences of the same term with the same auxiliary information to the same VTI. VTIs are always approved and domain-specific. They can be misspelled or accepted.

VTI without Auxiliary Information   If the dictionary allows nonunique classification-level terms but has no auxiliary information—which is possible for MedDRA if nonunique dictionary terms are related to different higher-level dictionary terms—you classify an omission by creating a Verbatim Term Individual (**VTI**) in the All Verbatim Term Omissions tab. The classification applies only to the occurrence you code.

Actions   If you cannot classify a term because the term itself is flawed, you can request that it be corrected by using an **action**.

1. From the Omission Management menu, open Classify VT Omissions.

2. Query for Unclassified Verbatim Terms

3. Do one:

   ■   Look for a Good Match and Classify a Term

   ■   Apply an Action to a Term if the verbatim term is faulty

In either case you can add an Informative Note to the term; see "Creating an Informative Note for the Classification" on page 10-9.

Query for Unclassified Verbatim Terms

Query for verbatim terms that TMS was not able to autoclassify or that a TMS user has declassified. These are called *omissions*.

■   The **Distinct Verbatim Term Omissions** tab lists each distinct verbatim term omission that meets your query criteria. If the same term has been entered for ten patients, the term is displayed once.

■   The **All Verbatim Term Omissions** tab lists every source term—verbatim term occurrence—that meet your criteria. If the same term has been entered for ten patients, all ten occurrences are displayed. Additional information about each occurrence is also displayed. Query in the Distinct tab, select the term, and view all occurrences in the All tab.

Action and Note Indicators   In both Verbatim Term Omissions tabs an "A" icon appears beside verbatim term omissions that have an associated Action.

An "N" icon indicates that there is a note associated with the term.

Figure 10–1   Action Indicators in the Distinct Verbatim Terms Tab

To query for unclassified verbatim terms:

1. From the **Omission Management** menu, select **Classify VT Omissions**. The Classify VT Omissions window opens.

2. (Optional) Select the Omission Filter settings you want to use for this session. See "Setting Filters" on page 10-25 and "DT Filter for Auxiliary Information" on page 10-9 for more information. To filter again, click the **Filter** or **DT Filter** button.

   TMS displays the query results in the **Distinct Verbatim Term Omissions** tab.

3. In the **Distinct Verbatim Term Omissions** tab, select a verbatim term.

   - If the dictionary has nonunique terms in the classification level and auxiliary information, like WHO-Drug Format C, open the **All Verbatim Term Omissions** tab. When you select a term, its auxiliary information is displayed at the bottom of the tab.

   - If the dictionary has nonunique terms in the classification level and no auxiliary information, you have two choices:

     – Select a term in the **Distinct Verbatim Term Omissions** tab. The classification you make applies to all existing occurrences of this term.

     – Select a term in the **Distinct Verbatim Term Omissions** tab and click the **All Verbatim Term Omissions** tab to see all occurrences of the term. Select an occurrence. The classification you make applies only to this occurrence.

   - If the term is flawed in such a way that it cannot be classified, apply an action to it. See "Apply an Action to a Term" on page 10-7.

   - Otherwise, proceed to Look for a Good Match.

Look for a Good Match

Search for a dictionary term or VTA to which to classify the verbatim term. TMS displays classification level dictionary terms and VTAs in the bottom block. Use one or more of the following methods to find a good match, and then highlight it.

1. Highlight the term you want to classify in the **Distinct** or **All Verbatim Term Omissions** tab above.

2. If the term's **Search** field has a value in the upper block, double-click the term. In the Classifications tab, TMS displays the candidate terms found during Autoclassification by the Search Object algorithm listed in the **Search** field. One of these is likely to be a good match for the verbatim term.

   If so, click **Classify VT**. If not, continue to search as follows.

3. Select one or more of the following in the **Classifications** block to narrow your query results:

   - **Query**

     – Select **Standard** to conduct a standard Oracle query, using the wildcard `%` for multiple characters and `_` for single characters.

     – Select **Context** to conduct a query using Oracle *inter*Media Text retrieval capabilities. See "Querying in Windows" on page 2-9 for more information.

   - **Search Type**. Choose one of the user-defined search objects, or algorithms, if any, or **Open Query**. See "Candidate Terms and Search Objects" on page 10-2.

   - **Dictionary Term** and **Level**. If a VTA is selected in the **Classifications** block, TMS displays one of the following in the **Dictionary Term** and **Level** fields:

- If you have not defined a Dictionary Term Display Procedure, the **Dictionary Term** field shows the term to which the selected VTA is mapped.

- If you specified a procedure in the **Dictionary Term Display Procedure** field in the Define Dictionaries window, the **Dictionary Term** field shows the term yielded by this procedure, and the **Level** field displays that term's level. See "Writing a Procedure to Change the Dictionary Display" on page 3-39.

■ At the bottom of the Classifications tab, select either **Current** or **All** data. If you select **All**, you see dictionary terms and VTAs that have been deleted as well as current ones.

■ **DT (Dictionary Terms)**. At the bottom of the Classifications tab, select either:

- **No** to see only VTAs.

- **All** to see all dictionary terms that match the other criteria.

- **Approved** to see only approved dictionary terms that match the other criteria.

- **Not Approved** to see only unapproved dictionary terms that match the other criteria.

Note: Depending on the Non Appr DT setting for the current dictionary/domain combination, you may or may not be able to classify a verbatim term to a nonapproved dictionary term. See "Assigning a Dictionary to a Domain" on page 6-38 for more information.

■ **VTA**. At the bottom of the Classifications tab, select either:

- **No** to see only dictionary terms.

- **All** to see all VTAs that match the other criteria.

- **Approved** to see only approved VTAs that match the other criteria.

- **Not Approved** to see only unapproved VTAs that match the other criteria.

Note: VTIs are not displayed in the Classifications tab.

■ Click the **DT Filter** button to filter on VT auxiliary information, if any. See "DT Filter for Auxiliary Information" on page 10-9.

4. Execute the query by pressing **F8** or selecting **Execute** from the **Query** menu. Results appear in this block.

> **Note:** in addition, you can use these buttons:
>
> - Highlight a term and click **Above** to see all of a term's relations in levels above. See "The Above Button" on page 10-10.
>
> - Highlight a term and click **Below** to see all of a term's relations in levels below. See The Below Button on page 10-10.
>
> - Select **Extended Search** to open the Extended Search window. See "Using the Extended Search Button" on page 10-9.
>
> - Reset the filter. See "Setting Filters" on page 10-25.
>
> - Filter on auxiliary information. See "DT Filter for Auxiliary Information" on page 10-9.

Classify a Term

To classify a term:

1. In the upper block, select the verbatim term to classify. In the Classifications tab, select the dictionary term or VTA to which you want to classify it.

2. The **Global?** box is selected or deselected by default according to settings elsewhere in TMS. You can override the default for a particular VTA:

   - Select the **Global?** box to make the VTA available in all domains where a Domain VTA for the same term does not exist.

   - Deselect the **Global?** box to make the VTA available only in the current domain.

3. If the verbatim term is spelled correctly, select **Accepted** from the **VTA/VTI Subtype** list. If not, select **Misspelled**. Future occurrences that are misspelled the same way will be classified the same way. Specifying whether the verbatim term is spelled correctly when you classify provides more accurate records.

4. Enter a **Comment** indicating why you chose this term for the classification.

5. Click **Classify VT** or **Create VTI** (available only when using the All... tab above). TMS enters the repository term to which you have linked the verbatim term in the **Dictionary Term** field in the upper block.

   > **Note:** If you make a mistake during manual classification or a dictionary term is added that is a better match for a classified verbatim term, you can reclassify the term (see "Reclassifying and Declassifying Verbatim Terms" on page 11-6).

Apply an Action to a Term

If you cannot classify a term, it may be appropriate to assign an Action to it instead. Use actions in the following situations:

- If the verbatim term is flawed, assign an Action to the term requesting that the term be fixed. TMS sends the Action text back to the external source system. In Oracle Clinical, the next Batch Validation creates a discrepancy for the question response with the Action text as the comment and the TMS user (you) as the creator of the comment. See "Answerable Actions" on page 7-6 for more information.

Approval of Answerable Action assignments may be required for particular dictionary domain; see "Approving Action Assignments" on page 10-14.

- You may need to send a message to the external system that does not require a response, or you may not have the resources to support a dialog. See "Unanswerable Actions" on page 7-7.

- If you need to speak to your supervisor or do research before you can classify a term, you can assign an Action to the term that says you are working on it. When other coders see the Action text, they will know not to duplicate your efforts. See "Internal Actions" on page 7-7.

You must select an Action that has already been defined (see "Defining and Using Actions" on page 7-6). Actions apply to all occurrences of the same verbatim term omission (VTO) in the same dictionary/domain combination.

To send a message to the external system about a single occurrence of a verbatim term, you can use a discrepancy message. See "Applying a Discrepancy Message" on page 10-8 for instructions.

Applying an Action

To apply an Action to a verbatim term:

1. In the top block of the Classify VT Omissions window, query for the verbatim term to which you want to apply an Action. Select the term.

2. In the bottom block, click the **Distinct VTO Actions** tab to open it.

3. From the **Action** drop-down list, select the Action you want to apply to the omission. TMS displays the Action type and the default Action text in the corresponding fields to the right; see "Action Types" on page 7-6. You can modify the Action text if necessary.

   TMS displays only actions that are valid options for the current user. TMS displays only Internal Actions and Unanswerable Actions to users without the Approve or Reclassify privilege or the overriding profile setting. Users with the Approve privilege, or Reclassify privilege, or overriding profile setting can see all active actions in the drop-down.

4. Click **Apply Action**. TMS assigns the Action to the term and lists the Action in the lower part of the tab. The action applies to all occurrences of the verbatim term.

   > **Note:** If you are working in a dictionary with nonunique terms in the classification level, the system does not apply the action to occurrences of the verbatim term that have been classified to a VTI. For example, if Tylenol with Country equal to France has been classified but Tylenol with Country equal to any other value has not been classified, the action is not applied to Tylenol with Country equal to France but is to all other occurrences of Tylenol.

5. Save.

Applying a Discrepancy Message

To send a message about a single verbatim term occurrence to the external system, use a Discrepancy Message. This may be helpful if an external system user responds to an Answerable Action for a particular verbatim term occurrence.

1. In the **All Verbatim Term Omissions** tab, query for the verbatim term to which you want to apply a Discrepancy (or select the verbatim term in the **Distinct Verbatim Terms** tab, and then click the **All Verbatim Term Omissions** tab and select the correct occurrence).

2. In the appropriate verbatim term's row, click in the **Omission Status** field and press F9. The **Omission Status** dialog box opens, with a list of valid omission statuses.

3. Select the appropriate omission status. It must be a status that the external system will recognize.

4. Click **OK**. The **Omission Status** dialog box closes, and TMS populates the **Omission Status** field with your choice.

5. Enter a comment about this omission in the **Action Text** field.

6. Save. The next time you run the batch job for data exchange with the external system—Batch Validation in the case of Oracle Clinical—TMS sends this text to the external system.

Creating an Informative Note for the Classification

You can apply a note to the verbatim term or see the term's current status and note history by clicking the **Status/Notes** button or selecting **Status/Notes** from the **Options** menu. See "Using the Status/Notes Pop-up Window" on page 10-21 for more information.

DT Filter for Auxiliary Information

Click the **DT Filter** button to search for dictionary terms using specific values in auxiliary information associated with dictionary terms; for example, in WHO-Drug C, Pharmaceutical Form, Strength, Strength, or Country.

Enter one or more values and click **OK**, or click **VTO Autofill** and the system displays the auxiliary values for the omission currently selected in the All Verbatim Term Omissions tab.

For example, to search for a dictionary term for the drug Carbamazepine marketed in Spain:

1. Click **DT Filter**.

2. Enter Spain in the Country field.

3. Click **OK**. The system displays the value "Spain" in the DT Filter line at the bottom of the tab.

4. Press F7 to enter query mode.

5. Enter Carbamazepine in the Term field in the Classifications tab.

6. Press F8 to execute the query. The system returns any existing dictionary terms for Carbamazepine that include Spain as the Country value.

Using the Extended Search Button

Use this window to search all domains and levels for a match to a verbatim term.

1. From the **Dictionary** list, choose the dictionary you want to search. TMS displays the current dictionary by default.

2. From the **Search Type** list, choose a user-defined search algorithm or an open query.

   If you choose a **Search Type** other than **Open Query**, TMS populates the **Verbatim Term** field with the current term from the Classify VT Omissions window and uses it as a Filter in the query. You can modify it, but you cannot effectively delete it. The search algorithm must have a verbatim term parameter and it will use the last one entered.

   If you choose a **Search Type** of **Open Query**, TMS does not use a value entered in the **Verbatim Term** field.

The Above Button

Highlight a term in the **Classifications** block and select **Above** to see all the term's relations in higher dictionary levels. These are displayed in the tree structure with the lowest-level term at the top left and the highest-level term at the bottom right. Click any related term in the tree structure to see more information in the **Contents** tab. Click the **Relations** tab to see information about the relation between the highlighted term and the next lower term.

The Below Button

Highlight a term in the **Classifications** block and select **Below** to see all the term's relations in lower dictionary levels. These are displayed in the tree structure with the highest-level term at the top left and the lowest-level term at the bottom right. Click any related term in the tree structure to see more information in the **Contents** tab. Click the **Relations** tab to see information about the relation between the highlighted term and the next lower term.

In WHO-Drug C, the levels below contain the auxiliary information for the dictionary term.

The Indication Button

If you click the Indication button from Manage VT Omissions, the Classify Indication Omissions window opens, displaying indication omissions for the current term, if any.

Classifying Indication Omissions

An indication omission can occur in a dictionary like WHO-Drug Format C where a single dictionary term may be mapped to multiple terms in the next higher derivable level, and the correct derivation depends on the indication for which the drug is being taken by a patient.

The **Distinct Indication Omissions** tab displays each verbatim term with its indication, the level up to which terms can be successfully derived, the term that is derived in that level, and the next higher level from which a term should be derived. This tab displays each verbatim term/indication combination once.

1. Query for a verbatim term, dictionary term, and/or indication in the **Distinct Indication Omissions** tab and select it. You can see all occurrences of the same verbatim term/indication combination in the **All Indication Omissions** tab.

   The **Higher Level Terms** tab automatically displays higher-level terms that are related to the dictionary term selected in the Distinct Indication Omissions tab.

2. Select the higher-level term to which you want to classify the lower-level dictionary term for its verbatim term/indication combination.

3. Click **Classify**. The system will derive the high-level term for all existing occurrences of the same verbatim term with the same indication the next time data is exchanged with the source data system, and all future occurrences after they appear in TMS.

Approving and Unapproving VTAs

If a verbatim term assignment (VTA) is approved, the classification is immediately available for derivation and Autoclassification. Once a VTA is approved, no new occurrences of the same verbatim term show up as omissions.

If a VTA is not approved, the user must manually classify each new occurrence of the verbatim term, but TMS proposes the unapproved VTA's dictionary term as the only Candidate Term. The user must either accept the Candidate Term or reclassify the verbatim term (see "Global and Domain VTAs" on page 11-2). Working with Nonapproved VTAs gives you an opportunity to review classifications that TMS would otherwise create automatically, may give you a better feel for the data and help you catch mistakes.

You can approve Nonapproved VTAs when you are satisfied that the classification is correct and no longer want to approve each occurrence of the verbatim term (see "Autoclassification" on page 3-35). You can also undo a VTA's approval to change the assignment of new occurrences of a verbatim term.

The reference codelist TMS_CONFIGURATION contains a setting that determines whether Nonapproved VTAs are or are not be used for derivation to the external system. If Nonapproved VTAs are derived, the external system sees no difference between derivations based on approved and Nonapproved VTAs.

For details on creating VTAs, promoting them to global status or demoting them to domain-specific status, see Chapter 11, "VTA and VTI Maintenance."

This section includes:

- Understanding the Approve VTAs Window on page 10-11
- Changing the Subtype and Adding Comments on page 10-13
- Approving a Nonapproved VTA on page 10-13
- Declassifying a VTA on page 10-14
- Browsing the Audit Trail on page 10-14

Understanding the Approve VTAs Window

You start to change the approval status of a VTA by querying for it in the upper part of the Approve VTAs window. There are two blocks in this part of the window:

- The **Distinct Verbatim Term Assignments** block shows the unique verbatim terms from this database that have been to terms in the selected dictionary. This block also displays information about the dictionary term to which this verbatim term is currently mapped.

- By contrast, the **All Verbatim Term Assignments** block shows verbatim term information for each instance of each verbatim term introduced to TMS by the selected database, and provides information from the external system about these verbatim terms.

For each VTA they describe, the **Distinct** and **All Verbatim Term Assignments** blocks display the following information about a VTA.

Appr?   If selected, the VTA is approved for use in Autoclassification and manual classification. If not selected, TMS proposes the VTA as a Candidate Term but manual approval is required for each occurrence.

Glb?   (Global?) If selected, the VTA is available in all domains.

Subtype   The VTA is either **Misspelled** or **Accepted** (correctly spelled).

Origin   How this term was introduced to this TMS dictionary.

Dictionary Term   (Distinct block only) By default, this field shows the dictionary term to which the verbatim term is assigned, if any.

The remaining blocks on the **Distinct Verbatim Term Assignments** block provide read-only information about the term displayed in the **Dictionary Term** field. For details on any of these fields, see "Current Level and Term" on page 12-3.

Value 1 - 8 Fields   (All VTA blocks only) When an external system is fully integrated with TMS, the Approve VTAs window displays column information for verbatim terms originating in that external system. You define these columns in the Define External Systems window as the **Value 1** through **Value 8** fields for that external system. In addition, when you define detail information for a column, TMS displays the text in the field corresponding to the column in blue. The user can then use the drill-down function to see detail information. See "Defining External System Information in TMS" on page 5-28.

Filter   Click the Filter button to change the filter settings to refine query results from this window. You can save filter settings as an activity list for future use. See "Setting Filters and Creating Activity Lists" on page 10-24 for more information.

Above Current VTA   The system displays dictionary terms related to the current term in the levels above the current term. The tree is displayed upside-down: the VTA is displayed at the top, with the related terms in the next higher level displayed just below it, and the related terms in higher levels below those terms.

Declassify   To declassify a VTA rather than approve it, select the nonapproved VTA and click **Declassify**; see "Declassifying a VTA" on page 10-14 for more information.

> **Note:**   You must have either TMS_CLASSIFY_PRIV or TMS_ RECLASSIFY_PRIV to be able to declassify a VTA. Users with TMS_ RECLASSIFY_PRIV can declassify any VTA at any time.
>
> Users with TMS_CLASSIFY_PRIV, can declassify only terms they have personally classified, and only within a limited (configurable) time period. This functionality allows users who classify a term to an Unapproved VTA and then decide they have made a mistake, to query for the term in this window and undo the mistake by declassifying the term.

Declassify and Action   To declassify a VTA and add an Action to the term at the same time, click **Declassify and Action**. See "Apply an Action to a Term" on page 10-7 for more information.

Comment   (Distinct block only) Further user-defined information about the VTA. See "Changing the Subtype and Adding Comments" on page 10-13.

Status/Notes   (Click the **Status/Notes** button to see the history of all statuses the term has had in TMS and all notes ever assigned to the term. You can also add a new note to the term. See "Using the Status/Notes Pop-up Window" on page 10-21 for more information.

Changing the Subtype and Adding Comments

You can change a VTA's subtype or add a comment about it at any time, even if you are not changing that VTA's approval status. The **Verbatim Term Assignments** block displays the most recent comment about a VTA in that VTA's **Comment** field, on the right side.

You can change a VTA's subtype from **Misspelled** to **Accepted** or vice versa from the **Verbatim Term Assignments** block. In the VTA's row, change the value in the **Subtype** field, then save. An LOV is provided.

To enter a comment for a VTA, type the desired text in the **Comment** field for that VTA's row, and save. TMS retains your comment in the field, and updates the Audit Trail with your change.

Approving a Nonapproved VTA

To change a Nonapproved VTA to Approved status:

1. From the **Omission Management** menu, select **Approve VTAs** to reach the Approve VTAs window.

    If this is your first visit to one of the windows Approve VTAs, Classify VT Omissions, or Reclassify Verbatim Terms, the Filter window opens. Choose your Filter criteria by following the instructions in "Setting Filters and Creating Activity Lists" on page 10-24, and click **OK**.

2. Query for the VTA in either of the upper tab windows. If you use the Distinct Verbatim Term Assignments window, select the **Not Approved** option before entering your query criteria.

    The window returns all of the VTAs that match your query and Filter settings.

3. Click the **Appr?** box for the VTA you want to approve.

4. (Optional) Enter additional information about this change in VTA approval status. The Approve VTAs window provides two methods for adding context to these types of changes:

    -   In the **Verbatim Term Assignments** block, scroll to the right and enter a comment about this change to the VTA's approval status in the **Comment** field in that VTA's row. These comments can be helpful when you audit changes to a VTA in the **Audit Trail** tab window.

    -   Define one or more Informative Notes for this transaction. See "Using the Status/Notes Pop-up Window" on page 10-21 for more information.

5. Save.

Changing an Approved VTA to nonapproved status works the same way; query for the appropriate term and deselect the **Appr?** box to set it to nonapproved.

Declassifying a VTA

If you do not want to approve a nonapproved VTA, you can declassify it in the Approve VTAs window. The term becomes an omission and can be classified to a different dictionary term in the Classify VT Omissions window.

You can add an Action to a term as you declassify it. If the term itself is flawed and you need more information from the source system where the term was collected, you can assign an answerable external Action to the term and send its text to the external system; see "Apply an Action to a Term" on page 10-7 for information about actions.

To declassify a term without assigning an Action to the term, click **Declassify** or select **Declassify** from the **Options** menu.

To declassify a term and assign an Action to it, do the following:

1. Select the VTA you want to declassify.

2. Click **Declassify and Action** or select **Declassify and Action** from the **Options** menu. The **Action** pop-up window opens with information about the VTA displayed across the top.

3. In the **Action** field, select the name of the Action you want to apply from the list of values.

    The system displays the Action type and text of the Action you selected.

4. In the **Action Text** field you can change the default Action text if you wish.

5. Click **OK**. TMS declassifies the term and applies the Action to the term.

Browsing the Audit Trail

The bottom block in the Approve VTAs window also contains a tab window that displays an Audit Trail for the VTA selected in the **Verbatim Term Assignments** block. Records in the Audit Trail are presented from most current to oldest, and enable you to track who made each change to the VTA and when the changes occurred, or just browse the comments about the VTA.

Approving Action Assignments

This section includes:

- Understanding the Approve Action Assignments Window on page 10-15

- Approving a Nonapproved VTA on page 10-13

- Modifying the Action Assignment on page 10-18

For each dictionary/domain combination you can configure TMS to send omissions with actions applied directly to the external system or to pass through an approval stage. The approval stage allows an intermediate individual to decide if it is really necessary to take the extra time for the term to travel through the additional processing cycle. See "Creating Domains and Assigning Dictionaries to Domains" on page 6-37 for more information.

Users must have TMS_APPROVE_PRIV to approve Internal Action assignments. Users with TMS_CLASSIFY_PRIV can query in the Approve Action Assignments window, but cannot perform any operations there.

Understanding the Approve Action Assignments Window

The Approve Action Assignments window has two tabs above and two tabs below:

Distinct and All Action Assignments

In the tabs in the upper part of the Approve Action Assignments window, TMS displays only the current Action assignments. To see a history of the term's Action assignments, use the tabs in the lower part of the window.

There are two tabs in the upper part of the window:

- The **Distinct Action Assignments** tab shows Action assignments associated with unique verbatim terms.

  You can approve actions only in the Distinct Action Assignment window. TMS then applies the Action assignment to all occurrences of the same term.

- By contrast, the **All Action Assignments** tab shows all occurrences of the verbatim term that is selected in the **Distinct Action Assignments** tab, and provides external system info for each source term.

To see data, enter a query in the **Distinct Action Assignments** tab. Select a radio button to restrict the query as follows:

- **All** allows queries to retrieve both internal and Answerable Action assignments.

- **Approved** allows queries to retrieve only Answerable Action assignments that have been approved.

- **Not approved** allows queries to retrieve only Internal Action assignments, including those with and without base Answerable Actions. However, only those with base Answerable Actions can be approved.

> **Note:** Internal Actions defined with a base Answerable Action are used as an approval mechanism. When you query for nonapproved actions, you retrieve only Internal Actions (some of which may not have base Answerable Actions and therefore not require or support approval). When you approve one of these Internal Action assignments, TMS immediately assigns the base Answerable Action to the term in place of the Internal Action. Therefore, when you query for approved Action assignments, you retrieve only Answerable Actions.

> **Note:** TMS does not display unAnswerable Action assignments in this window.

Query results are also filtered by the settings in the Filter window. To check and change your filter settings, click the **Filter** button. See "Setting Filters and Creating Activity Lists" on page 10-24 for more information.

For each term, TMS displays the following information. Some information is displayed only on the Distinct tab or only on the All tab (as noted):

N (**Distinct** tab only.) If an N appears to the left of the Appr? field, the term has at least one note associated with it. Go to **Status/Notes** to see the term's notes.

Appr? If selected, the Action is approved to proceed to the external system.

Verbatim Term In the **Distinct Action Assignments** tab, this is the distinct verbatim term. In the **All Action Assignments** tab, this is a particular occurrence of the verbatim term.

Omission Status (**All** tab only.) The TMS status associated with the omission occurrence.

Action Text The text of the Action currently applied to the term.

Action Owner (**All** tab only.) The user name of the person who assigned the Action to the term.

Action The name of the Action currently applied to the term.

Action Type The type of the Action currently applied to the term. See "Action Types" on page 7-6.

Status (**Distinct** tab only.) The current status of the term to which the Action is assigned.

Action App Owner (**All** tab only.) The external system that currently owns the omission.

Update? (**All** tab only.) If checked, this omission record has been updated, and the next data exchange job with the external source data system (such as Oracle Clinical Batch Validation) will process the record.

Instance Name (**All** tab only.) The name of the database where the term was collected in the external system.

Source Term ID (**All** tab only.) Source_term_id, occurence_id, def_instance_id (database), def_integration_key (external system) together form the unique identifier of a source term in TMS.

Occurrence ID (**All** tab only.) The source_term_id, occurrence_id, def_instance_id (database), def_integration_key (external system) together form the unique identifier of a source term in TMS.

Entry Ts (**All** tab only.) The time and date when the source term entered TMS.

Source Term Alt Key (**All** tab only.) Alternative source record identifier; can be used from the external system to provide a different access patch into TMS.

External System (**All** tab only.) The external source data system where the term was collected.

Value 1 - 8 Fields (**All** tab only.) These fields display information from the external system associated with each particular occurrence of the verbatim term. For Oracle Clinical these values include Project, Study, Patient, Visit, and more. For other systems your company can specify which external system to display.

Distinct and All Action Assignment Audit Trail

There are two tabs at the bottom of the window:

■ **Distinct Action Assignment Audit Trail**. This tab displays information about the actions that have been applied to the term in the past. (The current Action is displayed in the upper portion of the window. These tabs display only the previous actions, if any.)

■ **All Action Assignment Audit Trail**. This tab displays the actions applied to the particular occurrence of a verbatim term selected in the **All Action Assignments** window. It also displays reply comments entered in the external system, if any. If an Action was deleted without being replaced with another Action, a blank row appears.

These tabs display the following information:

Omission Status   (**All** tab only.) The omission status of the verbatim term occurrence at the time the Action on that row was applied.

Action Text   The text of the Action.

Action Owner   (**All** tab only.) The user name of the person who performed the operation represented by that row of Action audit trail. For records owned by TMS record, it is the TMS user who assigns the Action. For records owned by an external system, it is the external system user who enters a reply.

Action   The name of the Action.

Action Type   The Action type: Answerable, Unanswerable, or Internal. See "Action Types" on page 7-6 for more information.

Action App Owner   (**All** tab only.) The application currently responsible for the Action; either TMS or the external system.

Entry TS   The date and time the Action was assigned.

Created By   (**Distinct** tab only.) The user name of the person who assigned the Action.

Buttons
The window includes the following buttons along the bottom:

Filter   Click the Filter button to change the filter settings to refine query results from this window. You can save filter settings as an activity list for future use. See "Setting Filters and Creating Activity Lists" on page 10-24 for more information.

Reject Assignment   If you do not agree with an Action assignment and do not want to substitute a different one, select term and click **Reject Assignment** or select **Reject Assignment** from the **Options** menu. TMS immediately removes the Action assignment and saves. There is no Undo.

Modify Assignment   To change the Action text before you approve the assignment, or to assign an entirely different Action, click **Modify Assignment** or select **Modify Assignment** from the **Options** menu. See "Modifying the Action Assignment" on page 10-18 for more information.

Status/Notes   (Click the **Status/Notes** button to see the history of all statuses the term has had in TMS and all notes ever assigned to the term. You can also add a new note to the term. See "Using the Status/Notes Pop-up Window" on page 10-21 for more

information.

Approving a Nonapproved Internal Action Assignment

To change a nonapproved Action assignment to Approved status:

1. From the **Omission Management** menu, select **Approve Action Assignments** to reach the Approve Action Assignments window.

    If the Filter window opens, enter Filter criteria; see "Setting Filters" on page 10-25.

2. Query for the Action assignment in either of the upper tabs. If you use the Distinct Verbatim Term Action Assignments window, select the **Not Approved** option before entering your query criteria.

    The window returns all of the verbatim terms with Action assignments that match your query and filter settings.

3. Click the **Appr?** box for the Action assignment you want to approve.

4. Save.

Unapproving an Approved Action Assignment   Changing an approved Action assignment to nonapproved status works the same way; query for the appropriate term and deselect the **Appr?** box to set it to nonapproved. TMS then applies the Internal Action associated with the Answerable Action to occurrences of the term currently owned by TMS, and future occurrences of the term. Any occurrences of the term that are currently owned by the external system are not affected by the unapproval.

Modifying the Action Assignment

You can change an Action's text for a particular term before you approve the Action assignment or even approve the assignment of a completely different Action, including an Action of a different type.

Modify Action Text   To modify only the Action text, do the following:

1. Select the term whose Action's text you want to change.

2. Click the **Modify Assignment** button or select **Modify Assignment** from the **Options** menu.

3. Enter the new text in the **Action Text** field.

4. Click **OK**.

Assign a Different Action   Only one Action can be assigned to a particular term at a time. If you assign a different Action to a term, you effectively remove the previous Action assignment.

To assign a different Action to the term, do the following:

1. Select the term whose Action's text you want to change.

2. Click the **Modify Assignment** button or select **Modify Assignment** from the **Options** menu.

3. In the **Action** field, select a different Action from the drop-down list. The list displays all active Answerable Actions available. TMS enters the **Action Type** and **Text** for the Action you selected.

4. Modify the text in the **Text** field if necessary.

5. Click **OK**.

Term Statuses

TMS uses the following statuses:

VTO   Verbatim Term Omission. Terms that cannot be by the Autoclassification process are called **omissions**. Substatus: **CRE (Created)**.

- **CRE (Created)**. A verbatim term is not through Autoclassification and enters the pool of omissions (no prior status).

- **RCRE (Recreated)**. A VTA is declassified, returning to the omissions pool (from status VTA_D).

- **UNALLOC (Unallocated)**. An omission is deallocated, returning to the unallocated omissions pool (from status VTO_ALLOC).

VTO_ALLOC   Allocated Verbatim Term Omission. When an omission is allocated to a user as a task (for classification), TMS gives it a status of VTO_ALLOC. Substatuses:

- **CRE (Created)**. The omission was allocated to a coder directly without passing through the VTO state (no prior status).

- **ALLOC (Allocated)**. The omission was allocated to a coder through pool allocation (from status VTO or, if this is a re-allocation, from status VTO_ALLOC) or through manual allocation.

- **RCRE (Recreated)**. Either an omission's Action assignment has been deleted or a VTA has been de; the term re-enters TMS as an omission and is allocated.

VTA_UA   Unapproved Verbatim Term Assignment. The system gives this status when a user manually creates an unapproved VTA for a term and when Autoclassification finds a direct match between a term and an unapproved VTA. Substatuses:

- **CRE (Created)**. The classification was performed by a coder or approver on a term (from status VTO, VTO_ALLOC, ACT_INT, ACT_ALLOC, ACT_EXT, or no prior status).

- **RCRE (Recreated)**. The verbatim was previously in status VTA_UA, then was not, and now is again; for example, when a verbatim term is declassified and then again to an unapproved VTA.

- **REC (Re)**. The verbatim term has been reclassified but is still an unapproved VTA.

- **UNALLOC (Deallocated)**. An allocated VTA is turned back to the unallocated pool (from status VTA_ALLOC).

VTA_ALLOC   Allocated Unapproved VTA. The system gives this status when an unapproved VTA is allocated to a user for approval.

- **ALLOC (Allocated)**. An unapproved VTA is allocated to an approver (through pool or manual allocation) for review (from status VTA_UA).

- **CRE (Created)**. An omission is to an unapproved VTA and directly allocated to an approver. The verbatim term may or may not be associated with an Action (from status VTO_ALLOC, ACT_INT, ACT_ALLOC, or ACT_EXT).

VTA_A   Approved Verbatim Term Assignment. The system gives this status to terms automatically to Approved VTAs, and to unapproved VTAs after they are approved,

and to terms that are in a dictionary/domain combination where approval of VTAs is not required. Substatuses:

- **CRE (Created)** (from status VTO, VTA_ALLOC, VTO_ALLOC, ACT_INT, ACT_ALLOC, r ACT_EXT, or no prior status).

- **RCRE (Recreated)**. The verbatim was previously in status VTA_A, then was not, and now is again; for example, when a verbatim term is declassified and then again.

- **REC (Re)**. The verbatim term has been reclassified.

VTA_D   Declassified Verbatim Term Assignment. The system gives this status to VTAs that are declassified. Substatuses:

- **DEC (Declassified)** The verbatim term is declassified (from status VTA_UA, VTA_ALLOC, or VTA_A).

- **DEL (Deleted)**. The verbatim term associated with a VTA is changed, so the VTA is deleted (from status VTA_UA, VTA_ALLOC, or VTA_A).

VTA_GLB   VT Globally Resolved. The system gives this status to verbatim terms when a domain VTA is promoted to a global VTA. Substatuses:

- **CRE (Created)**. The system gives this substatus to the to the domain VT that is now globally resolved.

- **DEC (Declassified)**. The system gives this substatus to the old domain VTA (from any VTA_ status). The Declassified substatus refers to the fact that the domain-specific VTA is declassified and supplanted by the global VTA.

- **RCRE (Recreated)**. The verbatim was previously in status VTA_GL, then was not, and now is again; for example, when a verbatim term is declassified from a global VTA and then again to a global VTA.

VTI   VTI (Verbatim Term Individual). The system gives this status to verbatim term omission is classified as a Verbatim Term Individual in a dictionary with nonunique terms in its classification level. Substatuses:

- **CRE (Created)**. The system gives this substatus to the to VTI when it is initially created.

- **RCRE (Recreated)**. The VTI was previously declassified and is now reclassified.

VTI_D   VTI Declassified. The system gives this status to a VTIs when it is declassified. Substatuses:

- **DEL (Deleted)**. The system gives this substatus to the to the VTI when it is deleted.

- **DEC (Declassified)**. The system gives this substatus to a VTI when it is declassified.

ACT_INT   Internal Action. TMS gives this status to omissions with Internal Actions assigned. The Action requires approval before TMS sends the term with the Action back to the source system. Substatuses:

- **CRE (Created)**. A user has assigned an Internal Action to the term (from status VTO, VTO_ALLOC, or no prior status).

- **RCRE (Recreated)**. The verbatim was previously in status ACT_INT, then was not, and now is again; for example, when a verbatim term has an Internal Action

assigned, then no longer assigned (due to its being approved and the base external Action being assigned, or due to the term's being) and then an Internal Action is again assigned to the term.

- **UNALLOC (Deallocated)**. The term has an Internal Action assigned and was formerly allocated to a user, but the user allocation has been rejected (from status ACT_ALLOC).

ACT_ALLOC   **Allocated Internal Action Assignment**. TMS gives this status to omissions with Internal Actions that are allocated as a task (for Action approval) to a user.

- **ALLOC (Allocated)**. The term has an Internal Action assigned and is allocated as a task to a user for approval from the unallocated pool (from status ACT_INT) or is re-allocated from one approver to another (from status ACT_ALLOC).

- **CRE (Created)**. A coder creates an Internal Action assignment and TMS allocates it directly to an approver (from status VTO, VTO_ALLOC, or no prior status).

ACT_EXT   **External Action Assignment**. TMS gives this status to omissions with external Actions that are ready to be sent, or have been sent, to the external source data system. Substatuses:

- **CRE (Created)** (from status VTA_ALLOC, VTO_ALLOC, or no prior status).

- **RCRE (Recreated)**. The verbatim was previously in status ACT_EXT, then was not, and now is again.

ACT_D   **Deleted Action Assignment**. TMS gives this status to omissions whose assigned internal or external Action is deleted. This is a rare case, only existing when an Action assignment is deleted and no underlying source data exists, such as when you have created an Action assignment directly in the Maintain Action Assignments form.

Substatus: **DEL (Deleted)** An Action assignment is deleted (from status ACT_INT, ACT_ALLOC, or ACT_EXT).

Using the Status/Notes Pop-up Window

You can see the status history of any term from all windows where you can change a term's status by clicking the Status/Notes button at the bottom of the window. You can also add a note at any time, including when you change the status; for example, to explain why you changed the status.

In the Browse VT History (under the Repository menu) you can view the Status/Notes window but you cannot add notes.

To view a term's status history from any of these windows, do the following:

1. Go to one of the following windows:

   - Approve Action Assignments (under Omission Management)

   - Approve VTAs (under Omission Management)

   - Classify VT Omissions (under Omission Management)

   - VT History (under Omission Management)

   - Reclassify Verbatim Terms (under VTA Maintenance)

   - Promote/Demote VTAs (under VTA Maintenance)

- Maintain Action Assignments (under VTA Maintenance)

- Task Allocation by Term (under Task Allocation)

2. Query for the term you want.

3. Select the term.

4. Click **Status/Notes** at the bottom of the window.

   The **Status/Notes** window opens. The term, with its domain and dictionary, are displayed at the top of the window. The window is divided into two sections: Status History and Note History.

You can add notes to the term; see "Adding a Note to a Term" on page 10-23 for more information.

Status History

The term's status history is displayed with the most recent status at the top and the first status at the bottom. For each status, the system displays the following information:

- **Status**. The status and substatus names, displayed as *status/substatus*; for additional information see "Term Statuses" on page 10-19.

- **Created By**. The user who manually changed the status or ran the process (such as Batch Validation) that changed the status.

- **Created**. The timestamp of the status creation or change of status, including the date and time.

Actions   TMS also displays the following information for Action assignments:

- **Action**. The name of the Action.

- **Action Type**. The Action type; either Internal, Answerable, or Unanswerable.

- **Act Ref Type**. If the Action is an Internal Action, and if it is based on an external Action, the external Action's type appears; either Answerable or Unanswerable.

- **Act Text**. The actual text of the Action.

> **Note:**   Within the time period of each Action status in a term's status history, the system displays only the last Action applied to the term, if any. You can see the complete Action history of a term in the HTML Browser; see "Viewing a Term's Action History" on page 14-24.

Note History

On the right side of the window the system displays notes that have been applied to the term. All the notes are related to the term's status history (for example, notes applied to a term in the Repository Maintenance form, which are of type Content or Relation, are not displayed here; see "Defining Informative Note Attributes" on page 7-22).

However, the notes do not necessarily have any relation to the status displayed next to them. Notes and statuses are both displayed in chronological order, but because you can create a note without changing the status, and can change the status without creating a note, status changes and notes that correspond logically may not be displayed next to each other.

TMS displays the following information for notes.

- **Label**. The label for the Informative Note Attribute on which the Informative Note is based.

- **Note**. The note text entered by a user.

- **Created By**. The system displays the user name of the person who added the label and/or note. If you add a label or note, the system enters your user name when you save your work.

- **Creation Time**. The system displays the date and time when the label and/or note was added. If you add a label or note, the system enters the timestamp when you save your work.

Adding a Note to a Term

To add a note to a term, do the following:

1. Query for the term and open the Status/Notes window from the appropriate window; see "Using the Status/Notes Pop-up Window" on page 10-21.

2. Click in the first empty Label field. If necessary, select Insert from the Record menu.

3. Click the ellipsis **(…)** to display the Informative Notes Attributes list of values.

4. Select the Informative Note Attribute on which you want to base your note.

> **Note:** TMS displays only attributes of type Workflow and data type Memo, and only those with "Applies To" defined as Term History; see "Defining Informative Note Attributes" on page 7-22.

5. Enter text in the Note field.

6. Save.

Using the VT History Window

In the VT History window you can see the status history of any term, whether it is currently an omission, a verbatim term with an Action, a VTA, or a VTI. This is the only window in TMS where you can see the status history of terms with any current status and create notes for any term.

To view the current status and substatus of any term as well as to view the complete status history of any term and add a label or note associated with a status change, do the following:

1. In the Omission Management menu, select VT History. The VT History window opens.

2. Query for the term whose history you want to see, or query for a particular status to see all terms at that status. See "Querying in Windows" on page 2-9. The system displays the following information:

   - **Term**. The term name.

   - **Domain Name**. The domain in which the term applies, or Global if it applies in all domains. If the term is valid in more than one domain, whether it is not is not also global, it is listed multiple times, once for each domain/dictionary combination.

- **Dictionary Name**. The dictionary to which the term is mapped. If a term is mapped to multiple dictionaries, it is listed multiple times, once for each domain/dictionary combination.

- **Status/Substatus**. The term's current status and substatus in the domain/dictionary combination on the same row; see "Term Statuses" on page 10-19.

- **Assigned**. If a task associated with the term—classifying the VTO, approving an Action assignment or unapproved VTA—is currently assigned to a user, the person's user name appears in this column.

- **N**. If the letter **N** is displayed next to a term name, there is at least one note associated with the term.

3. To view the complete history of a term, including all notes ever associated with the term, select the term and click **Status/Notes** at the bottom of the window. See "Using the Status/Notes Pop-up Window" on page 10-21.

    You can also add notes from there; see "Adding a Note to a Term" on page 10-23.

Creating and Using Activity Lists

This section includes:

- Using Activity Lists on page 10-24

- Setting Filters and Creating Activity Lists on page 10-24

- Maintaining Activity Lists on page 10-28

Using Activity Lists

You can use the Activity List window in the Omission Management menu as your starting point for work every day. You can set up your activity list to organize the tasks allocated to you (if your company is using task allocation) and keep track of the tasks you have not yet done. Each activity lists a single type of task—omissions to be classified, unapproved VTAs or Action assignments to be approved—in a particular domain/dictionary combination or in all domains and/or all dictionaries.

You can click the activity name to see and work on the individual tasks allocated to you in the appropriate window: Classify Omissions, Approve VTAs, or Approve Action Assignments. When you close the window you return to the Activity List window and you can proceed to a different activity.

To create an activity, enter criteria in the filter pop-up window in the Classify Omissions, Approve VTAs, or Approve Action Assignments and save the filter as an activity (see "Setting Filters and Creating Activity Lists" on page 10-24).

The Activity List window always displays a summary of the current results of the filter, including the task type, dictionary, domain, term status, and current total.

You can also create activity lists of other users' tasks, though you will only be able to see terms to which you have standard Data Access Group security access.

Setting Filters and Creating Activity Lists

In the **Classify VT Omissions**, **Approve Action Assignments**, and **Approve VTAs** windows you can limit the results returned by queries in the window by setting **Filter** criteria. You can save the filter criteria settings as an activity list and access it through the **Activity List** window. See:

- Setting Filters on page 10-25
- Saving Filter Settings as an Activity on page 10-27

Setting Filters

The values you set here limit the results retrieved by queries in the window. For example, if you specify a dictionary, TMS displays only terms in that dictionary.

Your default filter criteria settings are determined by your TMS user profile. Any changes you make to the filter settings continue in effect throughout your current session, even if you change windows, until you change the setting again. If your user profile does not set defaults, and you have not selected any during your TMS session, the Filter window opens automatically when you launch any of these windows.

To set filter criteria, do the following:

1. Choose **Options**, then **Filter**, or click the **Filter** button. The Filter pop-up window opens.

2. Select a dictionary and domain to work in:

   - **Domain**. Select a value from the drop-down list.

   - **Dictionary**. Select a value from the drop-down list.

3. To narrow the search, select values for one or more of the following fields:

   - **Assigned**. To see your own assignments, enter your user name or [LOGIN_ USER] (the default value). You can enter other people's user names or leave the field blank to see assignments to all users, but the results will be limited to the terms to which you have security access through a Data Access Group.

   - **TMS DB**. Select the database. There may be only one option.

   - **Status**. Select a status.

     – For omissions, the valid statuses are: VTO and Allocated VTO.

     – For VTAs, the valid statuses are: Unapproved VTA and Allocated VTA.

     – For actions, the valid statuses are: Internal Action and Allocated Action.

     Only certain statuses make sense in the context of a particular window. If you select a status that does not apply in context, the system ignores the status value altogether and returns results based on the other criteria.

   - **Substatus**. You can select a substatus only if you first select a status.

     > **Note:** The system does not limit the selection of substatuses to the substatuses appropriate to the status you selected. If you select a substatus that is invalid for the status, the query does not retrieve any records. For a list of statuses and their substatuses, see "Term Statuses" on page 10-19.

4. To specify external system criteria for the filter, click the **External System** tab and enter values for one or more of the fields:

   - **External System**. Select an external source data system from the list of values, or select All Systems.

     When you select an external system, TMS refreshes the screen to display the information stored for each source term in TMS; for example, for Oracle

Clinical TMS displays Study, Patient, Document Number, and five other values.

> **Note:** If you have created a VTA proactively using the Repository Maintenance or Repository Authoring window, but a corresponding source term has never entered TMS from an external system, TMS displays the VTA only if you select All Systems.

- **Source Database**. If the external system feeds data from multiple databases into TMS, you can specify a single one if you wish.

- **External System-Specific Information**. TMS displays up to eight (8) additional fields custom-defined for the external system you specified.

> **Note:** If you selected All Systems in the External System field, Values 1…8 contain all values for all systems. For example, if you are using TMS with both Oracle Clinical and AERS, and Value 3 is defined as Patient for Oracle Clinical and Case for AERS, the list of values in the Value 3 column includes both OC Patients and AERS Cases.

5. When you call the Filter window from Classify VT Omissions or Approve Action Assignments (but not Approve VTAs) there is a third filter tab called **Actions**. You can enter values as follows:

- **Action Workflow**. You can choose to retrieve only one of the following:

   - **Rerouted Actions only**. Terms with actions assigned that have returned to TMS from an external system.

   - **Actions not owned by TMS**. Terms with actions assigned currently owned by the external source data system; no Action can be taken within TMS on the terms.

   - **No Open Actions**. Terms without an Action currently assigned.

   - **Any Open Actions**. Terms with an Action currently assigned.

   - **Nonapprovable Actions**. Terms with an Action assigned that does not require approval.

- **Action App. Owner**. The name of the application that currently owns the Action; either TMS or an external system.

- **Action Owner**. The user name of the person to whom the Action is allocated, if any.

- **Omission Status**. The omission status of the term, if it is an omission.

6. To save these filter settings as an activity for future use, click **Save As Activity**; see "Saving Filter Settings as an Activity" on page 10-27.

7. If you did not save the settings as an activity, do one of the following:

- Click **OK** to implement these Filter settings for your current TMS session and close the Filter window.

- Click **Cancel** to revert settings to their values when you opened the Filter window. The Filter window closes.

- Click **Clear** to revert the settings to their values when you opened the Filter window. The Filter window remains open.

- Click **Restore** to revert Filter window values to the default settings for your TMS user profile. After clicking Restore, clicking Cancel cannot bring back the settings you had when you opened the filter window. The Filter window remains open.

Saving Filter Settings as an Activity

You can save filter settings as an activity list to organize your work in TMS. See "Creating and Using Activity Lists" on page 10-24 for information.

To save filter settings as an activity list accessible from the Activity Lists window, do the following:

1. From the Filter window, click **Save As Activity**. The Save Activity pop-up window opens.

2. Enter values as follows:

   - **Activity Text**. This text is displayed in the Activity Lists window and serves to identify the filter. You can enter text much longer than the field; up to 300 characters. Note that the Activity Lists window can display 29 characters in this field without scrolling.

     Tip: Since you can filter only on one task type at a time (omissions, VTAs, or Action assignments), use the same beginning for your activity text for each task type where the other filter criteria are the same.

     For example, if you are querying for your own tasks, begin with the same code for the same domain/dictionary combination for each task type. If you are querying for another user's tasks, begin the activity text with that person's user name.

   - **All Users?** If checked, this activity list is displayed in the Activity Lists window of all users, though each user can see only the terms to which he or she has security access. You can check this field only if you have the OPA_ ADMIN security role.

   - **All Dictionaries?** If checked, the filter retrieves terms from all dictionaries. If unchecked, the filter retrieves terms from the dictionary you specified in the Filter window.

   - **All Domains?** If checked, the filter retrieves terms from all domains. If unchecked, the filter retrieves terms from the domain you specified in the Filter window

   - **Group by Status?** If checked, the system displays terms grouped by status in the Activity List window, so that each line in the activity list shows a particular status and the quantity is the number that meet the criteria and are of that particular status.

   - **Group by Substatus?** Check **Group by Substatus** only if you have also checked **Group by Status**. If checked, the system displays terms grouped by substatus in the Activity List window, so that each line in the activity list shows a particular status/substatus combination and the quantity is the number that meet the criteria and are of that particular substatus.

> **Note:** If you pick a particular status/substatus in the filter, the Activity List window groups terms by status/substatus even if you do not officially group by status/substatus when saving the activity.

3. Click **OK**. TMS closes the Save Activity pop-up window and saves the filter query as an activity.

4. Do one of the following:

   - Change the settings and create a new activity.

   - Click **OK** to implement these Filter settings for your current TMS session and close the Filter window.

   - Click **Clear** to revert the settings to their values when you opened the Filter window. The Filter window remains open.

   - Click **Cancel** to revert settings to their values when you opened the Filter. The Filter window closes.

   - Click **Restore** to revert Filter window values to the default settings for your TMS user profile. The Filter window remains open.

Maintaining Activity Lists

Use the Maintain Activities window to clean out outdated filters that are taking up space in your Activity List. You can also rename activities here.

If you have the OPA_ADMIN database role you can see all activities, regardless of who created them. Otherwise you see only the activities that you created.

To delete an activity:

1. In the Omission Management menu, go to the Maintain Activities window.

2. Query for the activity you want to delete. See "Querying in Windows" on page 2-9. You can query on the type, whether or not the list is available to all users, and/or the activity text.

3. Select the activity you want to delete by clicking its name.

4. In the Record menu, select Delete.

5. Save.

Omission Management Reports

TMS includes the following reports under the Omission Management menu:

- VTA/VTI Creation Report on page 10-29
- Classification Statistics Report on page 10-29
- VT Domain Differences Report on page 10-30
- VT Auxiliary Classifications Report on page 10-30
- Unclassifiable Indication Omissions Report on page 10-31
- Indications Assignments Report on page 10-31

VTA/VTI Creation Report

You can view the verbatim term assignments (VTAs) and verbatim term individuals (VTIs) created for a particular dictionary within a specific time period and the author of the VTA.

The VTA/VTI Creation Report displays the following:

- The affected Dictionary
- The affected Domain
- The Verbatim Term for which the assignment is made
- The Classification Type: VTA or VTI
- The Dictionary Term to which the VT is assigned, and its ID
- The date and time of the VTA or VTI creation
- The author of the VTA or VTI

To generate a **VTA/VTI Creation Report**:

1. Select **Omission Management** in the navigator window
2. Select **VTA/VTI Creation Report**.
3. Fill in the general report specifications. See "Setting Parameters" on page 2-13.
4. Enter a value for **Job Specific** parameters:
   - **Active dictionaries**: Choose the dictionary on which to run the report
   - **Start Date:** Type in the Start Date of the time period for which to run the report. Use the MM-DD-YYYY format where MM is the first three letters of the month.
   - **End Date:** Type in the End Date of the time period for which to run the report. Use the MM-DD-YYYY format where MM is the first three letters of the month.
   - **Template**: Select the report template. If your company has created a custom template, it appears in the list of values. The Oracle Template is the default template.
5. Submit the job. See "Running a Job" on page 2-16.

Classification Statistics Report

This report summarizes the state of classification in the TMS repository. It displays the number of outstanding omissions, misspelled terms, VTIs, approved and unapproved VTAs and the total number of terms.

The Classification Statistics Report has two sections, which display the following:

- **Omissions** Displays the Domain on which the report is run, the number of Unique Verbatim Term Omissions (VTOs) and the total Omissions.
- **VTAs** Displays the Domain, the approval status of Accepted and Misspelled VTAs against various Search Objects as well as the Totals.
- **VTIs** Displays the Domain and the number of Accepted and Misspelled VTIs that were created by direct or manual match or loaded.

To generate a **Classification Statistics Report:**

1. Select **Omission Management** in the navigator window

2. Select **Classification Statistics Report**.

3. Fill in the general report specifications. See "Setting Parameters" on page 2-13.

4. Enter a value for **Job Specific** parameters:

   ■ **Active dictionaries**: Choose the dictionary on which to run the report

   ■ **Domain:** Choose the appropriate domain.

   ■ **Template**: Select the report template. If your company has created a custom template, it appears in the list of values. The Oracle Template is the default template.

5. Submit the job. See "Running a Job" on page 2-16.

VT Domain Differences Report

The VTA/VTI Domain Differences Report compares the VTAs and VTIs in two domain/dictionary combinations, returning those that are mapped to different terms in these domains.

To run the VT Domain Differences Report:

1. From the **Omission Management** menu, select **VT Domain Differences**.

2. Fill in the general report specifications. See "Setting Parameters" on page 2-13.

3. Fill in the job-specific specifications, all of which are required:

   ■ **First Domain/Dictionary**: Select the first domain/dictionary combination from the list of values.

   ■ **Second Domain/Dictionary**: Select the second domain/dictionary combination from the list of values.

   ■ **Template**: Select the report template to use.

4. Submit the job. See "Running a Job" on page 2-16.

VT Auxiliary Classifications Report

The VT Auxiliary Classifications report lists all VTI classifications in a particular domain/dictionary combination that have auxiliary information. For each VTI the system displays:

■ The auxiliary levels and values

■ The VTI ID

■ The dictionary term and its ID, code, and alt code

■ The instance and the user or system that created the VTI

To run the report:

1. From the **Omission Management** menu, select **VT Auxiliary Classifications**.

2. For the general parameters, see "Setting Parameters" on page 2-13.

3. Enter job-specific parameters:

   ■ **Active Dictionary**

   ■ **Domain**

   ■ **Template**

4. Submit the job. See "Running a Job" on page 2-16.

Unclassifiable Indication Omissions Report

The Unclassifiable Indication Omissions report shows, for a particular dictionary/domain combination, indication omissions that are not classifiable because the related source terms were not supplied with indications. Indication Omissions that are not linked to any source term are not included in the report.

For each source term the report displays:

- The source term ID, the external system where it originated, and any external system values associated with it

- The instance

- The verbatim term and term type (VTA or VTI)

- The lower dictionary term and its ID and code

To run the report:

1. From the **Omission Management** menu, select **Unclassifiable Indication Omissions**.

2. For the general parameters, see "Setting Parameters" on page 2-13.

3. Enter job-specific parameters:

- **Active Dictionary**

- **Domain**

- **Template**

4. Submit the job. See "Running a Job" on page 2-16.

Indications Assignments Report

The Indications Assignments report shows the current indication assignments for the local instance as a whole or for a particular dictionary and/or domain. For each assignment the report displays:

- The verbatim term, term type (VTA or VTI), and its ID

- VTA approval status

- Indication

- The lower dictionary term and its ID and code

- The higher dictionary term and its ID

- The user or system that created the assignment

- The Valid From date

To run the report:

1. From the **Omission Management** menu, select **Indications Assignments**.

2. For the general parameters, see "Setting Parameters" on page 2-13.

3. Enter job-specific parameters:

- **Active Dictionary** (optional)

- **Domain** (optional)

■ **Template**

4. Submit the job. See "Running a Job" on page 2-16.

11

VTA and VTI Maintenance

The Oracle Thesaurus Management System (TMS) **VTA Maintenance** menu, available if you have the tms_reclassify_priv role, provides access to windows that you can use to manipulate verbatim term assignments (VTAs) and verbatim term individuals (VTIs) in TMS. Users with tms_classify_priv can also access the Reclassify Indication Assignments window.

This section includes:

- [Promoting and Demoting VTAs](#) on page 11-1
- [Maintaining Action Assignments](#) on page 11-4
- [Reclassifying and Declassifying Verbatim Terms](#) on page 11-6
- [Reclassifying Indication Assignments](#) on page 11-10
- [Performing High-Level Reclassification](#) on page 11-11
- [Copying Domains](#) on page 11-13
- [VTA and VTI Maintenance Reports](#) on page 11-16

Promoting and Demoting VTAs

A Verbatim Term Assignment, or VTA, is the mapping between a verbatim term and the dictionary term to which it is classified. VTAs may be available globally or in one or more TMS domains (see "Global and Domain VTAs" on page 11-2) and they may or may not require manual approval. In the Promote/Demote VTAs window you can change a VTA's status from Global to Domain or from Domain to Global. You change a VTA's approval status in the Approve VTAs window under the **Omission Management** menu option. See "Approving and Unapproving VTAs" on page 10-11 for more information.

Default settings for Approved/Nonapproved and Global/Domain statuses for VTAs are set the in the reference codelist TMS_CONFIGURATION. You can override the default approval status for a particular dictionary/domain combination; from the **Definition** menu, select **Define Domains**, then **Dictionaries**. You can override the default for a particular VTA in the Classify VT Omissions window for both types of status. Changing a VTA's status in the Promote/Demote VTAs window overrides all previous settings.

About the Window

TMS displays VTAs in all dictionaries and all TMS domains together. TMS displays Global VTAs first, in alphabetical order, then Domain VTAs grouped by domain, with

domains listed alphabetically. VTAs are displayed in alphabetical order within their domain.

If a verbatim term is classified to different dictionary terms in two or more different TMS domains, it is listed twice or more, once for each VTA. In the text block, the verbatim term is listed with information about the VTA.

Global and Domain VTAs

A Domain VTA is available only in the TMS domain(s) in which it was created. A Global VTA is available across all TMS domains. Domain VTAs take precedence over Global VTAs; if a Domain VTA and a Global VTA exist for the same verbatim term within a domain, TMS uses the Domain VTA.

> **Note:** Even if the current domain allows classifying terms to nonapproved dictionary terms (for example, dictionary terms that are no longer valid in the current dictionary version), VTAs linked to nonapproved dictionary terms are not displayed in this window because they cannot be promoted. TMS does not allow global VTAs to be linked to nonapproved dictionary terms.

When You Promote a Domain VTA to Global Status

When you promote a Domain VTA to global status, TMS makes the newly promoted Global VTA available to all TMS domains. Thus, after you promote a Domain VTA that maps the source term "pain in right foot" to the dictionary term "foot pain NOS", all domains in that instance that do not already have Domain VTAs defined for the source term "pain in right foot" will classify that source term according to the Global VTA.

However, domains might already have Domain VTAs defined to classify this source term. Because these Domain VTAs override the classification specified in the Global VTA, there are two possible outcomes in base dictionaries:

- If the Domain VTA's mapping differs from that defined in the Global VTA, TMS retains the Domain VTA. To use the same example, if a Domain VTA specifies that "pain in right foot" should classify to the more specific dictionary term "right foot pain," TMS will keep that Domain VTA, and studies using that domain will continue to map that source term in the same way.

- On the other hand, if the Domain VTA is exactly the same as the newly promoted Global VTA, there isn't anything for the Domain VTA to override; both global and domain behavior are the same. In this case, TMS removes the Domain VTA, and in this domain, TMS classifies this source term according to the Global VTA. You can promote a Domain VTA to a Global VTA to make it available in all TMS domains.

The outcome is different for Domain VTAs in virtual dictionaries. If a Domain VTA exists in a virtual dictionary, and it maps to the same dictionary term as a newly promoted Global VTA, TMS retains the Domain VTA.

Demoting Global VTAs to Domain Status

When you demote a Global VTA to a Domain VTA, you must specify in which TMS domains you want TMS to create a Domain VTA. If you specify a TMS domain that already has a Domain VTA for that term, TMS will not override the Domain VTA already created. If you do not specify a domain that used the Global VTA, TMS creates an omission during the next Batch Validation. You will be able to create a Domain VTA for it during classification.

Approval Status for Global and Domain VTAs

When you demote a Global VTA to a Domain VTA, TMS treats it as a new VTA and assigns approval status to it according to setting of the **Approval Required?** box in the Define Domain Dictionaries window for the relevant domain.

When you promote a Domain VTA to a Global VTA, TMS assigns it the approval status indicated in the GLOVTAAPPRREQD reference codelist (see "GLOVTAAPPRREQD" on page 3-27 for more information.

Informative Notes

All Informative Notes associated with a VTA expire when you promote or demote the VTA. For more information on creating Informative Notes for VTAs, see "Using the Status/Notes Pop-up Window" on page 10-21.

Promoting a VTA

To promote a Domain VTA to a Global VTA:

1. From the **VTA Maintenance** menu, select **Promote/Demote VTAs** to reach the Promote/Demote VTAs window.

2. Query for the VTAs you want.

3. Highlight the VTA you want to promote.

4. Click **Promote to Global VTA**.

5. (Optional) You can type a comment about this VTA promotion in the **Comment** field, which is at the bottom of the window. You can then audit these changes in the Browse Repository Data window; click **Browse Dictionary** to reach this window directly from Promote/Demote VTAs.

6. Save.

Demoting a VTA

To demote a Global VTA to a Domain VTA:

1. From the **VTA Maintenance** menu, select **Promote/Demote VTAs** to reach the Promote/Demote VTAs window.

2. Query for the VTAs you want.

3. Highlight the VTA you want to demote.

4. Click **Demote to Domain VTA**.

5. Highlight the domain(s) in which you want the VTA to be available.

> **Note:** You can select more than one record using the standard Windows Shift+Click and Ctrl+Click behavior.

6. (Optional) You can type a comment about this VTA demotion in the **Comment** field, which is at the bottom of the window. You can then audit these changes in the Browse Repository Data window; click **Browse Dictionary** to reach this window directly from Promote/Demote VTAs.

7. Save.

Maintaining Action Assignments

This section includes:

- [Assigning an Action to a Term](#) on page 11-5
- [Rejecting an Action Assignment](#) on page 11-5
- [Modifying an Existing Action Assignment](#) on page 11-5

You can use actions to communicate with an external source data system or internally in TMS about a term by assigning an Action to the term; see "Defining and Using Actions" on page 7-6. When a verbatim term enters TMS that requires an Action, you can assign the Action in the Classify VT Omissions window in the Omission Management menu; see "Apply an Action to a Term" on page 10-7 for more information.

If Action assignment approval is required in a particular dictionary/domain combination, you can approve Action assignments in the Approve Action Assignments window. Upon approval, TMS automatically sends the base Answerable Action to the external system; see "Approving Action Assignments" on page 10-14 for more information.

Users with TMS_APPROVE_PRIV and TMS_RECLASSIFY privileges can create, modify, and delete Action assignments in this window. Users with TMS_CLASSIFY_ PRIV can query in this window. They can also update or delete Action assignments they created, within the time period set by the reference codelist value MTACTHOUR. If this value is set to zero (0) they cannot perform any operations in this window.

About the Maintain Action Assignments Window

In the Maintain Action Assignments window you can do the following:

- You can **proactively assign actions** to verbatim terms that have not yet entered the system. If you think a term is likely to be passed to TMS from the external system, you can create a verbatim term and assign an Action to it here. TMS then applies the Action to the term when it does occur in the current dictionary/domain combination. By assigning actions to terms early in a study, you can reduce the number of omissions that must be manually processed.

- **View all the Action assignments** in the system, one dictionary/domain combination at a time, including current and past Action assignments.

- **Reject Action assignments**.

Dictionary/Domain   You can view terms in a single dictionary/domain combination at a time. If you have not already selected a dictionary and domain to work in during this session, you must select one when you enter the window. If you have already selected them, the domain name appears in parentheses in the window title bar and the dictionary is displayed in the **Dictionary** drop-down list.

To change domains, select **Change Domains** from the **Options** menu.

Status/Notes   If a term has one or more notes associated with it, an **N** appears to the left of the term name. Click **Status/Notes** to see the notes.

Current/All   When you query for terms, select **Current** to see only actions that are currently applied to terms. Select **All** to see both current and past assignments.

Assigning an Action to a Term

TMS enables you to assign actions proactively to terms you think may be collected and sent to TMS during the course of a study. You can also query for existing Action assignments and update them, and view past Action assignments.

To create a verbatim term and assign an Action to it, do the following:

1. From the **VTA Maintenance** menu, select **Maintain Action Assignments**.

    If the Choose Domain window opens, select a dictionary and domain. If it doesn't open and you want to change the current dictionary/domain combination, select **Options** and then **Change Domain**. Then select a dictionary and domain.

2. In a new row, in the **Distinct Verbatim Terms** field, enter the name of the verbatim term to which you want to assign an Action.

3. In the **Action** list, choose the Action definition you want to assign to the term.

    TMS populates the **Action Type** and **Action Text** fields with the default values for the Action you selected. See "Action Types" on page 7-6 for more information.

4. In the **Action Text** field, you can change the default text for this Action to customize it for the current dictionary/domain combination.

5. Save. TMS saves this Action assignment for the current dictionary/domain combination.

Rejecting an Action Assignment

When you reject an Action assignment for a verbatim term, TMS removes the Action from all occurrences of that verbatim term in the current dictionary/domain combination that are owned by TMS. Furthermore, TMS does not apply the Action to subsequent occurrences of that verbatim term in the current dictionary/domain combination.

To delete an Action from a verbatim term:

1. From the **VTA Maintenance** menu, select **Maintain Actions**.

    If the Choose Domain window opens, select a dictionary and domain. If it doesn't open and you want to change the current dictionary/domain combination, select **Options** and then **Change Domain**. Then select a dictionary and domain.

2. Enter a query in any field; you can query for a term or for an Action. TMS returns all Action assignments that meet the query criteria.

3. Click in the row to select the Action assignment you want to delete.

4. Click **Reject Assignment**. TMS does not prompt you to confirm the deletion, and removes the Action assignment from this dictionary/domain combination.

Modifying an Existing Action Assignment

You can make the following changes to existing Action assignments:

- Change a term's assignment to an entirely different Action. Only one Action can apply to a term at a time in a particular dictionary/domain combination. If you change the Action, TMS applies the new Action to current and future occurrences of the term in the current dictionary/domain combination.

- You can change the Action text. In an external Action (answerable or unanswerable) this is the default text displayed in the external system. For Internal

Actions this text is displayed within TMS, in the Classify VT Omissions window and Approve Action Assignments window.

The external system omission status can be changed only in the Action definition; see "Defining and Using Actions" on page 7-6 for more information.

Reclassifying and Declassifying Verbatim Terms

This section includes:

- About the Reclassify Verbatim Terms Window on page 11-7
- Reclassifying a Verbatim Term on page 11-9
- Declassifying a Verbatim Term on page 11-9

The Reclassify Verbatim Terms window enables you to reclassify and declassify verbatim terms. The **reclassification** process changes the mapping of a verbatim term from one dictionary term to another. By contrast, **declassification** breaks a verbatim term's mapping without remapping it to a new dictionary term. Declassifying a verbatim term creates an omission.

You may want to reclassify or declassify a verbatim term that has already been classified if:

- You add a term to a dictionary that is a better match for a verbatim term than the one to which it is now mapped.
- You do not like a Candidate Term of type Nonapproved. (You must either accept Candidate Terms of type Nonapproved or reclassify the verbatim term; see "Candidate Terms and Search Objects" on page 10-2.)
- You make a mistake during manual classification.

You may map a verbatim term to a different dictionary term in different TMS domains. In the Reclassify Verbatim Terms window you see classified verbatim terms in one TMS domain at a time, and the changes you make affect only that TMS domain unless the **Global?** box is selected. See "Global and Domain VTAs" on page 11-2.

VTAs that you create during reclassification inherit the approval status of the VTA they replace. TMS maintains a full audit trail of reclassifications.

> **Note:** VTIs are always approved and are never global.

Reclassifications take effect immediately at the master instance and after symmetric replication, which runs frequently and automatically, at other instances.

> **Note:** If you are using replication, reclassification of Global VTAs is allowed only on the master instance.

When you reclassify a verbatim term, all version-specific Informative Notes for the old VTA or VTI expire, and TMS assigns them the same end timestamp as the old VTA or VTI. The new VTA or VTI will not have these Informative Notes associated with it.

To access the Reclassify Verbatim Terms window, you must either:

- Have tms_reclassify_priv privileges, in which case you can reclassify VTAs and VTIs created by any user.

- Have classified a verbatim term recently (within the number of hours set in the reference codelist TMS_CONFIGURATION), in which case you can see only VTAs and VTIs you have created, and can modify only those you created within the set time limit.

About the Reclassify Verbatim Terms Window

There are two tabs in the upper half of Reclassify Verbatim Terms that display verbatim term information:

- The **Distinct VTAs/VTIs** tab lists each distinct verbatim term that meets your query criteria. If the same term has been entered for ten patients, the term is displayed once.

- The **All VTAs/VTIs** tab lists every source term—verbatim term occurrence—that meet your criteria. If the same term has been entered for ten patients, all ten occurrences are displayed. Auxiliary information about each occurrence is also displayed.

Common Fields in the Distinct and All VTAs/VTIs Tabs

For each VTA or VTI they describe, both tabs display the following information.

Glb?   (Global?) If selected, the VTA is available in all domains. (VTIs are never global.)

Type   **VTA** means the term is classified with a verbatim term assignment to a unique dictionary term. **VTI** means the term is classified with a verbatim term individual to a nonunique dictionary term, using auxiliary information.

Subtype   VTAs and VTIs are either **Misspelled** or **Accepted** (correctly spelled).

Appr?   If selected, the VTA is approved for use in Autoclassification and manual classification. If deselected, TMS proposes the VTA as a Candidate Term but manual approval is required for each occurrence. (VTIs are always approved.)

Origin   How this term was introduced to this TMS dictionary.

Distinct Verbatim Term Assignments

In addition to the fields described in "Common Fields in the Distinct and All VTAs/VTIs Tabs" the Distinct VTA/VTI tab includes information about the dictionary term to which the VTA or VTI is linked.

Dictionary Term   By default, this field shows the dictionary term to which the verbatim term is assigned, if any. If you specify a Dictionary Term Display Procedure for this dictionary, this field may display a different term depending on how you wrote the procedure. See "Writing a Procedure to Change the Dictionary Display" on page 3-39 for specifics on the use of this functionality.

DT Appr?   If checked, the dictionary term is approved. If unchecked, the dictionary term is not approved. Depending on the Non Appr DT setting for the current dictionary/domain combination, you may or may not be able to classify a verbatim term to a nonapproved dictionary term. See "Assigning a Dictionary to a Domain" on page 6-38 for more information.

The remaining fields on the **Distinct Verbatim Term Assignments** block provide read-only information about the term displayed in the **Dictionary Term** field. For

details on these fields, see "Current Level Block Fields" on page 12-13.

All VTAs/VTIs

In addition to the fields described in "Common Fields in the Distinct and All VTAs/VTIs Tabs" the All VTAs/VTIs tab includes information about each occurrence of each verbatim term in the selected database:

Update?   If checked, data relating to this verbatim term has been updated, so the VT is flagged to be updated in the external system during the next data exchange. These changes—such as a change to the verbatim term, a change to the dictionary term to which it is assigned, a change to the VTA or VTI, or a change to the relation—may be made in the repository or during a dictionary upgrade, for example.

Source Term ID   Unique identifier for this verbatim term.

Occurrence ID   Unique internal identifier for this occurrence of this verbatim term.

External System   The external system that fed the term into TMS.

Value 1 - 8   These fields can contain information about the verbatim term occurrence that is stored in the external system. Your company determines what information, if any, is displayed in these fields. If text is displayed in blue you can select it and click the Drill Down icon to see additional information. For instructions on setting up this feature, see "Defining External System Information in TMS" on page 5-28.

Auxiliary Information   If the current dictionary has nonunique terms in its classification level, TMS displays the auxiliary information collected for each source term at the bottom of the All VTAs/VTIs tab.

Repository Terms

Use the **Repository Terms** block to search for the dictionary term or VTA to use as the new target of the changed VTA. After you execute a query, TMS displays terms that match the query in the classification level(s) and the verbatim term level. You can then choose one of the matches as a target for the reclassification.

For information about these fields, see "Current Level Block Fields" on page 12-13.

Data Filter Radio Buttons Before entering your query criteria in one of the fields, choose which of these data groups you want to include in your search:

- Current VTAs and dictionary terms, or all VTAs and dictionary terms (both current and deleted)

- Approved dictionary terms only, Nonapproved dictionary terms only, both of these subsets, or none. Choosing **None** restricts your search to VTAs.

- Approved VTAs only, Nonapproved VTAs only, both of these subsets, or none. Choosing **None** restricts your search to dictionary terms.

More Tools in addition, you can use these buttons:

- Highlight a term and click **Above** to see all of a term's relations in levels above. See "The Above Button" on page 10-10.

- Highlight a term and click **Below** to see all of a term's relations in levels below. See "The Below Button" on page 10-10.

- Select **Extended Search** to open the Extended Search window. See "Using the Extended Search Button" on page 10-9.

- Reset the filter. See "Setting Filters" on page 10-25.

- Filter on auxiliary information. See "DT Filter for Auxiliary Information" on page 10-9.

Reclassifying a Verbatim Term

1. From the **VTA Maintenance** menu, select **Reclassify Verbatim Terms**. TMS opens the Reclassify Verbatim Terms window, and sets default values for many of the fields according to your profile settings. See "Setting Filters and Creating Activity Lists" on page 10-24 to change these values for your session.

2. In the **Distinct VTAs/VTIs** tab, select one search criterion for VTAs:

   - **No** restricts your search to VTIs.

   - **Approved VTAs** only, **Nonapproved VTAs** only, or **All VTAs**.

     Note: This filter does not apply to VTIs, which are always created as Approved. If the current dictionary has nonunique terms in the classification level, all VTIs are displayed.

3. Query for the VTA or VTI you want to reclassify.

4. In the **Repository Terms** tab, select one search criterion for both VTAs and dictionary terms:

   - **No** retrieves none.

   - **Approved** only, **Nonapproved** only, or **All** VTAs or dictionary terms.

     Note: This filter does not apply to VTIs, which are not displayed here.

5. Select one:

   - **Current** retrieves current terms only.

   - **All** retrieves current and deleted terms.

6. Query for terms to which you might want to reclassify the verbatim term.

7. Highlight the term to which you want to reclassify the verbatim term and click **Reclassify**. The new classification is immediately available.

8. (Optional) To add an Informative Note for this reclassification, click **Status/Notes** or select **Status/Notes** from the **Options** menu. See "Using the Status/Notes Pop-up Window" on page 10-21.

Declassifying a Verbatim Term

You can declassify a verbatim term without reclassifying it to another dictionary term. This declassification creates an omission. The newly created omission will appear in the Classify VT Omissions window once you run Synchronization in the local TMS instance from which the verbatim term originated. In earlier releases, you had to run Autoclassification to see the omission created in TMS; now, only Synchronization is

necessary. However, to update discrepancies in Oracle Clinical, you still must run Batch Validation.

When you declassify a verbatim term, all Informative Notes associated with that VTA expire.

1. From the **VTA/VTI Maintenance** menu, select **Reclassify Verbatim Terms**. TMS opens the Reclassify Verbatim Terms window, and sets default values for many of the fields according to your profile settings. See "Setting Filters and Creating Activity Lists" on page 10-24 to change these values for your session.

2. In the **Distinct VTAs/VTIs** tab, select one search criterion for VTAs:

   ■ **No** restricts your search to VTIs.

   ■ **Approved VTAs** only, **Nonapproved VTAs** only, or **All VTAs**.

   > **Note:** This filter does not apply to VTIs, which are always created as Approved. If the current dictionary has nonunique terms in the classification level, all VTIs are displayed.

3. Query for the VTA or VTI you want to declassify.

4. To declassify a VTA, stay in the **Distinct VTAs/VTIs** tab. You can also declassify a VTI in this tab, but to see its auxiliary information, select it in the **Distinct VTAs/VTIs** tab and then open the **All VTAs/VTIs** tab.

5. Click **Declassify** or **Declassify and Action**, which is available from the Distinct tab for VTAs and the All tab for VTIs. If you click **Declassify and Action**, select an action and modify its text if necessary.

Reclassifying Indication Assignments

An indication omission can occur in a dictionary like WHO-Drug Format C when a single dictionary term is mapped to multiple terms in the next higher derivable level and the correct derivation depends on the indication for which the drug is being taken by a patient. You can classify indication omissions, creating indication assignments, under the Omission Management menu.

In the Reclassify Indication Assignments window you can:

■ Reclassify the indication assignment to a different higher level term.

■ Declassify the indication assignment, creating an indication omission.

The **Distinct Indication Assignments** tab displays each verbatim term with its indication, the level up to which terms could be successfully derived even before the indication omission was classified, the term that is derived in that level, and the higher level term to which the indication omission is assigned. The Distinct Indication Omissions tab displays each verbatim term/indication combination once.

1. Query for an indication omission assignment in the **Distinct Indication Assignments** tab and select it. You can see all occurrences of the same verbatim term/indication combination in the **All Indication Assignments** tab.

2. The **Higher Level Terms** tab automatically displays the higher-level terms that are related to the dictionary term selected in the Distinct Indication Assignments tab.

   ■ To reclassify the current lower level dictionary term to a different higher level term for the current verbatim term/indication combination, select the higher level term and click **Reclassify**.

- To declassify the indication assignment and create an indication omission, select the higher level term and click **Declassify**.

Performing High-Level Reclassification

You can use the High-Level Reclassification window to view the verbatim terms that can be reclassified to different derivation paths, and change the higher-level terms to be derived from these verbatim terms on a record-by-record basis.

You may want to use this functionality if the correct classification for a term depends on a characteristic of a patient. For example, an indication may primarily affect one body system in adults and in children. The same verbatim term for the indication needs to be linked to different body systems for adults and children.

This section includes:

Rules and Constraints

High-level reclassification is effective only where TMS is fully integrated with an external system; otherwise TMS cannot associate changes with the source data.

The relation you create here, which is called a High-level Classification (HLC), supersedes both the Primary Link and a Domain Primary Link that may have been defined for a term.

You can create HLCs only between terms in derivable levels with a Many Cardinality relation on the parent side.

The High-Level Reclassification Window

The window includes:

Lists

The values you select in these five lists filter different aspects of your data queries.

Dictionary This list includes all dictionaries associated with the current domain. Select the one in which you want to work.

DT Level - Assigned Level This list includes all level relations where:

- There is Many Cardinality defined on the parent side; that is, where a term in the child level can be related to more than one term in the parent level

- The relation between the levels is on the derivation path (both levels are defined as Derivable)

Classification Type Select **HLC** to display terms with HLC relations, **Non-HLC** to display terms that have Primary Link or Domain Primary Link relations, and **All Terms** to display both of these subsets.

External System If TMS is fully integrated with an external system such as Oracle Clinical, and your company has set TMS up to import external system information, the external system will appear in the list. Select any one system to see its custom

information. If you select **All Systems**, you see verbatim terms collected from all external systems, but you will not see any custom external system information about the verbatim terms.

Eligibility When set to **Terms Eligible for HLC**, queries will return terms that have multiple parents.

Upper Block: Classifications

Each row represents three relations along the derivation path for a particular occurrence of a verbatim term. An occurrence represents a single patient record, or source term--a single response collected at a particular visit by a particular patient. TMS displays all occurrences of the verbatim term you query for, with the verbatim term on the left and the dictionary levels you chose from the list in the middle and on the right.

The **Classif.** field shows the type of relation that exists between the two dictionary terms for this verbatim term occurrence. Since a Primary Link is required for Many Cardinality relationships along the derivation path, the classification type for all relations listed here that you have not changed is either Primary Link (PL) or Domain Primary Link (DPL), which overrides the PL in a particular domain. When you modify a classification here, by definition you create a High-level Classification (HLC) that overrides both the PL and the DPL for this occurrence only.

You can query only on the **Verbatim Term**, **Verbatim Term ID**, **Classif.**, **Source Term ID**, **Occurrence ID**, and **External System Info** fields.

If you select an external system from the list at the top of the window, TMS displays external system information for the verbatim term occurrence on the far right. If more information is available for these external values, they are displayed in blue. To view the information, select **Drill Down** from the **Navigate** menu, or click the **Drill-down** icon.

Lower Block: Higher-level Terms

The lower block displays all related terms in the higher derivable level (assigned level) that you chose from the list at the top of the window.

Performing High-Level Reclassification

To derive a different dictionary term for a particular occurrence of a verbatim term (a single patient record):

1. Select a value from the **External System** and **Dictionary** lists (see "Lists" on page 11-11).

2. From the **Dict. Level - Assigned Level** list, select the pair of levels where the level from which you want to derive a different value is the assigned level (parent level; displayed on the right).

3. In the upper block, query for the verbatim term for which you want to derive a different term.

4. Scroll down, using the Source Term ID, Occurrence ID or external system information to find the occurrence of the verbatim term for which you want to derive a different term.

5. Highlight the occurrence.

6. In the lower block, enter a query for related terms in the parent level. If you enter an unrestricted query, TMS displays all related terms in the parent level, and

shows which ones are defined as Global Primary Links and/or Domain Primary Links.

7. Depending on the type of classification that already exists for this occurrence, do one of the following:

Modify Classification

If the occurrence is currently linked to the Global or Domain Primary Link, and you want to override that with a link to a different term, do the following:

1. Highlight the term you want to derive for this occurrence.

2. Click **Modify Classification**.

3. Save. TMS creates an HLC for that verbatim term occurrence.

Undo Classification

If the occurrence is currently linked to an alternative term via an HLC, and you want to switch to the predefined Primary Link, or, if there is one, the Domain Primary Link, do the following:

1. Click **Undo Classification**.

2. Save.

Notes:

Domain Primary Links supersede Global Primary Links. If a Domain Primary Link exists but you want to derive the Global Primary LinkGlobal Primary Link for a particular patient record, you must create an HLC for that verbatim term occurrence.

You can define and update global and Domain Primary Links in the Maintain Repository Data window. To define a Domain Primary Link, you must be in the domain in which you want a Domain Primary Link.

Copying Domains

Domain copying enables you to copy VTAs and Actions from one domain to another. The process is flexible: for example, you can copy every VTA into the target domain, or only those of a particular approval status or subtype. Coupled with the use of virtual dictionaries, domain copying can help you conduct studies against multiple versions of a dictionary. This section includes:

- Planning a Domain Copying Operation on page 11-14

- Copying Domain Information Using the TMS User Interface on page 11-14

- Copying Domain Information with the TMS API on page 11-15

- Handling Errors that Arise from the Domain Copying Process on page 11-15

- Reassigning Oracle Clinical Elements on page 11-15

- Using Domain Copying with Virtual Dictionaries on page 11-16

Planning a Domain Copying Operation

Some of the rules that govern domain copying operations depend upon your dictionary and domain choices. Consult the following sections before copying VTAs and Actions from one domain and dictionary to another.

Domains

You can copy information from multiple source domains into a single target domain, but not in a single transaction. The Copy Domains window restricts domain copying to exchange between a single source domain and a single target domain, so you must copy each source domain's information individually.

When different VTAs and Actions exist in the source and target domains, the existing target domain's information takes precedence; keep this hierarchy in mind as you plan the order in which you load the multiple source domains into the target.

Dictionaries

Actions and VTAs can be copied between domain/dictionary combinations regardless of dictionary structure.

Actions

You cannot copy Actions to or from the Global domain.

Copying Domain Information Using the TMS User Interface

To copy VTA and/or Action information from one domain to another using the Copy Domains window:

1. From the **VTA Maintenance** menu, select **Copy Domains**. The Copy Domains window opens.

2. Specify the source domain and dictionary in the window's **Source** block.

3. In the **Target** block, specify the target domain and dictionary.

4. Select **Copy VTAs?** to copy VTAs, and Copy VTIs? to copy VTIs from the source to the target domain.

   If you choose to copy VTIs, additional fields appear. For each auxiliary information level in the source domain dictionary, specify the equivalent level in the target domain dictionary.

   Note: Providing these mappings creates more accurate matching critieria for the copy logic.
   ---

5. Choose the **SubType** of the VTAs and VTIs you want to copy into the target domain. You can choose to copy all subtypes, only Accepted, or only Misspelled VTAs and VTIs.

6. From the **Appr?** list, choose the approval status of the VTAs you want to copy into the target domain. You can choose to copy only Approved VTAs, only Nonapproved VTAs, or all VTAs.

7. Select the **Copy Actions?** box to copy the Actions that have been linked to verbatim terms into the target domain.

8. Click **Copy**. TMS records any errors resulting from this domain copying in the TMS error log; see on page 11-15 for details.

Copying Domain Information with the TMS API

The package TMS_USER_MT_DOMAIN handles domain copying from external systems. See the *Oracle Thesaurus Management System Technical Reference Manual* for package details and restrictions.

Handling Errors that Arise from the Domain Copying Process

The domain copying process can generate errors when the source and target domain and dictionary are incompatible. For example, if the copying operation would overwrite a VTA or Action in the target domain/dictionary, or change its approval status or subtype, TMS will not change the VTA in the target domain/dictionary combination, and generates an error.

If a domain copying operation generates any errors, TMS saves all of the operation's errors in an error log. You can check if your domain copy generated errors by clicking **View Errors** after the domain copying operation completes. TMS opens the Maintain Domain Copying Error Logs window.

To view the errors generated from one domain copying transaction, select the error log's row from the upper half of the window. TMS displays the specific errors from that transaction in the lower half of the window, listing the details of each error in the **Error Msg** field.

The information in error messages is delineated by pound (#) symbols, as in the following example. Alternating items appear in bold here for clarity; the errors themselves appear as delineated strings in the **Error Msg** field.

Source Domain=BASE#Source Dict=BASE#**Target Domain=GLOBAL**#Target Dict if present=BASE#**SourceVT=TROUSERS**#SourceVTId=1000499#**Dict Term=SLACKS**#Dict Term Id=1000487#**ORA-20000: 245800-Record already exists.**

The first four items identify the source domain/dictionary combination and target domain/dictionary combination for this operation. The next four items describe which VTA is inconsistent in the source and target; the message provides the name and ID of the source verbatim term, and the name and ID of the dictionary term to which it classifies. The final item is the error message itself; in this example, the source VTA already exists in the target domain and dictionary.

Deleting Domain Copying Error Logs

You can also delete specific error logs by clicking in the log's row and selecting **Record**, then **Delete**, or delete all domain copying error logs by clicking the **Delete All** button at the bottom of the window.

> **Note:** The **Delete All** button deletes all of the domain copying error logs for all users in the installation.

Reassigning Oracle Clinical Elements

When you copy information from one TMS domain to another, you must reassign any elements from an external system that were associated with the original domain to the new domain.

In Oracle Clinical, for example, you must reassign projects and studies that are assigned to Domain A in the above example to Domain B. To switch domains for a study or project:

1. Navigate to Maintain Domain Elements; from Oracle Clinical's **Plan** menu, select **TMS Domains Elements**. Oracle Clinical prompts you to choose a study or project.

2. Choose a study or project.

3. Query for the domain you want to reassign, and click **TMS Domain Elements**.

4. Select a new domain, and click **Save**.

Using Domain Copying with Virtual Dictionaries

If you are using Oracle Clinical, the combined use of domain copying and virtual dictionaries allows you to continue coding some studies against a specific release of a dictionary while updating your base dictionary for use by other studies.

For example, and Study 12345 is coding against Domain A with one version of MedDRA 5.0 and your organization updates to the latest version of MedDRA, to continue coding Study 12345 against MedDRA 5.0:

1. In TMS, create a virtual dictionary with a cut-off date just prior to the date of the base dictionary update. This virtual dictionary will contain MedDRA 5.0 data and all the changes made to the dictionary until the cut-off date.

2. Create a new domain (Domain B) that contains the virtual dictionary for MedDRA 5.0.

3. In Oracle Clinical, go to Plan, then TMS Domains, and change to the virtual dictionary for the desired studies.

4. Copy domain information from the Domain A to Domain B. The next time you run Batch Validation for the selected study, the new dictionary will be used as the target/coding dictionary.

VTA and VTI Maintenance Reports

The following are VTA Maintenance reports:

- Nonapproved VTs Report on page 11-16
- Classification to a New Domain Report on page 11-17
- Verbatim Term Modifications Report on page 11-18
- X Areas with Outstanding Changes Report on page 11-19

Nonapproved VTs Report

The Non Approved VTs report displays nonapproved VTs in chronological order, with the most recent first. It also displays actions and notes associated with the VTs.

> **Note:** You can generate this report in MS Excel.

The **Non Approved VTs** report provides the following information about VTs:

- The dictionary and domain names
- The verbatim term
- Dictionary term to which the VT maps
- The dictionary term's level

- The VT's status

- The user ID of the person who created the VT

The report provides the following information about actions:

- The dictionary and domain names

- The verbatim term

- The action text

- The action code as it appears in TMS

- The action description

- The user ID of the person who created the action

To run the Non Approved VTs report:

1. Select the **Nonapproved VTs Report** from the **VTA Maintenance** menu.

2. Fill in the general report specifications. See "Setting Parameters" on page 2-13.

3. Enter a value for each job-specific parameter:

   - **Active Dictionaries**: Select the dictionary from the list of active dictionaries, for which you want to run this report.

   - **Domain**: You can optionally select a domain if you want the domain name to appear in the report along with the dictionary name.

     > **Note:** If you select a domain here that does not contain the dictionary you want to run the report for (selected in Active Dictionaries above), then your report will not show any data.

   - **Template**: Select the template for the report. If your company has created a custom template, it appears in the list of values. The Oracle Template is the default.

4. Submit the job. See "Running a Job" on page 2-16.

Classification to a New Domain Report

This report is useful when you are planning to move to a new domain and a different dictionary or a new rendition of the same dictionary; for example from WHO-ART to MedDRA. This report displays information about the amount of recoding that is required for this move. You can use the Copy Domain feature to make the coding similar in the two domains, if you so require. See "Copying Domains" on page 11-13.

The report displays:

- The omissions that will remain omissions

- The omissions that will be resolved (VTAs or VTIs)

- The VTAs and VTIs that will remain VTAs or VTIs

- The VTAs and VTIs that will be reclassified

- The VTAs and VTIs that will be declassified (omissions)

For each verbatim term, the report displays:

- The term type (VTA or VTI)

- The VT status and ID

- The original domain's dictionary term and its ID

- The new domain's dictionary term and its ID

To run the **Classification to a New Domain Report**:

1. Select the **Classification to a New Domain Report** from the **VTA Maintenance** menu.

2. Fill in the general report specifications. See

3. Enter a value for each job-specific parameter:

    - **Original Domain**: Select the domain from which you are moving your classifications.

    - **Original Dictionary**: Select the dictionary under the original domain from which the VTAs need to be reclassified.

    - **New Domain**: Select the domain to which you are moving the VTA classifications.

    - **New Dictionary**: Select the dictionary under the new domain to which the VTAs will be reclassified.

    - **Template**: Select the template for the report. If your company has created a custom template, it appears in the list of values. The Oracle Template is the default.

4. Submit the job. See

Verbatim Term Modifications Report

You can keep track of the coding changes that a Verbatim Term (VT) has undergone within a specified time period and data set with the VT Modifications Report. It specifies the type of change in the VT - classification, declassification, reclassification and applied actions. You can also view the author of the change and the reason for it.

You can confine your search to a verbatim term in a particular external system or include all external systems. You can define an External Value to narrow your search to a particular data set. For example, in Oracle Clinical, you can define an External Value as a study and assign an External Value Number to each study. TMS generates a report based on the specified study only.

The Verbatim Term Modifications Report displays the following:

- **Dictionary:** The selected dictionary or dictionaries on which the report is run.

- **Domain:** The active domain or domains on which the report is run.

- **Verbatim Term**: The Verbatim Term queried.

- **External Value**: This indicates the query criteria, if any.

- **VT Status**: The changes undergone by the VT (classification, declassification, reclassification and applied actions), the coder User ID, and the time of change.

- **Dictionary Term & Action Text:** This is the Dictionary Term for an approved VTA. For an Action, the Action text is displayed.

To run the Verbatim Term Modifications Report:

1. Select **VTA maintenance** in the navigator window.

2. Select **Jobs**.

3. Select **VT Modifications**.

4. Fill in the general report specifications. See "Setting Parameters" on page 2-13.

5. Enter a value for **Job Specific** parameters:

   ■ **Active Dictionaries**: Choose the dictionary on which to run the report. TMS searches all dictionaries by default.

   ■ **Domain**: Select the appropriate domain. TMS searches all domains by default.

   ■ **Coder User Name:** Select the User Name of the coder.

   ■ **External System:** Select the relevant External System.

   ■ **External Value number:** Select the relevant External Value Number from the field. If not coded in the External System, the report displays External Value Number as N/A.

   ■ **External Value**: This field is activated only if the External Value Number is already selected. The value corresponds with the External Value Number as pre-defined.

   ■ **Verbatim Term**: Type in the Verbatim Term to be queried.

   ■ **Start Date:** Type in the Start Date of the time period for which to run the report. Use the MM-DD-YYYY format where MM is the first three letters of the month. The default date is JAN-01-1900.

   ■ **End Date:** Type in the End Date of the time period for which to run the report. Use the MM-DD-YYYY format where MM is the first three letters of the month. The default date is AUG-15-3501.

   ■ **Template**: Select the report template. If your company has created a custom template, it appears in the list of values. The Oracle Template is the default template.

6. Submit the job. See "Running a Job" on page 2-16.

X Areas with Outstanding Changes Report

This report displays X Areas for which the data exchange job between the external system and TMS should be run. If the external system is Oracle Clinical, an X Area is a study, and the report displays the study names for which Batch Validation should be run.

If you are not running the data exchange job on a regular basis—for example, you have finished data collection in a study—you can run this job to detect X Areas where data cleaning in the external system or changes such as reclassifications in TMS have produced data changes that should be propagated to the other system.

If the external system is not Oracle Clinical, you can write a function to display a meaningful name for X Areas if appropriate; see "Defining the External System in TMS" on page 5-30 and "Defining Views and Functions in the External System" on page 5-33.

The report displays the X Area and other external system information for:

■ omissions that originated in all instances

■ source terms that originated locally only

This report is available on either the master or local TMS installation.

To run the report, do the following:

1. Fill in the general report specifications. See "Setting Parameters" on page 2-13.

2. Enter values for the following job-specific parameters:

   - **External System**. Select the external source data system whose X Areas you want to know about.

   - **Template**. Select a template; if your company has created a custom template, it appears in the drop-down list. The Oracle template is the default.

3. Submit the job. See "Running a Job" on page 2-16.

12

Repository Maintenance

You maintain the terms and relations in the Oracle Thesaurus Management System (TMS) Repository from windows under the TMS menus **Repository Maintenance** and **Translation Reports**. Windows under both menus are available to users with the tms_maintain_priv role.

If you have a distributed environment, you must perform all Repository Maintenance functions at the master instance.

Repository Maintenance includes:

- About Repository Data on page 12-1
- About the Maintain Repository Data Window on page 12-2
- Modifying Repository Data in the Maintain Repository Data Window on page 12-9
- Using the Repository Authoring Window on page 12-26
- Creating Informative Notes for Terms and Relations on page 12-24
- Using the Maintenance Wizard on page 12-26
- Viewing and Deleting Dictionary Loading Error Logs on page 12-29
- Using the Release Label Authoring Window on page 12-30
- Activating Data on page 12-33
- Repository Maintenance Reports on page 12-34
- Translation Reports to Identify Inconsistent Data on page 12-36

About Repository Data

As described in Chapter 6, the core of the TMS Repository consists of one or more predefined dictionaries. After dictionary loading, you can manipulate this dictionary data by adding, deleting, or changing dictionary terms and the relationships between them.

> **Note:** By default, you can only edit company or domain terms. In rare cases, you may want to grant users with the tms_maintain_priv role the ability to edit external terms as well. To open this functionality to these TMS users, navigate to the Installation Reference Codelists window, query for the codelist TMS_CONFIGURATION, and set the long value for MAINTAINEXT to Y.

TMS enables you to make these changes to the Repository using two windows under the Repository Maintenance menu in Maintain Repository Data and Repository Authoring. You can use either of these windows to manipulate term data, browse relations, or create strong relationships, but you can create named relations in Repository Authoring only. While strong relationships are possible only in Strong Dictionaries, you can create named relations for any type of dictionary. For example, named relations are commonly used to link terms in data-driven Weak Dictionaries, but you can also define named relations between two terms in MedDRA.

Because strong dictionary relationships link terms in the same dictionary, and are used for derivation and autoclassification purposes, you cannot define strong cross-dictionary links. As a result, all cross-dictionary links are named relations, and are created in the Repository Authoring window.

The first two sections of this chapter describe how to create terms and relations in the Maintain Repository Data and Repository Authoring windows. For information on using the Maintenance Wizard to perform more complex manipulations of terms and relations, see "Using the Maintenance Wizard" on page 12-26. To define Informative Notes for terms and relations, see "Creating Informative Notes for Terms and Relations" on page 12-24.

Data Currency and Data Source Filters

The top of this window enables you to focus your search in two areas. Use the **Data Currency** option in the upper left part to view all data or only current data, and you can choose to view preliminary data and/or historical data as well as current data by making a selection in the **Data Source** list to the upper right. See "Modifying Repository Data in the Maintain Repository Data Window" on page 12-9 for more information.

Controlling the Default Approval Status of New Terms

When you create new terms in the Maintain Repository Data and Repository Authoring windows, you can designate them as Approved or Nonapproved by selecting or clearing the **Appr?** (Approved?) box for that record. The TMS_ CONFIGURATION reference codelist setting DTAPPRFLAG controls the default setting of the approved box in each window: if DTAPPRFLAG is set to Y, the default setting for **Appr?** is checked; if DTAPPRFLAG is set to N, the default setting is unchecked.

For instructions about changing the DTAPPRFLAG setting, see "Other Installation-wide Codelists" on page 3-30. You must have the tms_define_priv role to change reference codelist settings.

For more instructions about creating new terms, see:

- "Adding and Deleting Relationships Between Terms" on page 12-10 for adding new terms using the Maintain Repository Data window.

- "Creating New Terms" on page 12-20 for adding new terms using the Repository Authoring window.

About the Maintain Repository Data Window

The Maintain Repository Data window has a dictionary hierarchy tree structure on the left. See "Using the Tree Structure" on page 2-4 for information on how to interpret it. When you click on a dictionary level you see the following:

- [Term and Relation Icons](#) on page 12-3
- [Current Level and Term](#) on page 12-3
- [Display of Relationships in Strong Dictionaries](#) on page 12-3
- [Display of Named Relations](#) on page 12-5
- [Using an Extended Search](#) on page 12-6

Term and Relation Icons

In the leftmost column on the right side of the screen, TMS displays *T* and line (_) icons that represent the state of each term and its relation to the current term, respectively. For the current term, which is highlighted in the middle block, TMS displays only the term icon.

TMS indicates that a term has been deleted by displaying a red *X* in place of the *T* icon to the left of the term. TMS indicates that a relation has been deleted by displaying a red *X* in place of the line icon to the left of the term. Visible black *T* and line icons indicate current terms and relations, respectively.

If one of these icons is not displayed, and there is no red *X* in its place, the term or relation is in a predict table awaiting Activation.

Current Level and Term

The **current level** is the level displayed in the **Current Level** block at the middle of the screen at any given time. You can select the current level by clicking on it in the dictionary hierarchy tree structure, at which point TMS displays the short name of the current level in the Term column in this block. To view data, you must then execute a query in the middle block.

The **current term** is the term displayed on a blue background in the **Current Level** block. After a query, TMS makes the first record the current term by default. You can change the current term by selecting another one with your cursor, or by querying for it. TMS displays data on screen as it relates to the current level and term—that is, the upper block displays terms in the level above the current level and related to the current term, and the lower block displays terms in the levels below the current level that are related to the current term.

To move related terms of the current term into the **Current Level** block, click on the level you want in the tree structure while the current term is displayed on the right, or double-click on the related term.

To see unrelated terms in any level, click on the level you want in the tree structure and then execute a query in the **Current Level** block.

The **Current Level** block in the middle of the window is the only one in which you can change the sort order (see ["Changing the Sort Order"](#) on page 13-9).

Display of Relationships in Strong Dictionaries

In strong dictionaries, the Maintain Repository Data window displays the parent and child relationships for the current term. The parent terms are in the **Level Above** block: if the current term is related to a reference term in the dictionary level above the current level, this reference term will appear in the **Level Above** block. If two (or more) levels are related to the current level just above it in the hierarchy, the current level is displayed twice (or more) in the tree structure. The level displayed in the **Level**

Above block depends on which display of the current level you highlight in the tree structure.

> **Note:** If you query in this block, TMS retrieves only terms that meet the query criteria AND are related to the current term. To create a new relation for the current term, either type the name of the term to which you want to create a relation or press F9 and select a term from the Parent Terms window. You must save your work.

Similarly, each row in the **Level Below** block represents a child term and its relation to the current term. The same rules apply as in the **Level Above** block: each row represents a relationship from the current term to a child term one level below it in the hierarchy.

Figure 12–1 shows parent and child relationships in the Maintain Repository Data window. In this example, named relations have been suppressed. You can either display or hide (suppress) the display of named relations in the Maintain Repository Data window by changing the setting SUPPRESNAMDRELS in the installation reference codelist TMS_CONFIGURATION. A description of this setting is on page 3-26.

Figure 12–1   Strong Relationships in the Maintain Repository Data Window

In the example in Figure 12–1, the current term "Muscle and tendon injuries" has a parent term of "Injuries NEC" and several child terms including "Tendon rupture" and "Repetitive strain injury."

Display of Named Relations

Because the terms in weak dictionary folders do not necessarily belong to a hierarchy of dictionary levels, the Maintain Repository Data window cannot organize the relationships into parent and child terms. Instead, this window arranges the terms by their use of Indicator and Reciprocal Indicator halves of named relations.

To illustrate the presentation of these relations with an example, assume that a TMS installation contains the named relations in the following table.

Table 12–1   Named Relations Defined for this Example

Indicator	Reciprocal Indicator
Broader Term Partitive	Narrower Term Partitive
Is a	(none)
Abbreviation	(none)

In Figure 12–2, the term "California" is currently selected from a weak dictionary folder of geographic place names.

Figure 12–2   Named Relations in the Maintain Repository Data Window

The window organizes relations into two groups:

■ In the upper block, TMS displays relations in which the current term is the reference term in the relation. Relations appearing in the upper block always use the Indicator half of a named relation, so you can read them from the upper block downward to the current term. In the geography dictionary example, the relations read "CA – Abbreviation – California" and "United States – Broader Term – California."

- In the lower block, TMS displays relations in which the current term is the first term in the relation. The direction in which you read these relations depends on the type of named relation. One-directional relations such as "California – is a – state" use the Indicator relationship, so you should read these relations from the current block to the lower block. Bi-directional relations that appear in the lower block such as "Los Angeles – Narrower Term – California" use the Reciprocal Indicator relationship, so read these relations from the lower block to the current block.

You can also use the Maintain Repository Data window to view the named relations you define for strong dictionaries. For both parent and child terms, the Maintain Repository Data window shows whether the given relation is linked via a strong or a named relation. If STRONG appears in the **Relation** field, that row represents a strong relationship; if either an Indicator or Reciprocal Indicator relationship appears, the given terms are linked by a named relation.

Using an Extended Search

Extended Searches enable you to search all domains, dictionaries and levels for a term. You can focus your Extended Searches in the following ways:

- Choose a **Search Type**. You can use a Cross Search (which searches all domains, dictionaries, and levels), or choose a user-defined search object. See "Defining Search Objects" on page 7-10.

- Choose a **Data Currency** option. You can restrict searches to current data only, or all data (current and expired data).

- Choose a **Data Source**. You can restrict Extended Searches to pre-dictionary data only, dictionary data only, or the combination of both.

- Enter search criteria about the term. The Extended Search window includes fields for every detail about a term.

To perform an Extended Search, from the Maintain Repository Data window:

1. Click **Extended Search**, or choose **Options**, then **Extended Search**. The Extended Search window opens.

   If you are working on the Verbatim Term Level in the Maintain Repository Data window and you choose a **Search Type** other than **Open Query** (see "Defining Search Objects" on page 7-10), TMS populates the **Verbatim Term** field with the current term from the Maintain Repository Data window and uses it as a filter in the query. You can modify it, but you cannot effectively delete it. The search algorithm must have a verbatim term parameter and it will use the last one entered.

   If you choose a **Search Type** of **Open Search**, TMS does not use the value entered in the **Verbatim Term** field, and the Extended Search window opens in Query mode.

2. If desired, enter your search criteria and execute the query. TMS returns any matches.

3. Choose the term you want. If you click **OK**, TMS populates the **Current Level** block of the Maintain Repository Data window with the results of your Extended Search. If you click **Cancel**, TMS displays the same data in the Maintain Repository Data window as before you executed the search.

Navigating to Data in the Maintain Repository Data Window

Take the following steps to open the Maintain Repository Data window and navigate to the data you want to modify:

STEP 1: Open the Window and Select Domain and Dictionary

1. From the **Repository Maintenance** menu, choose **Maintain Repository Data**.

   If you have selected a domain during your TMS session, TMS opens the Maintain Repository Data window and displays that domain in the window's title bar. If you have not selected a domain during this session, the Choose Domain window opens. You must choose a domain before you can modify any Repository data.

2. If the Choose Domain window opens, click the name of the domain in which you want to make your changes and the dictionary containing the data you want to modify, and click **OK**.

   > **Note:** If you do not want to use any of the displayed domains to make your changes, you can click **Cancel** to close this window, then select **Definition**, and then **Define Domains**, to create a new domain. See "Creating Domains and Assigning Dictionaries to Domains" on page 6-37 for full instructions.

3. TMS displays the Maintain Repository Data window, with the current TMS domain in the window's title bar. You can change TMS domains at any time by selecting **Options**, then **Change Domain**.

STEP 2: Choose or Create an Activation Group

TMS activates data in groups. Each term and relation you modify must be part of an Activation Group. TMS activates all data in the group in the same batch job. Therefore:

- Put only those terms and relations you want to have activated at the same time into the group.

- Put all terms with links to each other in the same group, or TMS will not be able to validate their relations.

Activation Groups are associated with one or more dictionaries. You can view and modify terms belonging only to the dictionaries associated with the Activation Group you choose. The dictionaries associated with each Activation Group are listed below it in the tree structure. You can associate dictionaries with more than one Activation Group.

Terms and relations awaiting Activation are stored in the predict (pre-dictionary) tables. When TMS successfully activates data, it moves them to the production tables. Terms and relations that TMS cannot activate because they violate hierarchy definitions remain in the predict tables with an error message.

Activation Groups continue to exist after Activation, even if all the terms and relations they contain are successfully activated. Data that fails Activation remains in the Activation Group.

Choosing an Activation Group In the tree structure, current Activation Groups are listed with a quill pen icon. To see information about an Activation Group, click on it in the tree structure in the Maintain Repository Data window. The Activation Group window appears.

The dictionaries that are associated with the Activation Group are listed in the box at the bottom of the window. The **In Domain?** box shows whether or not each dictionary is associated with the current TMS domain. If it is not, even though it is associated with the current Activation Group, the dictionary does not appear in the tree structure below the Activation Group and you cannot view or modify it without changing TMS domains or associating it with the current TMS domain.

Creating a New Activation Group　To create a new Activation Group in this database:

1. Highlight the words **Activation Group** in the tree structure.

2. Select **Insert Record**. TMS populates several of the fields on the right with default values.

3. Enter the new group's name, description, the short name, and name of the dictionary or dictionaries you want to associate with the group. You can find current dictionary information by selecting **Define Dictionaries** under the **Definition** menu.

4. **Save**. TMS enters a value in the **In Domain?** box for each dictionary and populates the tree structure below the Activation Group with all the dictionaries that are associated with it.

Deleting an Activation Group　To delete an Activation Group from this database:

1. Click the Activation Group's name in the tree structure.

2. Delete all dictionaries from the Activation Group by clicking in the rows in the dictionaries within the **Activation Group** block and deleting every row.

3. Click in the **Activation Group** field on the right side of the Maintain Repository Data window and delete the record.

TMS removes the Activation Group from the installation.

STEP 3: Choose a Data Currency Option

You can view deleted terms and relations by choosing **All** from the **Data Currency** option at the top of the window.

You can differentiate deleted data from current and historical data by checking the line and *T* icons in the left-hand column. If the line icon has been replaced by a red *X*, the relation has been deleted. If the *T* icon has been replaced by a red *X*, the term has been deleted. If a term has been deleted, all of its relations have been deleted as well.

> **Note:**　Selecting the **Current Data** option does not prevent you from seeing preliminary data (selecting **Pre-Dictionary Data** or **All Data** from the **Data Source** list).

STEP 4: Choose a Data Source

The Maintain Repository Data window enables you to view **pre-dictionary data** (terms awaiting Activation), including:

■ Terms and relations that are being added, modified, or deleted but have not yet been activated.

■ Terms and relations that could not be activated because they violated a level relation definition or other user-defined rule, with their error messages.

You can choose to display Active dictionary data only, pre-dictionary data only, or all data, by selecting a value from the **Data Source** list at the top of the screen.

If you choose to view all data, you can differentiate between dictionary and pre-dictionary terms by checking the column to the left of the term. Pre-dictionary terms do not have any icons in that column. Pre-dictionary terms that have failed Activation also have a value in the **Error Message** field.

> **Note:** Terms with a **DML** value of **Update** or **Delete** exist in the production and predict tables simultaneously until they undergo Activation. TMS displays these terms twice; the term displayed with a value in the **DML** field exists in the predict tables as preliminary data, and the term without a **DML** value exists in the production tables as active data.
>
> Terms with a **DML** value of **Insert** exist in the predict tables only.

STEP 5: Choose a Level and Term

TMS does not display any data in the window until you complete Steps 1 and 2 in this section.

1. In the tree structure on the left, highlight the level in which you want to add, modify or delete a term.

2. Enter a query in the middle section of the window on the right (the **Current Level** block). See "Querying in Windows" on page 2-9 for information on how to enter a query.

3. Execute the query. If the current level is a group level, terms from all its sublevels are displayed, with the sublevel identified in the Level column.

4. Highlight the term you want to work with in the **Current Level** block. This becomes the **current term**. TMS arranges the data on the screen as it relates to the current term; related terms in the level above the current level are listed in the block above, and related terms in the level below are listed in the block below.

Modifying Repository Data in the Maintain Repository Data Window

In the Maintain Repository Data window, you can perform two basic types of dictionary data maintenance:

■ Maintaining Terms, Relations, and VTAs on page 12-9

■ Resolving High-Level Omissions on page 12-17

Maintaining Terms, Relations, and VTAs

You can add company and domain terms to an external dictionary within its hierarchical structure (**company term**s and **company relation**s are available to all TMS domains; **domain term**s and **domain relation**s are available only within the TMS domain in which you define them). You can define strong relations between VTAs, company and domain terms, and external terms (terms supplied in an external dictionary such as MedDRA or WHO-Drug).

You can modify or delete company and domain dictionary terms, and their relations. See "Deleting Terms from a Dictionary" on page 12-11 and "Modifying a Term" on page 12-12.

> **Note:** TMS only allows you to create an external relation between the terms in a strong dictionary when the level relation is defined as mandatory.

Verbatim Term Assignments (VTAs) are represented by their verbatim term and Verbatim Term Individuals (VTIs) are represented by their verbatim term and auxiliary information. VTAs may be available either globally or only in a TMS domain (see "Global and Domain VTAs" on page 11-2), and inherit the global/domain status of their verbatim term. VTIs are available only in a TMS domain (see "Nonunique Coding Level Dictionaries" on page 6-8).

When you save your changes, TMS creates a new record in the predict tables for the term or relation to be inserted, updated, or deleted. When you requery you can see both the current record from the production tables and the new record in the predict tables. When you activate data, records that pass Activation move from the predict to the production tables, and the old production records are no longer current. Terms and relations that do not pass Activation remain in the predict tables with an error message.

This section includes general instructions for tasks, followed by detailed information on the fields in each block of the window, as follows:

- Adding and Deleting Relationships Between Terms on page 12-10
- Deleting Terms from a Dictionary on page 12-11
- Adding a Term to a Dictionary Level on page 12-11
- Modifying a Term on page 12-12
- Define Strong Relations to Terms in Other Levels on page 12-12
- Current Level Block Fields on page 12-13
- Upper Block Fields on page 12-15

Adding and Deleting Relationships Between Terms

To add a relationship between terms:

1. Highlight the lower level term in the **Current Level** block.

2. In the upper block, either enter the upper level term or choose it from a list of values.

3. Save.

> **Note:** You can also modify information about a relation, such as the **Relation Status** or **Relation Comment**, by editing these fields in the upper block.
>
> You cannot directly change the type from VTA to VTI or vice versa. If term uniqueness changes and such a change is legitimate, you must delete the existing relation and create a new one.

To delete a relationship between terms:

1. Highlight the lower level term in the **Current Level** block.

2. Highlight the higher level term in the upper block.

3. Either select **Record** and then **Delete**, or press Shift+F6.

4. Save.

Deleting relations from established dictionaries can be more complex because dictionary rules require mandatory relations between levels.

Deleting Terms from a Dictionary

When you delete a term from a dictionary, you also delete all relations from that term to other dictionary terms. Before activating data after deleting a term, you may want to examine pre-dictionary DML statements to reassign the relations before they are deleted.

Further, if you delete a term from a dictionary's classification level (or group level), TMS automatically marks all associated VTAs and VTIs for deletion as well. This deletion applies both to automatically and manually classified VTAs and VTIs. The verbatim terms that had mapped to this classification level term will become omissions upon the next Synchronization.

To delete a term and the relations to it from a dictionary, highlight the term and either select **Record** and then **Delete**, or press Shift+F6, then save. TMS creates a delete statement for the term and its relations in the pre-dictionary tables.

> **Note:** If you accidentally delete the wrong term or relation, you can undo the change by either requerying or by exiting the window before you save the changes. When exiting, TMS prompts you to save your changes; click **No** in these situations.

Adding a Term to a Dictionary Level

To add a term to the desired dictionary level:

1. Highlight the level to which you want to add it in the tree structure.

2. Place your cursor in the Terms column of the **Current Level** block in the middle of the window and select **Insert Record**.

> **Note:** You cannot add terms directly to a group level. However, if the current level is a group level, you can modify or delete terms in any of its sublevels.

3. Fill in the fields as necessary to define the term the way you want. For descriptions of the fields, see "Current Level Block Fields" on page 12-13. If you choose the Subtype **Company**, the term will be available in all domains. If you choose the Subtype **Domain**, the term will be available only in the current domain.

4. Save. TMS enters the term in the predict contents table. You must activate the current Activation Group before you can use the new term. See "Activating Data" on page 12-33.

After adding a term to a dictionary level, you can create relations between the new company or domain term and terms in related levels. Where the dictionary structure defines level relations as mandatory, you must create a relation. Where the dictionary structure defines a level relation as requiring a Primary Link, you must create a Primary Link. See "Define Strong Relations to Terms in Other Levels" on page 12-12.

Modifying a Term

To modify a term:

1. In the tree structure, highlight the level of term you want to modify.

2. In the **Current Level** block in the middle of the screen, query for the term.

3. Highlight the term.

4. Fill in the fields as necessary to redefine the term the way you want. For descriptions of the fields, see "Current Level Block Fields" on page 12-13.

5. Save. TMS enters the updated term in the predict contents table. You must activate the current Activation Group before you can use the new term. See "Activating Data" on page 12-33.

Define Strong Relations to Terms in Other Levels

You can define a link, or relation, between terms in related levels. Relations between terms must follow the rules defined for the relation between their levels:

- If the level relation is defined as Mandatory, you must create a relation between the new term and at least one term in each level with a mandatory relation to the current level.

- If the relation between the two levels is defined as One-to-one, you can define only one relation. If the relation is defined with Many Cardinality on the side of the noncurrent level, you can create relations with any number of terms in the noncurrent level.

- If the relation is defined as requiring a Primary Link, you must define a Primary Link (this applies only to relations defined with Many Cardinality on the side of the noncurrent level). See "Defining Domain and Company Primary Links" on page 12-13.

Reading Defined Level Relations   You can see the level relation rules defined for the dictionary in the tree structure on the left side of the window.

- **Optionality**. The tree structure shows mandatory relations between levels with a solid line; optional relations have a dotted line. A relation may be optional at one end and mandatory at the other.

- **Cardinality**. The tree structure shows Many Cardinality (a level relation where a term in one level may have a relation with more than one term in another level) with a branch; if only one relation is allowed, the relation line is not branched.

  The following diagram shows a level relation that is Mandatory and Single Cardinality on the parent level end and Optional and Many Cardinality on the child level end; that is, terms in the child level must have a relation to only one term in the parent level, and terms in the parent level may or may not have one or more relations with terms in the child level.
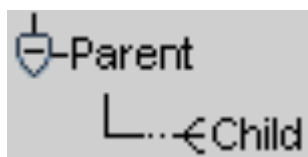
- **Primary Links**. If many relations are allowed but a **Primary Link** to one of them is required, TMS displays a **p** above the relation icon. Similarly, if a **Primary Path** link is required, TMS displays a **pp** above the relation icon.

Defining Domain and Company Primary Links   You must define a Primary Link between a company or domain term and a term in another level where the level relation requires a Primary Link. If the lower-level term is a domain term, the **Relation Subtype** is automatically entered as **Domain**, and the Primary Link applies only within the current domain, because the term exists only in the current domain.

The following rules apply to creating Primary Links (with the **PL?** box) between domain terms and company or external terms:

- A Primary Link between a lower-level domain term and a higher-level domain term (in the same domain) is allowed. The Primary Link applies only to the domain where the terms exist.

- A Primary Link between a lower-level domain term and a higher-level company or external term is allowed. The Primary Link applies only to the domain where the term exists.

- A Primary Link between a lower-level company or external term and a higher-level company or external term is allowed. The Primary Link applies to all domains.

- A Primary Link between a lower-level company or external term and a higher-level domain term is not allowed.

You can override a Global Primary Link (PL) within a particular domain by defining a Domain Primary Link (DPL), using the **DPL?** box. You can use this box only for company or external terms that already have a Global Primary Link defined, and only for a relationship that is already activated.

(You cannot create a Primary Link in a relation of subtype **Domain** between two company or external terms. However, you can create a relation of Subtype **Company** or **External**, activate the relation, and then define a Domain Primary Link (DPL) between the terms.)

Defining a Relation Between Terms   You can create relations only between terms that have already been loaded or defined and activated. You can create terms only in the **Current Level** block. To define a strong relation, take the following steps:

1. Highlight the term in the current level for which you want to create a relation to another term.

2. Put your cursor in the **Term** field in the upper block and select **Record**, then **Insert**.

3. If you know the term you want to link to, enter it.

   If not, press F9 to display a list of values. TMS displays a shortcut menu labeled **Terms and Relations**. Enter free-form text and wildcards to limit the search. For example, a search on *PH%* finds terms beginning with *ph*. Select the term to which you want to link the new term.

4. Check the information about the relation. The only value you can change is **Sub Type** (see "Sub Type" on page 12-16).

5. Save. TMS enters the relation in the predict tables.

   You must activate the relation before you can use, or create a DPL for the new relation. To activate the current Activation Group with all its terms and relations, click **Transfer Data** at the bottom of the window.

Current Level Block Fields

The **Current Level** block, in the middle of the Maintain Repository Data window, provides the following term information.

Level_Name   Under this heading, which is the name of the dictionary level, the system displays a term name in each row. Free-form text of up to 300 alphanumeric characters.

DML   Display-only. The transaction to be performed on the term. When you save changes, TMS enters a value here (for update and delete transactions, you cannot see the new value until you requery). The next time you run Activation on this Activation Group, TMS will try to perform this transaction on this term.

Level   The short name of the level to which the term belongs.

Type   Displays the term type: Dictionary Term, VTA, or VTI.

Code   Optional. Enter a value here according to your company's policy. This field is designed to serve as the primary key within this dictionary; must be unique within the dictionary. Free-form text; up to 30 alphanumeric characters. You can create an LOV and validation for this field in the Details window; see "Defining Level Details" on page 6-25.

Alt Code   Optional user-defined field. You can create an LOV and validation for this field in the Details window; see "Defining Level Details" on page 6-25. Indexed. Designed to accommodate a TMS-wide unique ID. Free-form text up to 30 alphanumeric characters.

Sub Type   TMS creates new terms in dictionary levels as Company terms unless you deselect **Glb?**, in which case TMS creates them as Domain terms. TMS creates new terms in the Verbatim Term Level as **Accepted** unless you choose **Misspelled** from the **Sub Type** list.

 See "Misspelled VTAs and VTIs" on page 10-2 for more information.

Glb?   (Global?) If selected, this term is available in all domains. If not selected, it is either available company-wide or only in the current domain, depending on the term's subtype.

Appr?   (Approved?) This option has a somewhat different meaning depending on whether it refers to a VTA or a dictionary term.

- If selected for a VTA, the VTA is approved and does not require manual approval. TMS can use the VTA in Autoclassification.

- If selected for a dictionary term at the classification level, the term is valid in the current dictionary version and approved as a potential classification in all domains.

> **Note:**   It is possible to allow classification to nonapproved dictionary terms in a specific dictionary domain. See "Assigning a Dictionary to a Domain" on page 6-38 for more information.

Status   Optional. Enter a value to describe the status of the term.

Category   Optional. MedDRA supplies categories for some of its terms, such as Diagnosis, Adverse Event, or Undefined Procedures. These are displayed here. For other dictionaries, you can assign categories to terms according to your company's policies. Free-form text; up to 65 alphanumeric characters. You can create an LOV and validation for this field in the Details window; see "Defining Level Details" on

page 6-25.

ID   Display-only. For each term, TMS creates an ID that is unique across TMS and serves as the primary key in TMS.

Comment Text   Optional. Use for any information that would be helpful to others using the dictionary. For example, "This drug should be included in the next version of WHO-Drug, and I have submitted it," or "This may seem wrong, but I've checked with the investigator." Free-form text; up to 200 alphanumeric characters.

Error Msg   Display-only. Populated by TMS during Activation. If TMS cannot activate this term, TMS leaves it in the predict tables and enters an error message in this field giving the reason(s) it failed Activation.

Created By   Display-only. The user ID of the person who added the current term or executed the load script.

Creation   Display-only. The creation timestamp for the current term.

Valid Until   Display-only. TMS uses this value to maintain an audit trail of terms. When a term is created, it receives a default end timestamp in the distant future (currently August 3500). When a term is updated, a new term record is created and the old term's end timestamp is set to the creation timestamp of the new term. This field is used to recreate the dictionary at a given point in time.

Deleted By   Display-only. The TMS user who deleted this term, if any.

Trans.Id.   Not used in this release.

Values 1-4   User-definable fields. Use these fields according to your company's policies and study design. You can define up to four fields, their data type and length, and LOV to appear here. See "Defining Level Details" on page 6-25. If no values are defined for these fields, they will not appear on the screen.

When you enter data for a term record in any of the **Value 1-4** fields, you can also choose **<null>** from the LOV.

Upper Block Fields

The upper block provides the following term and relation information:

Term   The name of the term. Free-form text of up to 300 alphanumeric characters.

Relation   The named relation instantiated between the current term and the term in this upper block row. Displays **STRONG** for strong relations.

DML   Display-only. The transaction to be performed on the relation. When you save changes, TMS enters a value here (for update and delete transactions, you cannot see the new value until you requery). The next time you run Activation on this Activation Group, TMS will try to perform this transaction on this relation.

Level   Display-only. The Dictionary Short Name and Level Short Name of the dictionary and level to which the term belongs.

Code   Optional, user-defined field. Enter a value here according to your company's policy. This field is designed to serve as the primary key within this dictionary; must be unique within the dictionary. This field is free-form text, up to 30 alphanumeric

characters, and indexed. You can create an LOV and validation for this field in the **Level Details** tab window of the Define Dictionaries form; see "Defining Level Details" on page 6-25 for detailed instructions.

PL?   The **Primary Link?**/**Primary Path Link?** box, if selected, denotes that the current term is either a Primary Link or Primary Path Link this level for the current term. Refer to the relation line between the levels in the TMS tree structure to determine the relationship between the selected levels.

DPL?   (Domain Primary Link?) If selected, this term is the Primary Link in this level for the current term in the current domain.

RGlb?   (Relation Global?) If selected, this relation is valid across all domains.

Type   Relation Type. In general, the relation inherits the type of the higher-level term. The **Domain Restricted** type is used for relations that have been defined as Domain Primary Links and have not yet been activated.

Sub Type   Relation Subtype. Initially inherits the subtype of the higher-level term. To make a relation valid only in the current domain, you can change relations of subtype **External** or **Company** to **Domain**. You can change the subtype to **External** or **Company** (both of which are global relations) from **Domain** only if the higher-level term in the relation is an external or company (global) term.

If the subtype is **Accepted** or **Misspelled**, the relation is a VTA; you can change this subtype at any time. You can also change a VTA's status from **Global** to **Domain** or vice versa in the Promote/Demote window under the **VTA Maintenance** menu.

R. Status   (Relation Status) Optional. Enter a value to describe the status of the relationship.

R. Trans. Id   (Relation Transaction ID) Not used in this release.

Alt Code   Optional, user-defined field. You can create an LOV and validation for this field in the Details window; see "Defining Level Details" on page 6-25. Indexed. Designed to accommodate a TMS-wide unique ID. Free-form text up to 30 alphanumeric characters.

ID   Display-only. For each term, TMS creates an ID that is unique across TMS and serves as the primary key in TMS.

Appr?   (Approved?) Display-only, refers to the term on this line, not the relation. This setting has a somewhat different meaning depending on whether it refers to a VTA or a dictionary term. Selected by default.

If selected for a VTA, the VTA is approved; TMS can use the VTA in Autoclassification and manual approval is not required.

If selected for a dictionary term, the term is approved as a potential classification. If you are adding one or more domain or company terms that you want to use instead of an external dictionary term, you can clear the **Appr?** box for the external dictionary term.

Category   Optional. MedDRA supplies categories for some of its terms, such as Diagnosis, Adverse Event, or Undefined Procedures. These can be displayed here. For other dictionaries, you can assign categories to terms according to your company's policies. Free-form text; up to 65 alphanumeric characters. You can create an LOV and

validation for this field in the **Level Details** tab window of the Define Dictionaries form; see "Defining Level Details" on page 6-25 for detailed instructions.

Relation Comment   Optional. Use for any information that would be helpful to others using the dictionary. Free-form text; up to 200 alphanumeric characters.

Relation Error Msg   Display-only. Populated by TMS during Activation. If TMS cannot activate this relation, TMS leaves it in the predict tables and enters an error message in this field giving the reason(s) it failed Activation.

Relation Created By   Display-only. The user ID of the person who added the current term or executed the load script.

Relation Created   Display-only. Timestamp for the creation of this relation.

Relation Valid Until   Display-only. TMS uses this value to maintain an audit trail of relations. When a relation is created, it receives a default end timestamp in the distant future (currently August 3500). When a relation is updated, a new relation record is created and the old relation's end timestamp is set to the creation timestamp of the new relation. This field is used to recreate the dictionary at a given point in time.

Relation Deleted By   Display-only. The TMS user who deleted this relation, if any.

Entry Ts   Display-only. The creation timestamp for the current term.

Values 1-4   User-definable fields. Use these fields according your company's policies and study design. You can define up to four fields, their data type and length, and LOV to appear here. See "Defining the Dictionary Levels" on page 6-20. If no values are defined for these fields, they will not appear on the screen.

Resolving High-Level Omissions

When a derivable term is not related to any terms in the next higher derivable level, TMS cannot derive a term from the higher level, and creates a high-level omission.

TMS does not return high-level omissions to the external system as discrepancies. It creates these omissions within TMS and returns values to the external system for all the derivable levels possible. When a high-level omission is resolved in TMS, the missing high-level value is derived to the external system during the next data exchange (Batch Validation in Oracle Clinical).

High-level omissions must be resolved at the master or single instance in the Maintain Repository Data window.

1. Open the Maintain Repository Data window via the **Repository Maintenance** menu.

2. Select **High Level Oms**. TMS displays the High-level Omissions window.

3. Execute a query for the high-level omissions you want to resolve. For each omission TMS displays the verbatim term, its classification level term, and the derivable higher level where there is no related term.

4. Click on the omission you want to resolve.

5. Click **OK** to return to the Maintain Repository Data window. In the block in the middle of the window, TMS displays the dictionary term that needs a relation to a term on the higher derivable level. If the current term ever had any relations in the

higher level, these are displayed in the upper block with the term and/or relation icon shown as deleted (displayed with a red **X)**.

6. Find a higher-level term to which to link the current term. There are two ways to do this:

   - Press F9 to display a list of values for terms in the level above the current term. TMS displays a shortcut menu labeled **Terms and Relations**. Enter free-form text and wildcards to limit the search. For example, a search on *PH%* finds terms beginning with *ph*.

   - Query on any field(s) in the upper block.

7. Select the higher-level term you want to derive.

8. Save.

Using the Repository Authoring Window

The Repository Authoring window enables you to instantiate strong and named relations between terms, both in strong dictionaries and in weak dictionary folders. You can also use this window to create new terms in any dictionary, maintain existing terms and relationships, launch the Maintenance Wizard, run Activation in Check or Transfer mode, and query for particular terms and relationships that may have failed Activation.

This section includes:

- Structure of the Window on page 12-18
- Creating New Terms on page 12-20
- Creating Named Relations Between Terms on page 12-20
- Defining a Primary Path on page 12-22
- Setting Up Cross-Dictionary Relations on page 12-22
- Grooming Data that Failed Activation on page 12-23

Structure of the Window

Organizationally, the Repository Authoring window has three sections that follow the creation of a relation between terms (Figure 12–3).

Figure 12–3   The Repository Authoring Window

There are three sections in the window:

- The **Filter** block at the top of the window enables you to narrow the parameters of your query.

- The **Terms** block displays information about one term in the relation. You can view detailed information about one term in the **Terms** tab window, or view all terms that match your query in the **Multi Display Terms** tab window.

- The **Relations** block displays information about the other term in the relation, as well as information about the strong or named relation between the terms. As with the **Terms** block, you can view detailed information in the **Relations** tab window, or view less detailed data about all matches by using the **Multi Display Relations** tab window. In the example shown in Figure 12–3, TMS displays relations in the lower block that contain "Cardiac disorders," the term selected in the upper block.

When you double-click a record in the **Relations** block, TMS requeries for relations that contain the other term in the selected row. In the example in Figure 12–3, if you double-click the second row shown in the **Multi Display Relations** block ("Cardiac disorders – STRONG – Congenital cardiac diseases"), TMS displays "Congenital cardiac diseases" in the **Terms** block, and displays relations that contain that term in the **Relations** block.

There are four major functions you can perform in the Repository Authoring window: creating (or modifying, or deleting) terms in the Repository, creating (or deleting) relations between terms in the same dictionary, creating (or deleting) relations between terms in different dictionaries, and querying for named relations that failed Activation.

Creating New Terms

Creating new terms using the Repository Authoring window requires the **Terms** block only. To create a new term using the Repository Authoring window:

1. From the **Repository Maintenance** menu, select **Repository Authoring**. The Repository Authoring window opens.

2. From the **Filter** block at the top of the window:

   ■ Choose **Terms** from the **Master Query** field. You can only modify Repository data when this value is set to **Terms**. The **Relations** setting is used to query for relationships that failed Activation; see "Grooming Data that Failed Activation" on page 12-23.

   ■ Choose an **Activation Group**, **Domain, Dictionary** and **Level** in which you want to create the new term.

   ■ Use the defaults for all other Filter lists.

3. In the **Terms** block, enter the required information in the fields. The fields in this block correspond to those in the current block of the Maintain Repository Data window.

4. Save. TMS records your new term in the predict tables for the selected dictionary.

> **Note:** You can modify terms in the same manner. Query for the term you want to modify, then change any of the modifiable fields and click **Save**.

Creating Named Relations Between Terms

The Repository Authoring window enables you to link dictionary terms with any strong or active named relations that have been defined in your TMS installation. If you do not want to use any of the relationships that display in the **Relationship** field, from the **Definition** menu, select **Define Named Relations** to define a new one.

If a relationship already exists between two terms, you cannot change it to a different named relation. Instead, select the relationship in the **Relations** block, delete it, then create the new one you prefer.

To create a named relation between terms:

1. If the Repository Authoring window is not open, open it now. (From the **Repository Maintenance** menu, select **Repository Authoring**.)

2. From the **Filter** block at the top of the window:

   ■ Choose **Terms** from the **Master Query** field. You can only modify Repository data when this value is set to **Terms**. The **Relations** settings is used to query for relationships that failed Activation; see "Grooming Data that Failed Activation" on page 12-23.

   ■ Choose an **Activation Group**, **Domain**, **Dictionary** and **Level** in which you want to create the new term.

   ■ In the **Level** field, you can further focus your search within the selected dictionary by specifying a level in which you want to query, or search across all levels by selecting the blank value at the end of the list (Figure 12–4).

Figure 12–4   Selecting Blank Value from the Level Field in Repository Authoring
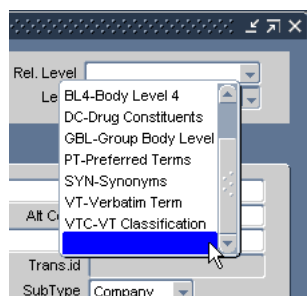
- Choose a related level in the **Rel. Level** field if you know the level of the reference term you want to use. If you leave this field blank, TMS enables searches of the entire dictionary for relations from the selected term.

- Use the defaults for all other filter lists.

3. In the **Terms** block, query for the term you want to use as the starting point of the relationship. If this term is linked to any terms in the selected reference level, TMS displays these matches in the **Relations** block. You can view all matches by clicking the **Multi Display Relations** tab.

4. Click in the bottom block and choose **Record**, then **Insert**. TMS creates a new, blank row in the **Relations** block and displays the Relations Filter at the top of the block. The Relations Filter specifies information about the named relation, including the dictionary in which your desired reference term is stored, the directionality of the relationship, the named relation you want to use, and the level in which the reference term is stored.

5. From the Relations Filter:

   a. Choose a dictionary. This example describes linking terms within the same dictionary; to view the steps required for creating a cross-dictionary link, see the next section, "Setting Up Cross-Dictionary Relations" on page 12-22.

   b. Choose a relationship and direction for the relation shown in the **Relation** fields. If you choose:

      * a uni-directional named relation and choose **From** in the Direction Filter, TMS populates your selected term on the left side of the Relation. If you choose **To** for a uni-directional relationship, the selected term appears on the right side.

      * a bi-directional named relation, your selected term appears on the left side in all cases. Making a selection from the Direction Filter in this situation controls whether the **Relationship** field displays Indicator or Reciprocal Indicator relationships.

   c. Choose a named relation to create between the selected terms.

   d. Decide if you want to use a uni-directional relationship or a bi-directional relationship. Bi-directional relationships contain reciprocal relationships in the reverse direction.

   e. Choose a level to query in the dictionary, or the blank row to query all levels.

6. In the **Relations** block rows, choose the reference term from the list of values.

7. Save. TMS inserts your new relationship into the pre-dictionary tables.

Defining a Primary Path

You can also use the Repository Authoring window to define Primary Path Links. To define a primary path, make the following changes to the directions above:

1. Choose a source term as described in the previous section.

2. In the **Rel. Level** filter, choose a group level that contains the levels in your selected derivation path.

3. Query in **Multi Display Relations** for the terms to which you want to create primary path relations.

4. Select the **PL?** flag for each term in the derivation path you want, and save.

> **Note:** You can query for primary path relations in a similar way. Choose the parent group level from the **Rel. Level** list in the **Filter Query** block, then query for the source term from the main part of the Repository Authoring window.

Deleting Terms

When you delete a term from a dictionary, you also delete all relations from that term to other dictionary terms. Before activating data after deleting a term, you may want to examine pre-dictionary DML statements to reassign the relations before they are deleted.

Further, if you delete a term from a dictionary's classification level (or group level), TMS automatically marks all associated VTAs or VTIs for deletion as well. This deletion applies both to automatically coded VTAs and VTIs and those created by manual classification. The verbatim terms that had mapped to this classification level term will become omissions upon the next Synchronization.

To delete a term and the relations to it from a dictionary, highlight the term and either select **Record** and **Delete**, or press Shift+F6, then save. TMS creates a Delete Statement for the term and its relations in the pre-dictionary tables.

> **Note:** If you accidently delete the wrong term or relation, you can undo the change by either requerying or by exiting the window before you save the changes. When exiting, TMS prompts you to save you changes; click **No** in these situations.

Setting Up Cross-Dictionary Relations

TMS enables you to define, maintain, and browse named relations between base dictionaries by using the Repository Authoring window. By using cross-dictionary links, you can migrate a study from one base dictionary to another more easily.

You can only define relations between terms whose dictionaries are linked via a cross-dictionary link. If the terms you want to relate are in dictionaries that are not linked, see "Defining a Cross-Dictionary Link" on page 6-35 to create the cross-dictionary link you need. Note that both dictionaries in a cross-dictionary link must be active base dictionaries.

You can only define named relations as cross-dictionary links; strong relationships are allowed only for intra-dictionary relationships. Definition of relationships is the same as for defining named relations within a single dictionary: you can only define one relationship at a time.

Assign Both Dictionaries to the Same Activation Group and Domain

In order to create linkage between terms in two dictionaries, you must add both dictionaries to the same Activation Group and domain. You can add dictionaries to an Activation Group in the Maintain Repository Data window, and add dictionaries to a domain in the Define Domains window.

Creating Cross-Dictionary Relations Between Terms

You can insert, modify, and delete relationships between terms in different dictionaries by using the Repository Authoring window. In the same window, you can browse cross-dictionary links by using the multi-record portion of the **Relations** block.

To insert a relation between terms in different dictionaries:

1. If the Repository Authoring window is not already open, open it now. (From the **Repository Maintenance** menu, select **Repository Authoring**.)

2. In the filter at the top of the window:

   - Choose the Activation Group and domain in which you want to work.

   - Choose the dictionary of one of the terms in the **Dictionary** field.

   - In the **Level** field, you can further focus your search within the selected dictionary by specifying a level in which you want to query, or search across all levels by selecting the blank value at the end of the list as shown in Figure 12–4.

3. In the **Terms** block, query for a term in the dictionary/level combination you specified in the **Filter** block.

   If the term selected has any relationships to terms in any dictionary including its own, these relationships are shown in the **Relations** blocks below.

   > **Note:**   Click the **Multi Display Relations** tab to view all of the terms to which the selected term is related, including any cross-dictionary relationships.

4. Select **Record**, then **Insert**, in a **Relations** block.

5. Query for the dictionary, level, and term to which you want to link, along with the type of relationship and directionality.

6. Save. TMS records the relation in the predict tables.

You can delete links between dictionaries in much the same manner; select one term in the **Terms** block, find the term to which it is linked in a **Relations** block, and select **Record**, then **Delete** to delete the link between them.

Grooming Data that Failed Activation

You can use the Repository Authoring window to examine either the terms or relations have failed Activation. Browse this data by performing the following steps:

1. From the **Filter** block at the top of the window:

   - Choose either **Terms** or **Relations** from the **Master Query** list. Errors relating to Term Activation are shown in the upper blocks, and those relating to relations in the lower blocks.

   - Choose the Activation Group, domain, and dictionary that you want to query.

■ In the **Data Source** list, choose **Pre-Dictionary Data**. Data that fails Activation remains in the pre-dictionary tables.

2. In either a **Terms** or **Relations** block, query for data that failed Activation. To query for all terms or all relations that failed in this Activation Group, domain, and dictionary, enter % in the **Error Msg** field. TMS returns all data for which the **Error Msg** field is not null.

TMS displays the relations that failed Activation, with the reason for each failure listed in the **Error Msg** field. In this mode, update is not allowed. You cannot resolve these failed relations, so browse them to determine the problem with the relation before deleting them in this window.

Creating Informative Notes for Terms and Relations

You can use Informative Notes to supply additional information for a single term or relation record. All Informative Notes are based on attributes that you define database-wide; see "Defining Informative Note Attributes" on page 7-22.

Notes: You can also create Informative Notes for all terms and relations in a dictionary, or for individual VTAs and VTIs. See "Defining Dictionary-wide Informative Notes" on page 7-26 or "Using the Status/Notes Pop-up Window" on page 10-21.

TMS does not allow you to associate Informative Notes with two types of relation: Domain Primary Links, and relations that are part of a primary path.

---

1. In either the Maintain Repository Data or Repository Authoring window, query for and select a term.

2. Click the **Informative Notes** button.

The Maintain Informative Notes window **Multi Display** tab opens, listing all Informative Notes that currently apply for the selected record, if any, except dictionary-wide notes inherited by the record. To see those, go to the Browse Informative Notes window; see "Browsing Informative Notes" on page 13-1.

3. Click in an empty row to create a new Informative Note.

4. In the **Informative Note** field, click the ellipsis (…). The list of values displays all the Informative Note Attributes available in the current dictionary/domain combination and with Applies To defined for the appropriate item: term, relation, term history, algorithm, or dictionary.

5. Find an attribute to use, and click **OK**. TMS populates the **Informative Note** and **Type** fields with your attribute selection.

6. Enter a **Value**:

■ For most **Standard** type notes, enter the text, date, or numeric value to display in the note.

■ For **Standard** type notes with a data type Memo, TMS displays the word Memo in blue type. To enter the text:

a. Double-click the word Memo or select the drill-down option. TMS opens the Memo (*Attribute Name*) window.

b. Enter or paste the text you want to store in this field.

      c. Click **OK**. The text is stored as a CLOB (character large object).

- For **URL** type notes, enter the URL beginning with a protocol (http:// or ftp://). To use variables, see "Entering Variables in URL Informative Notes" on page 12-25.

- For **Algorithm** type notes, enter the actual algorithm in a form TMS can interpret; see "Defining Informative Notes for SMQ Algorithms" on page 6-7.

7. Select the **All Versions?** box to apply this Informative Note to this term or relation despite any changes to the record.

Clear the box if you want the Informative Note to apply only to this version of the term or relation.

8. Enter a status for this Informative Note in the **Status** field (optional).

9. Save. TMS saves the Informative Note to the pre-dictionary tables. To move the record to production, you must run Activation.

Entering Variables in URL Informative Notes

You can use variables to add information in the text of the URL. For example, if you include the variable %TERM% in a dictionary-wide URL Informative Note, TMS substitutes the term name in the text of the URL. Thus, when you define the dictionary-wide URL Informative Note `https://www.google.com/#q=%TERM%`, the term `rubeola` inherits `https://www.google.com/#q=rubeola` as an Informative Note. You can connect to this URL from the Browse Informative Notes window.

During URL Informative Note definition, TMS restricts the available variables available.

In URLs for either terms or relations, you can include the following variables that map columns in the TMS_DEF_DETAILS table and apply to Informative Notes: DICT_INFO_HDR_ID, LABEL_TEXT, DEF_DETAIL_ID, LABEL_TEXT_UPPER.

For terms, TMS allows you to include the following variables that map to their counterpart columns in the TMS_DICT_CONTENTS table: TERM, TERM_UPPER, CATEGORY, COMMENT_TEXT, DICT_CONTENT_ID, DICT_CONTENT_CODE, DICT_CONTENT_ALT_CODE, and the VALUE_1 through VALUE_4 columns.

For relations, TMS allows the following variables that map to their counterpart columns in the TMS_DICT_RELATIONS tables, or to columns in the TMS_DICT_CONTENTS tables for the terms in the relation. Where _REF follows the term, you can include the same information from the reference term. The list of values includes: TERM (_REF), TERM_UPPER (_REF), DICT_CONTENT_ID (_REF), DICT_CONTENT_CODE (_REF), DICT_CONTENT_ALT_CODE (_REF), TERM_COMMENT_TEXT (_REF), DICT_RELATION_ID, and COMMENT TEXT.

To add a variable to a URL Informative Note:

1. Click in the **Value** field and enter the basic URL.

2. While the cursor is in the **Value** field, press F9 or select **Edit** to see the list of values.

3. Select a variable and click **OK**. TMS appends *%variable%* to any text in the **Value** field.

You can also cut and paste variables in the **Value** field to create a valid URL.

Using the Maintenance Wizard

The Maintenance Wizard performs the moving and exchange of terms selected in either the Maintain Repository Data or Repository Authoring window. Using this tool, you can perform six different functions: move relations, move terms up or down a dictionary hierarchy, move a term and copy its relations, move and merge a term, merge a term, or exchange terms. Maintenance Wizard functions are available for strong relations in strong dictionaries only.

You can launch the Maintenance Wizard from either the Maintain Repository Data window or the Repository Authoring window. This section includes the following Maintenance Wizard topics:

- Overview of Maintenance Wizard Tasks on page 12-26

- Move Relation on page 12-27

- Move Term on page 12-27

- Move Term and Copy Relations on page 12-28

- Move and Merge Term on page 12-28

- Merge Term on page 12-28

- Exchange Terms on page 12-29

- Undoing a Change Before Activation on page 12-29

Overview of Maintenance Wizard Tasks

To launch the Maintenance Wizard:

1. Select the term you want to modify, and click the **Maintain Wizard** button at the bottom of the Maintain Repository Data window. The Maintain Data window opens, with your term's Name, ID Number, Current Level, and Level Code at the top of the window.

2. Select a function to perform on this term, and click **Next**.

3. Complete the steps in the appropriate section:

   Move Relation on page 12-27

   Move Term on page 12-27

   Move Term and Copy Relations on page 12-28

   Move and Merge Term on page 12-28

   Merge Term on page 12-28

   Exchange Terms on page 12-29

Data Pending in the Predict Tables

If the changes you elect to make using the Maintenance Wizard either involve terms that are awaiting Activation in the predict tables or would create duplicates of terms that already exist, TMS will prompt you with an error message. Close the Maintenance Wizard and refresh data in the Maintain Repository Data window to determine the problem before you continue.

Move Relation

The Move Relation function enables you to delete one relation between the selected term and a term in a related level and to create a new relation on the same related level. To move a relation:

1. In the first window of Maintain Data, select **Move Relation** and click **Next**.

2. Select a level from the **Related Level** list, highlight the relation you want to remove, and click **Next**.

3. Query for a new term to which you want to link the selected term, and click **Next**. TMS summarizes your changes in the Maintain Data window.

4. Click **Finish** to execute the changes.

Move Term

When you move a term to a new level:

■ TMS deletes the term from its original level and deletes all the relations to the deleted term, with the following exception: TMS does not delete an original relation when the relation to the new level is still valid—for example, when the term has moved between levels of a group. In this case, the relation moves with the term.

For example, in a drug dictionary the classification level is a group level with two sublevels, Preferred Name and Synonym. You move Synonym A to the Preferred Name level, using the Move Term function. Verbatim Term A, which had been a child of Synonym A, becomes a child of the new term Preferred Name A.

> **Caution:** When you move a term to a new level, all its old relations are deleted (unless the relation is a VTA and you are moving the term from one sublevel of a classification group to another). If its old relations were optional, the new term and relations will be activated and *you may lose relations that you need.* If the old relations were mandatory, *the new terms and relations cannot be activated unless you create current relations.*
>
> You can use the Activation Check mode to see whether Activation will work without actually activating the new terms and relations. You may want to set up an Activation Group just for this purpose. See "Activating Data" on page 12-33 and "Creating a New Activation Group" on page 12-8.

TMS validates the new term and relations during Activation. To undo the moving of a term before Activation, see "Undoing a Change Before Activation" on page 12-29.

To move a term:

1. In the Maintain Data window, select **Move Term** and click **Next**.

2. Select the **Related Level** to which you want to move the selected term, from the list.

3. If you want TMS to delete all VTAs that would be orphaned as part of moving this term, select the **Delete VTAs when Mapped** box. TMS does not delete any VTAs whose relations are preserved.

If you choose not to select this box, you must manually reassign VTAs before performing Activation. Click **Next**.

4. TMS summarizes the changes. Click **Finish** to execute the move.

Move Term and Copy Relations

The Move Term and Copy Relations function expands upon the Move Term functionality by enabling the moved term to inherit all of the relationships of one of its parent terms. For dictionaries with group coding levels such as the MedDRA primary path dictionary, this inheritance is important because it ensures that you do not have to re-create the relationships for every term you move.

1. Select **Move Term** and **Copy Relations** from the Maintain Data window, and click **Next**.

2. Select a **Related Level** from the list.

3. If you want TMS to delete all VTAs that would be orphaned as part of moving this term, select the **Delete VTAs when Mapped** box. TMS does not delete any VTAs whose relations are preserved.

   If you choose not to select this box, you must manually reassign VTAs before performing Activation. Click **Next**.

4. TMS summarizes your selections in the final window. Click **Finish** to execute your changes.

Move and Merge Term

The Move and Merge Term command:

- Moves a term to a related level, and

- Merges all relations of the moved term to a different term on the same level

For example, if you move the MedDRA High Level Term *HLT1* up to the High Level Group Term level, the Move and Merge Term function enables you to move all relations of *HLT1* to another term on the High Level Term level.

To perform the Move and Merge Term function:

1. Select **Move and Merge Term** from the Maintain Data window, and click **Next**.

2. Select a **Related Level** from the list.

3. Query and select the term to which you want to merge the relations, and click **Next**.

4. TMS summarizes your selections in the final window. Click **Finish** to execute your changes.

Merge Term

The Merge Term function enables you to delete a term and merge all of the deleted term's relations to a different term on the same level. When you perform the Merge Term function, TMS deletes your selected term; you select the term to which you want to merge relations from the Maintenance Wizard.

To perform the Merge Term function:

1. Select **Merge Term** from the Maintain Data window, and click **Next**.

2. Query and select the term to which you want to merge relations, and click **Next**.

3. TMS summarizes your selections in the final window. Click **Finish** to execute your changes.

Exchange Terms

When you use the Exchange Terms function, TMS moves the parent term to its child level and creates relations with all its former child's related terms, while moving the child term to its parent level and creating relations with all its former parent's related terms. In other words, TMS exchanges the two terms while maintaining the same relations in related levels.

TMS validates the new term and relations during Activation. To undo the moving of a term before Activation, see "Undoing a Change Before Activation" on page 12-29.

To exchange terms:

1. Select **Exchange Terms** from the Maintain Data window, and click **Next**.

2. Select the level with which you want to exchange a term. TMS populates the list with applicable terms.

3. Select the term that you want to exchange with the selected term, and click **Next**.

4. TMS summarizes your selections in the final window. Click **Finish** to execute your changes.

Undoing a Change Before Activation

To undo a change made in the Maintenance Wizard, manually delete all affected terms in the predict table.

1. In the Maintain Repository Data window, select the same Activation Group you selected when you originally made the change.

2. For each term:

    a. In the middle block, query for the term in its new level. It will have an entry in the DML column.

    b. Highlight the term and delete it, using **Record**, and then **Delete**.

3. For each relation:

    a. In the upper block, query for the relation by querying for the higher-level term.

    b. Highlight the term and delete it, using **Record**, and then **Delete**.

Viewing and Deleting Dictionary Loading Error Logs

TMS generates an error log whenever one or more errors arise during dictionary loading. This section includes:

- Browsing Dictionary Loading Error Logs on page 12-29
- Deleting Dictionary Loading Error Logs on page 12-30

Browsing Dictionary Loading Error Logs

To browse the contents of a Dictionary Loading Error Log:

1. From the **Repository Maintenance** menu, select **Maintain Dictionary Loading Error Logs**. The window opens with the upper part in Query mode.

2. In the upper part of the window, query for the log you want to examine. You can query by Error Log ID, the User ID of the TMS user who loaded the dictionary, or the time and date that the dictionary was loaded.

3. Click in the row for the error log you want to examine. Error details are displayed in the lower part of the window.

Deleting Dictionary Loading Error Logs

You can also delete specific error logs by clicking in the log's row and selecting **Record**, then **Delete**, or delete all Dictionary Loading Error Logs by clicking the **Delete All** button at the bottom of the window.

> **Note:** The **Delete All** button deletes all of the Dictionary Loading Error Logs for all users in the installation.

Using the Release Label Authoring Window

This section includes:

- Defining a Release Label Named Relation Between Two Terms on page 12-30
- Checking Data Before Activation on page 12-32
- Transferring Data on page 12-32
- Additional Fields on page 12-32

You can use the Release Label Authoring window (under the Repository Maintenance menu) to create named relations of type Release Label between dictionary terms.

Release Label named relations allow you to track complex changes to dictionary terms in different versions of the same dictionary such as:

- merging two terms into a single term
- splitting a single term into two terms
- replacing one term with another term

Because these changes involve multiple inserts and deletes in a single audit transaction, TMS cannot automatically maintain a record that these terms are related. You can create relations here and view them in the HTML Browser using the Terminology search facility; see "Related Release Label Terms" on page 14-35.

You can also create relations between terms in different dictionaries in this window. You can view those relations in the HTML Browser as well.

TMS applies the named relation from the first term listed in the window (in the section that starts with the Domain field) toward the second term listed in the window (in the section that starts with the Ref. Domain field).

Defining a Release Label Named Relation Between Two Terms

To define a relation between terms using the RL named relationship, complete the following steps:

1. From the **Repository Maintenance** menu, select **Release Label Authoring**.

2. Across the top of the screen, select the a value for each of the following:

- **Group**. Select the Activation Group to use to activate this new relation.

- **Source**. Select either **Pre Dictionary Data**, **Dictionary Data**, or **All Data**. This value determines whether TMS retrieves terms from the predictionary tables, production tables, or both.

- **Named Relation**. Select the predefined named relation that you want to apply to the relation you are defining. TMS displays only named relations of type Release Label.

Note: If you select a Release Label that is not linked to a dictionary that is linked to the Activation Group you selected, the list of values in the Dictionary and Ref. Dictionary fields will not contain any values. You must either select a different Release Label or link the Release Label you selected to the dictionary you need; see "Defining Dictionary NRLs" on page 7-20.

---

3. The window includes two identical groups of fields, one for the first term in the relation and one for the reference term. TMS applies the named relation from the term in the Domain section toward the term in the Ref. Domain section.

Enter fields in both sections as follows. You can click in most fields to activate the link to the list of values on the right-hand side of the field.

- **Domain**. Select the domain associated with the term or select **Global** if the term is a global term. The list of values is restricted by the Activation Group you selected in the Group field.

- **Dictionary**. Select the dictionary that includes the term. The list of values is restricted by the domain you selected and the dictionaries associated with the named relationship you selected. It displays Release Labels and cut-off dates for all dictionary versions.

- **Release Label**. The system displays the Release Label of the dictionary/Release Label combination you selected in the Dictionary field. You can select a different Release Label of the same dictionary from the list of values or select NEXT or LATEST:

- **NEXT**. Use NEXT when one side of the relationship is in the predictionary table. When you click Transfer Data, the Activation process sets the Release Label to the one it creates for that Activation.

  For example, you can create the RL named relations from MedDRA 7 HLTs "Liver and spleen infections" and "Gallbladder infections" to the MedDRA 8 single term, "Hepatobiliary and spleen infections" before you activate MedDRA, 8 and include the named relations in the same Activation Group as MedDRA 8, specifying NEXT for the MedDRA Release Label. See "Release Label Named Relation Example 1" on page 7-18 for more information.

- **LATEST**. TMS sets the relation timestamp to the latest (current) version of the dictionary.

- **Cut-off Date**. The system automatically displays the cut-off date associated with the dictionary Release Label you specify.

- **Level**. Select the level of the dictionary that contains the term.

- **Term**. Select the term. TMS enters the appropriate subtype for the term in the **Term Subtype** field.

4. Use the instructions above to enter values that apply to the reference term in the relation in the reference section (beginning with **Ref. Domain**).

5. Select a **Relation Subtype**:

   - **Company**. Available for global terms only. Select **Company** if the relation is between two company terms or between a company term and an external term.

   - **Domain**. Select **Domain** if the relation is between two domain terms or if one of the terms is a domain term.

   - **External**. Available for global terms only. Select **External** if the relation is between two external terms.

6. Save. TMS saves the new data with the Activation Group in the predict tables.

Checking Data Before Activation

Click the **Check Data** button to verify the data. If TMS detects any errors, it displays a notification in the **Error Msg** field. See "Activating Data" on page 12-33 for more information.

Transferring Data

To transfer the defined relation and any other terms and relations currently in the selected Activation Group from the predictionary tables to the production tables, click the **Transfer Data** button. See "Activating Data" on page 12-33 for more information.

Additional Fields

The following read-only fields are displayed at the bottom of the window:

DML   This field displays the DML transaction being performed on the defined relation.

TMS uses a series of Insert and Delete transactions (collectively known as DML--Data Manipulation Language--transactions) to move data from the predictionary tables to the production tables. Each of these transactions are explained as follows:

■ When a predictionary record is saved, the **DML** value changes to **Insert**.

■ After a record is transferred into the production tables, the **DML** field displays a null value.

■ When a record is deleted from the production tables, a replica of that record gets automatically inserted within the predictionary tables. This replica is created with a **DML** value of **Delete**.

Error Msg   TMS displays a notification in this field if any errors are encountered on your clicking the **Check Data** button.

Created By   This field automatically displays the user name of the individual who created the relation. It is populated after the relation is saved the first time.

Creation   This field automatically displays the date and time the relation was created. It is populated after the relation is saved the first time.

Deleted By   This field is populated only when a relation is deleted from the production tables and activation/transfer of data is performed after such a deletion. It displays the user name of the individual who deleted the relation.

Valid Until   This field automatically displays the date and time until which the relation remains valid. It is populated after the relation is saved the first time.

Activating Data

The Activation process includes validation; TMS will not activate terms and relations that violate level definitions and other user-defined rules. See "About Activation" on page 6-33.

When you run Activation in Check mode, TMS invokes the Activation process but stops short of transferring the data to the production tables. You can view results of a Check mode Activation in either Maintain Repository Data or Repository Authoring windows, or by running the Preliminary Repository Report. In the Maintain windows, data that went through Activation still has DML statements displayed. Data that would fail Activation also now has error messages. You can make the necessary changes before activating the data using Transfer mode.

In Transfer mode, TMS runs the Activation process to completion, moving valid preliminary data to the production table. Invalid preliminary data remains in the predict tables associated with an error message.

There are two ways to invoke Activation in the GUI:

■ Click the **Transfer Data** button in the Maintain Repository Data or Repository Authoring windows. TMS immediately runs the Activation job on the current Activation Group and transfers valid preliminary data to the production tables.

   You can also run Activation in Check mode from the Repository Authoring window by clicking the **Check Data** button at the bottom of the window.

■ Use the Activate Data batch job from the menu. This option allows you to choose any Activation Group and run the job in either Check or Transfer mode.

To invoke Activation from the menu:

1. From the **Repository Maintenance** menu, select **Activate Preliminary Data**. The Activation batch job window appears.

2. In the first line enter the Activation Group you want to activate from the list. See "Modifying Repository Data in the Maintain Repository Data Window" on page 12-9 for information about Activation Groups.

3. In the **Activation Mode** field, choose **Check** to run in Check mode or **Transfer** to run in Transfer mode.

4. Schedule or submit the job. For more information on running and scheduling batch jobs, see "Setting Up and Running Reports and Other Batch Jobs" on page 2-13.

You can also run Activation from a SQL*Plus prompt by issuing the following command:

```
exec tms_user_activation.activateTerms(pActivationGroupId, pMode)
```

where pActivationGroupId is the internal ID number of the Activation Group, and pMode is 'C' for Check mode, and 'T' for Transfer mode.

Repository Maintenance Reports

The Repository Maintenance Reports are:

- Preliminary Repository Report on page 12-34
- Dictionary Version and Release Information Report on page 12-35

Preliminary Repository Report

In an Activation process, some terms and relations in the predictionary data fail activation or are not validated. The Preliminary Repository Report lists these terms and relations within a specified Activation Group. Specify the Activation Data Scope to view a set of predictionary data. To run this report, you must have maintenance privileges. The Preliminary Repository Report displays the following:

- **Errors Only**: These are terms and relations that have failed activation or have not been validated via the Check option in the Activation process. The Errors list also includes terms and relations associated with Informative Notes that have not been activated and validated.

- **All**: A list of terms and relations that have add, delete or update (DML) statements against them but do not get activated. Also, those terms and relations associated with Informative Notes that have DML statements but remain unactivated, are listed.

The Preliminary Repository Report displays data in four sections.

- **Pre-dictionary Contents:** Displays term names sorted by Activation Group, Domain, Dictionary and Dictionary level. For each term, the report also lists DML statements and error message.

- **Pre-dictionary Relations:** Displays relations sorted by Activation Group, Domain, Dictionary and Dictionary level. The report lists the source term and the reference term with their respective dictionary levels, the DML statement for the relation and the error message.

- **Dictionary Contents with Pre-dictionary Informative notes:** Displays content similar to that in the **Pre-dictionary Contents** section.

- **Dictionary Relations with Pre-dictionary Informative Notes:** Displays content similar to that in the **Pre-dictionary Relations** section.

The Preliminary Repository Report does not describe the Informative Notes in detail. To view the Notes, select the term or relation in the **Repository Authoring window**. Click on **Informative Notes**. The **Maintain Informative Notes** window displays information for the selected term.

To generate a Preliminary Repository Report:

1. Select **Repository maintenance** in the navigator window.

2. Select **Preliminary Repository Report**.

3. Fill in the general report specifications. See "Setting Parameters" on page 2-13.

4. Enter a value for **Job Specific** parameters:

   - **Activation Group**: Select an Activation Group from the list.

   - **Activation Data Scope**: Select the kind of data to be returned. The options are **All** and **Errors Only**.

   - **Template**: Select the template you want to use for this report. If your company has created a custom template, it appears in the list of values. The Oracle Template is the default template.

5. Submit the job. See "Running a Job" on page 2-16.

Dictionary Version and Release Information Report

TMS creates a new dictionary version each time you do one or more of the following and run Activation:

- add, modify, or delete a dictionary term

- modify a dictionary relation

- modify a cross-dictionary relation

- modify a dictionary term Informative Note

For example, suppose your current dictionary version is MedDRA 9. When you add a new dictionary term and run Activation for the first time, a new build of the dictionary is created (build =1) and now the dictionary version is MedDRA 9.1.

> **Note:** A dictionary is assigned a **version number** which is the same as its **Release Label Prefix**. For every subsequent activation of the dictionary a **build number** is generated. This build number is appended to the **Release Label Prefix** to provide the **Release Label**.
>
> In our example, MedDRA9.1, *9.* is the **Release Label Prefix** and *1* is the build number. The **Release Label** is *9.1*.

The **Dictionary Version and Release Information Report** tracks dictionary versions. The unique identifier for each version is the date and time stamp. If you have associated a virtual dictionary with your dictionary version, then it is also listed.

The **Dictionary Version and Release Information Report** displays the following data:

- Release Labels with cut-off dates

- Virtual Dictionaries, if any

The report identifies the name of the active dictionary that it was run for. It then lists the following information in a row:

- **Dictionary Name:** The full name of the active dictionary.

- **Dictionary Short Name:** The short name of the active dictionary as entered in the Define Dictionaries form.

- **Release Label Prefix**: This is the label prefix of the dictionary as entered in the Define Dictionaries form. See "Defining a Dictionary" on page 6-17.

  For each build of the active dictionary, the report gives the following information:

- **Release Label**: This column shows the Release Label of the dictionary version.

- **Cut-off Date**: This column lists the cut-off date and time for each release.

- **Virtual Dictionary Information**: In these columns you can see information related to any virtual dictionaries associated with the release.

  - **Dictionary Name**: This is the virtual dictionary name.

  - **Short Name**: This is the short name for the virtual dictionary as entered in the Define Dictionaries form.

To generate a **Dictionary Version and Release Information Report**:

1. Select the **Dictionary Version and Release Information Report** from the **Repository Maintenance** menu.

2. Fill in the general report specifications. See "Setting Parameters" on page 2-13.

3. Enter a value for each **Job Specific** parameter:

   - **Active Dictionaries:** Select the dictionary for which you want to run this report.

   - **Template:** Select the template for the report. If your company has created a custom template, it appears in the list of values. The Oracle template is the default.

4. Submit the job. See "Running a Job" on page 2-16.

Translation Reports to Identify Inconsistent Data

TMS uses Translation Derivation Links to associate terms from the global and local dictionaries via the dictionary code. Even though TMS requires that these two dictionaries have the same structure, the system does not validate that the dictionaries' term and relation data match exactly. Thus, for example, when TMS attempts to derive a term in the local dictionary with a matching dictionary content code, the corresponding term might not exist in the linked dictionary, or reside on a different dictionary level, or in a different domain.

TMS provides three reports that display many of these inconsistencies. All three reports are available under the **Translation Reports** node, which is accessible to all users who can access the **Repository Maintenance** node (that is, users with the tms_maintain_priv role).

- The Inconsistent Dictionary Codes Report shows the terms with dictionary content codes that exist in one dictionary but not the other, or terms in the two dictionaries with the same dictionary content code that exist in different domains or dictionary levels.

■ The Duplicate and Null Dictionary Codes Report shows the dictionary contents for codes that exist more than once in the same domain/dictionary combination. If two or more terms have null codes, the report includes these terms in the report as well.

■ The Inconsistent Dictionary Relations Report shows the relations to a term in one dictionary that are different than those relations to the same term in the other dictionary. The report includes differences in the relation's domain, related term, level, or relation type.

You can generate any of these reports in PDF or HTML format, and output in Preview mode, to a file, or to a printer. All three reports are resource-intensive, so run them sparingly.

Choosing Input Dictionaries

Both the Inconsistent Dictionary Codes Report and Inconsistent Dictionary Relations Report require two input dictionaries. For TMS to generate either report properly, the input dictionaries must have a Translation Derivation Link defined. If the input dictionaries are not linked, or if you enter the same dictionary twice, TMS outputs the report with an error message that instructs you to choose two different input dictionaries with a Translation Derivation Link between them.

Inconsistent Dictionary Codes Report

This report compares terms in the two dictionaries that share the same dictionary code. There should be a one-to-one relationship between the code values in these dictionaries, and the terms should reside in the same domain (whether or not the terms are global) and dictionary level. The Inconsistent Dictionary Codes Report thus generates a list of terms in each dictionary whose dictionary codes:

■ exist in one dictionary but not the other.

■ match, but reside in different domains or levels.

To run the Inconsistent Dictionary Codes Report:

1. Select **Inconsistent Dictionary Codes** from the **Translation Reports** menu.

2. Fill in the general report specifications. See "Setting Parameters" on page 2-13.

3. Enter a value for each **Job Specific** parameter:

   ■ Translation Dictionary

   ■ Second Translation Dictionary

   A Translation Derivation Link must exist between these dictionaries for the report to run correctly.

4. Submit the job. See "Running a Job" on page 2-16.

Duplicate and Null Dictionary Codes Report

For Translation Derivation to work correctly, dictionary codes should be unique within a dictionary. The Duplicate and Null Dictionary Codes Report examines all of the dictionary content codes in one dictionary, and outputs term information where terms share the same code. If two or more terms have null codes, the report includes those terms in the report as well. For each duplicate dictionary code, the report flags the code, its Dictionary Level, its Domain, and the number of times it appears in the dictionary.

To run the Duplicate and Null Dictionary Codes Report:

1. From the **Translation Reports** menu, select **Duplicate and Null Dictionary Codes**.

2. Fill in the general report specifications. See "Setting Parameters" on page 2-13.

3. Enter a value for each **Job Specific** parameter:

   - Translation Dictionary

   - **Template**

4. Submit the job. See "Running a Job" on page 2-16.

Inconsistent Dictionary Relations Report

Because TMS uses the dictionary hierarchies when deriving translation values, it is important that relations in these dictionaries match exactly. The **Inconsistent Dictionary Relations Report** shows relations to a term in one dictionary that are different than a relation to the same term in the linked dictionary. The report only considers relations between dictionary terms; VTAs are not included.

Any of the following qualifies two relations as inconsistent:

- The terms in the relation are not the same.

- The relations are in different domains.

- The terms in the relation are the same in both dictionaries, but terms with the same dictionary code reside on different levels in the two dictionaries.

- The relation types are different.

For both dictionaries, the report lists each inconsistent relation's Domain and Relation Type, and the Code, Level, and Term Name for the terms involved.

To run the Inconsistent Dictionary Relations Report:

1. From the **Translation Reports** menu, select **Inconsistent Dictionary Relations**.

2. Fill in the general report specifications. See "Setting Parameters" on page 2-13.

3. Enter a value for each **Job Specific** parameter:

   - Translation Dictionary

   - Second Translation Dictionary

   A Translation Derivation Link must exist between these dictionaries for the report to run correctly.

4. Submit the job. See "Running a Job" on page 2-16.

Part III

Browsing the Data Repository

This section describes the two interfaces available for browsing the Oracle Thesaurus Management System (TMS) repository: the classic Browse Repository window and the HTML Browser.

Part III contains the following chapters:

13

Browsing the Repository in TMS

Most data browsing windows are available to all Oracle Thesaurus Management System (TMS) users, and reside under the **Repository** menu in TMS. The functions all users can perform include:

- Viewing Data in the Browse Repository Data Window on page 13-3
- Browsing Informative Notes on page 13-1
- Viewing Data in the Browse VT Classification Data Window on page 13-13
- Browsing Verbatim Term History on page 13-14
- Repository Reports on page 13-14

If you have the tms_maintain_priv role, you can also browse the TMS Repository using the Maintain Repository Data and Repository Authoring windows. These are the only windows where you can see preliminary data. See Chapter 12, "Repository Maintenance" for more information.

You can view different types of data in the three Repository windows:

Table 13–1   Types of Data Displayed in the Three Repository Windows

Data Displayed	Browse Repository Data	Browse VT Classification Data	Maintain Repository Data
Dictionary Terms (User-defined And External)	Yes, All	No	Yes, All
Verbatim Terms with VTAs	Yes	Yes	Yes
Omissions	No	Yes	No
Deleted Data	Yes, Optional	No	Yes, Optional
Preliminary Data	No	No	Yes, Optional
Named Relations	Yes	No	Yes
Ext. System Information	No	Yes	No

Browsing Informative Notes

Informative Notes are data structures that add detail to terms, relations, VTAs, and dictionaries in the TMS Repository. Browsing a term or relation's Informative Notes in the Browse Informative Notes window enables you to:

- Hyperlink to internal or external documents about this data record.

- Browse character large objects defined for this term or relation.

- View the Informative Notes that were defined for a record at any point in the history of a TMS dictionary.

To browse the Informative Notes for a term or relation record:

1. In the Browse Repository Data window, select the term or relation you want to investigate:

   - To browse a term's Informative Notes, query for and select the term in the middle block.

   - To browse a relation's Informative Notes, query for one of the relation's terms in the middle block, then choose the relation from the upper block.

2. Click **Informative Notes**. TMS opens the Browse Informative Notes window and displays Informative Notes associated with your selected data record.

   For term records, TMS displays both the Informative Notes the term inherits from its dictionary and those specifically defined for the term. For relation records, TMS displays only the Informative Notes defined specifically for the relation. Figure 13–1 shows the Informative Notes for a relation record.

Figure 13–1   Browse Informative Notes Window for a Relation Record

This relation, United States – *Broader Term Partitive* – Massachusetts, has three Informative Notes attributed to it:

- A related site for this relation, which you can click to launch a Web browser showing the listed location.

- An ID that reflects the source dictionary's ID number for this relation.

- A character large object, giving more information about the approval of this relation in the TMS Repository. Double-click the word MEMO in the **Value** field or choose the drill-down option to view the contents of this Informative Note.

Viewing Data in the Browse Repository Data Window

The Browse Repository Data window displays dictionary terms and verbatim terms. It offers the following special features:

- Viewing deleted data. See "Viewing Current and Historical Data" on page 13-3.

- Complete Related Term information. See "Viewing Terms Related to the Current Term" on page 13-9.

- Conducting searches for common parent terms. See "Using a Child Search" on page 13-11.

- The Date feature enables you to browse a dictionary at a certain point in time.

The Browse Repository Data window displays a domain view of the TMS Repository. That is, you can see only those terms and relations, whether they are external, company, domain terms and relations or VTAs, that are assigned to the TMS domain you are currently in. The current TMS domain is displayed in parentheses in the window's title bar. You can go to a different TMS domain at any time by selecting **Options** and then **Change Domain** from the menu.

For information about understanding and using the dictionary hierarchy tree structure display, see "Using the Tree Structure" on page 2-4.

The Browse Repository Data window has a dictionary hierarchy tree structure on the left and three data blocks on the right, and includes the following features:

- Viewing Current and Historical Data on page 13-3
- Current Level Block on page 13-4
- Level Above Block on page 13-6
- Level Below Block on page 13-8
- Changing the Sort Order on page 13-9
- Viewing Terms Related to the Current Term on page 13-9
- Understanding the Origin Field on page 13-11
- Using a Child Search on page 13-11
- Using an Extended Search on page 13-12

Viewing Current and Historical Data

You can view deleted terms and relations as well as current ones by clicking the **All Data** option in the **Data Currency** group at the top of the Browse Repository Data window. You can differentiate deleted data from current data by examining the line and *T* icons in the left-hand column. Visible black *T* and line icons indicate current terms and relations, respectively. If the line icon has been replaced by a red *X*, the relation has been deleted. If the *T* icon has been replaced by a red *X*, the term has been deleted.

In addition, you can view the terms and relations that were current or deleted at any point in a dictionary's history. Performing a query against a dictionary on a particular date enables you to view which terms were current on that date, and the terms to which they related.

To query dictionary data on a specific date:

1. Click the **Date** option at the top of the window.

2. Enter a date (in DD-MON-YYYY format) in the **Date** field.

3. Query for dictionary data in the **Current Level** block. TMS returns the dictionary data matching your query, marking the currency of terms and relations on the selected date.

Current Level Block

TMS displays the **current level** in the middle block in the window, with data displayed in the rest of the window as it relates to the current level and term.

In strong dictionaries, TMS displays relations hierarchically. The level above shows parent terms of the current term, while the level below shows the current term's child terms. In weak dictionary folders, the level above shows relations that contain the current level as a reference term, while the level below displays terms that are reference terms in relations with the current term. See "Level Above Block" on page 13-6 and "Level Below Block" on page 13-8 for information on how to specify which one you want to view.

Current Level

The **current level** is the level displayed in the **Current Level** block at the middle of the screen at any given time. You can select the current level by clicking on it in the dictionary hierarchy tree structure. The header above the first column of the **Current Level** block also displays the name of the level selected in the navigator tree.

To view data, you must then execute a query in the middle block. You can change the display order by clicking on column heads.

Current Term

The **current term** is the term displayed on a blue background in the **Current Level** block. After a query, TMS makes the top record the current term by default. You can change the current term by selecting another one with your cursor, or querying for it. TMS displays data on screen as it relates to the current level and term—that is, the upper block displays terms in the level above the current level and related to the current term, and the lower block displays terms in the level below the current level that are related to the current term.

Changing the Data Displayed

To move related terms of the current term into the **Current Level** block, double-click on the related term or click on the level you want in the tree structure while the current term is displayed on the right. You can only move in one-level increments; if you select a level two or more levels removed from the current level, TMS displays a blank screen and you must execute a query to see data. However, you can move to the display of related terms in any related level as long as you move one level at a time.

To see unrelated terms in any level, click on the level you want in the tree structure and then execute a query in the **Current Level** block.

Current Level Block Fields

The current level provides the following term information:

Level_Name   The name of the term. Free-form text of up to 300 alphanumeric characters.

Level   The short name of the level in which this term is stored.

Code Optional user-defined field designed to serve as the primary key within this dictionary; must be unique within the dictionary.

Alt Code Optional user-defined field.

Sub Type Company, Domain and External are dictionary terms. Accepted and Misspelled are VTAs.

External terms are supplied by the vendor or were part of a legacy dictionary. They are global. Company terms were created in this company and are global. Domain terms were created in this company and are available only within one or more domains.

Accepted VTAs link a correctly spelled verbatim term to a dictionary term. Misspelled VTAs link a misspelled verbatim term to a dictionary term.

Glb? (Global?) If selected, this term is available in all domains. If not selected, it is available in the current domain. (It may be available in other domains as well.)

Appr? (Approved?) This setting has a somewhat different meaning depending on whether it refers to a verbatim or a dictionary term.

If selected for a VTA, the VTA is approved and does not require manual approval. TMS can use the VTA in Autoclassification.

If selected for a dictionary term, the term is approved as a potential classification. If you are adding more domain or company terms that you want to use instead of an external dictionary term, you can deselect the **Appr?** box for the external dictionary term.

Status Optional. Enter a value to describe the status of the term.

Category Optional. MedDRA supplies categories for some of its terms, such as Diagnosis, Adverse Event, or Undefined Procedures. These are displayed here. For other dictionaries, you can assign categories to terms according to your company's policies. Free-form text; up to 65 alphanumeric characters.

Origin The manner in which this term was introduced to the TMS Repository.

ID An ID that is unique across TMS and serves as the primary key in TMS.

Comment Text Optional. Use for any information that would be helpful to others using the dictionary. For example, "This drug should be included in the next version of WHO-Drug, and I have submitted it," or "This may seem wrong, but I've checked with the investigator." Free-form text; up to 200 alphanumeric characters.

Created By The user ID of the person who added the current term or executed the load script.

Creation The creation timestamp for the current term.

Valid To Date until which the term is valid.

Deleted By The TMS user who deleted this term, if any.

Trans.Id. Not used in this release.

Values 1-4 The last four fields are available for customization and may have descriptive labels.

Level Above Block

For strong dictionaries, the block at the top of the screen displays terms related to the current term that are in the level above the current level in the dictionary hierarchy. If two (or more) levels are related to the current level just above it in the hierarchy, the current level is displayed twice (or more) in the tree structure. The level displayed in the **Level Above** block depends on which display of the current level you click in the tree structure.

For weak dictionary folders, the upper block displays relations in which the current term is the reference term for the relation. Each row represents a term and its relation to the current term. The fields in the upper block are:

Term

For dictionaries with strong definition, this field displays the parent term of the term selected in the middle block. For dictionaries with weak definition, this field displays term(s) that have the selected term (in the middle block) defined as a reference term.

Relation

The relationship between the upper block term and the selected term (in the middle block). STRONG denotes a hierarchical relationship; any other description denotes a named relation.

Level

The name of the dictionary and level in which this term is stored.

Code

Optional user-defined field designed to serve as the primary key within this dictionary for the related term; must be unique within the dictionary.

PL?

The **Primary Link?/Primary Path Link?** box, if selected, denotes that the current term is either a Primary Link or Primary Path Link in this level for the current term. Refer to the relation line between the levels in the TMS tree structure to determine the relationship between the selected levels.

DPL?

The **Domain Primary Link?** box. If selected, this term is the Primary Link in this level for the current term in the current domain.

RGlb

The **Relation Global?** box. If selected, this relation is valid across all domains.

Appr?

(**Approved?**) This setting has a somewhat different meaning depending on whether it refers to a VTA or a dictionary term. Selected by default.

If selected for a VTA, the VTA is approved; TMS can use the VTA in Autoclassification and manual approval is not required.

If selected for a dictionary term, the term is approved as a potential classification.

Type

(**Relation Type**) Inherits the type of the term on this row: Dictionary or Verbatim.

Sub Type

(**Relation Subtype**) Company, Domain and External are relations to dictionary terms. Accepted and Misspelled subtypes apply to VTAs and VTIs.

External relations are supplied by the vendor or were part of a legacy dictionary. They are global. Company relations were created in this company and are global. Domain relations were created in this company and are available only within one or more domains.

Accepted VTAs and VTIs link a correctly spelled verbatim term to a dictionary term. Misspelled VTAs and VTIs link a misspelled verbatim term to a dictionary term.

R. Status

(**Relation Status**) Optional. Enter a value to describe the status of the relationship.

R.Trans.ID

Not used in this release.

Alt Code

Optional user-defined field.

ID

An ID that is unique across TMS and serves as the primary key in TMS for the related term.

Category

Optional, user-defined. MedDRA supplies categories for some of its terms, such as Diagnosis, Adverse Event, or Undefined Procedures. These are displayed here.

Origin

The manner in which this term was introduced to the TMS Repository.

Relation Comment

User-defined comment about the relationship between these two terms. Information in this field could provide information for users browsing the TMS Repository.

Created By

The user ID of the person who added the current term or executed the load script.

Created

Timestamp for the creation of this term.

Relation Created By

The TMS user who created this relation.

Relation Created

Timestamp for the creation of this relation.

Relation Valid Until

TMS uses this value to maintain an audit trail of relations. When a relation is created, it receives a default end timestamp in the distant future (currently August 3500). When a relation is updated, a new relation record is created and the old relation's end

timestamp is set to the creation timestamp of the new relation. This field is used to recreate the dictionary at a given point in time.

Relation Deleted By

The TMS user who deleted this relation, if any.

Values 1-4

The last four fields are available for customization and may have descriptive labels.

Level Below Block

For strong dictionaries, the block at the bottom of the screen displays terms related to the current term that are in the level below the current level in the dictionary hierarchy. For weak dictionary folders, the lower block displays relations in which the current term is the source term for the relation.

Each row represents a term and its relation to the current term. The fields are:

Relation

The named relation between the selected term (in the middle block) and this term (the reference term).

Term

The reference term for this relationship.

Level

The name of the dictionary and level in which this term is stored.

Code

Optional user-defined field designed to serve as the primary key within this dictionary for the related term; must be unique within the dictionary.

RGlb?

(**Relation Global?**) If selected, this relation is valid across all domains.

Appr?

(**Approved?**) This setting has a somewhat different meaning depending on whether it refers to a VTA or a dictionary term. Selected by default.

If selected for a VTA, the VTA is approved; TMS can use the VTA in Autoclassification and manual approval is not required.

If selected for a dictionary term, the term is approved as a potential classification.

Alt Code

Optional user-defined field.

Type

(**Relation Type**). Inherits the type of the term on this row: Dictionary or Verbatim.

Sub Type

(**Relation Subtype**). Company, Domain and External are relations to dictionary terms. Accepted and Misspelled apply to VTAs and VTIs.

External relations are supplied by the vendor or were part of a legacy dictionary. They are global. Company relations were created in this company and are global. Domain relations were created in this company and are available only within one or more domains.

Accepted VTAs and VTIs link a correctly spelled verbatim term to a dictionary term. Misspelled VTAs and VTIs link a misspelled verbatim term to a dictionary term.

R. Status

(**Relation Status**) Optional. Enter a value to describe the status of the relationship.

R.Trans.ID

Not used in this release.

Category

Optional. MedDRA supplies categories for some of its terms, such as Diagnosis, Adverse Event, or Undefined Procedures. These are displayed here.

ID

An ID that is unique across TMS and serves as the primary key in TMS for the related term.

Origin

The manner in which this term was introduced to the TMS Repository.

Comment Text

Optional. Information entered by the person who added, modified, or deleted this term.

Changing the Sort Order

In the **Current Level** block, you can sort terms using some of the column heads as criteria. Click on an active column head—one that looks like a button—once to indicate that you want to sort on it (alphabetically for character fields, numerically for numeric fields). Click again to sort in the opposite order.

> **Note:** TMS sorts numbers by the first digits, no matter how many digits the number contains, so that 12345 comes before 543 in ascending order.

Viewing Terms Related to the Current Term

When you highlight a term in the **Current Level** block and click either the **Above Current** or **Below Current** button, TMS launches the Drill Down Dictionary Hierarchy window, which displays the terms that relate to the current term in the selected direction, and the related terms. The data structure depicted in this window differs somewhat for terms in strong dictionaries and weak dictionary folders.

Display of Deleted Data in Virtual Dictionaries

In virtual dictionaries, some of the terms and relations you browse may have been deleted in the base dictionary since the virtual dictionary's **cut-off date**. The Drill

Down Dictionary Hierarchy window flags deleted records with a red **X**. If the **X** appears:

- On the line connecting the terms, the relation has been deleted.

- On the **T** icon to the immediate left of the term, the term has been deleted.

Strong Dictionaries in the Drill Down Dictionary Hierarchy Window

In strong dictionaries, you can see all the current term's related terms in the levels above or below it by selecting **Above Current** or **Below Current**. A shortcut window appears and displays all related terms in all levels either above or below the current term in the tree structure. Click on a related term, and information about it appears on the right. To see information about the relation between that term and the current term, click the **Relation** tab.

In the tree structure in both the Above Current and Below Current windows, the current term is listed at the top, with the related terms listed below. This means that terms in the levels above are actually displayed below the current term. However, their level short name is clearly displayed.

The tree structure of the Level Above window also displays derivable paths. The window denotes Primary Links and Primary Path Links with a green **P** next to the relationship. Primary paths are not visible in the Level Below window.

Weak Dictionary Folders in the Drill Down Dictionary Hierarchy Window

Because weak dictionary folders have no dictionary-defined hierarchy, and thus no parent or child relations, the Drill Down Dictionary Hierarchy window displays chains of relations that share the same named relation. For example, selecting the term "France" from a weak dictionary folder of geographic terms might yield the results in Figure 13–2 when you click **Level Above**.

Figure 13–2   Sample Results in the Level Above Window for a Weak Dictionary Folder

Starting with the current term "France," the window displays all relations that use the named relation "Narrower Term," up to the term "World." You can also display any relations from "France" that use other named relations by selecting a different relationship from the Named Relation list at the top of the window (Figure 13–3).

Figure 13–3 Choosing a Named Relation in Drill Down Dictionary Hierarchy
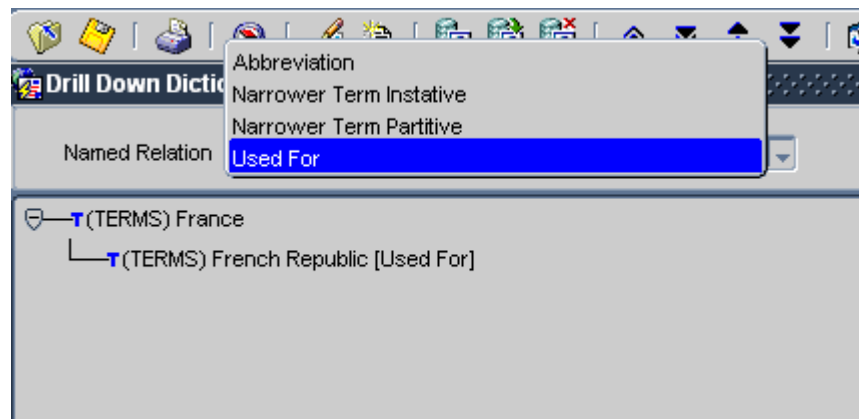
Clicking **Above Current** yields the relations that use the Reciprocal Indicator part of the named relation, while **Below Current** yields relations that use the Indicator. Thus, in the examples shown above, the named relations Abbreviation, Narrower Term Partitive, Narrower Term Instantive, and Used For are all Reciprocal Indicator relationships.

Understanding the Origin Field

The **Origin** field identifies the origin of a term in the TMS Repository. By default, the following search objects appear in the list, and you can include any other user-defined search objects in this list as well:

Origin Type	Definition
Loaded	Verbatim term was loaded into dictionary
Direct Match	Verbatim term exactly matches dictionary term or VTA
Non Appr VTA Match	Verbatim term matched to a Nonapproved verbatim term.
Domain Match	Verbatim term exactly matches an Approved VTA in another domain
Manual Match	Verbatim term manually classified to dictionary term or VTA

Using a Child Search

The **Child Search** option allows you to query for terms on a child level of the current level, and then search for their parent terms. You can choose to search for all parent terms of all selected child terms or for only those parent terms common to all selected child terms. This option is for strong dictionaries only.

To perform a child search:

1. Select a term in the current level and click **Child Search**. The Child Search window opens.

2. Open a query (F7, or select **Query** and then **Open**).

3. Filter the query by selecting the following attributes from the top of the Child Search window:

    a. If the current level has more than one child level, select the child level from which you want to start this query. You can include terms from different child levels in the same child search.

    b. Choose a **Standard** or **Context Server Query**.

4. Enter your search criteria in the right side of the window, and execute the query. TMS returns the matches that satisfy your query for child terms in the right side of the window.

5. Click in the row for any child term(s) you want to include in the search, and click the **<** button. TMS moves the term to the **Selected Term** block.

6. Select the **Any?** box to search for all parent terms of all child terms that you will select, or deselect it to search only for those parent terms common to all selected child terms.

7. Click **OK**. TMS returns you to the Browse Repository Data window and displays the results of the query—the parent terms yielded by the child search—in the middle block. If the query did not retrieve any records, the data in the middle block is unchanged.

Using an Extended Search

The Browse Repository Data window displays only one domain at a time, and restricts all queries to the selected dictionary level. To query for dictionary data across multiple domains or across all dictionary levels, you must perform an Extended Search. Extended Searches also enable you to search for terms using user-defined search objects.

After you find the results of your Extended Search and choose a term, TMS can close the Extended Search window and make that term the currently selected record in the Browse Repository Data window.

To perform an Extended Search:

1. From the Browse Repository Data window, launch the Extended Search window:

    ■ From **Options**, select **Extended Search**.

    ■ Click the **Extended Search** button.

The Extended Search window opens.

2. From the **Dictionary** list, choose the dictionary you want to search. TMS displays the current dictionary by default.

3. From the **Search Type** list, choose a user-defined search algorithm or an Open Query.

If you choose a **Search Type** other than **Open Query**, TMS uses the search object as a filter in the query. In addition, if you were working in the classification level or verbatim term level in the Browse Repository Data window, TMS populates the **Verbatim Term** field with the current term, which is also used as a filter in the query. You can modify the verbatim term, but you cannot effectively delete it. The search algorithm must have a verbatim term parameter and it will use the last one entered.

If you choose a **Search Type** of **Open Query**, TMS does not use a search object and does not use a value entered in the **Verbatim Term** field.

4. From the **Query Type** list, choose to use a standard Oracle query or a Context query that utilizes the Oracle *inter*Media Text retrieval capabilities. See "Querying

in Windows" on page 2-9.

5. Enter a query. You can limit the query by entering values in any fields in the **Text** block. For information on querying see "Querying in Windows" on page 2-9.

6. Execute the query. TMS returns the result of the query in the **Text** block.

7. If you find a term you want to view in the Browse Repository window, highlight it.

8. Do one of the following:

   - Click **OK**. The term you highlighted appears in the **Current Level** block of the Browse Repository Data window.

   - Click **Cancel**. TMS displays the same data in the Browse Repository Data window as before you executed the search.

Viewing Data in the Browse VT Classification Data Window

Use this window to view:

- verbatim terms (VTAs and VTIs).

- Unclassified verbatim terms (omissions).

- All occurrences of verbatim terms that have been collected at the local instance, if any.

- External system information associated with verbatim terms, if any.

Query in the lower block. If you have selected an external system from the list at the top of the screen, you can query on any of the external system fields. TMS displays all terms that fit the query in the top block, and in the lower block any occurrences of the top term that have been collected locally. To see local occurrences of another term, highlight it in the top block.

Fields in the upper block are:

- **Vt Upper**. A term retrieved by the query entered in the lower block.

- **?** If selected, this verbatim term has been mapped to a dictionary term.

- **Domain**. The domain with which this verbatim term is associated.

Fields in the lower block are:

- **Verbatim Term**. Occurrences of the verbatim term highlighted in the upper block that were collected at the local instance.

- **?** If selected, this verbatim term has been mapped to a dictionary term.

- **Domain**. The domain with which this verbatim term is associated.

- **Source Id**. Unique identification for this verbatim term.

- **Occur Id**. Unique identification for this occurrence of this verbatim term.

- **Integration Key**. Unique identification for the external system within TMS.

- **Additional Fields**. If an external system is available and selected from the list at the top of the window, up to eight additional fields with external system information associated with this occurrence appear here.

- **Alt Key**. Optional user-defined unique identification for this term.

Browsing Verbatim Term History

You can view any term's complete status history, including all notes associated with the term, in the Browse VT History window in the Repository menu. The window works the same way as the VT History window under the Omission Management menu except that you cannot add notes to a term. See "Using the VT History Window" on page 10-23 for more information.

Repository Reports

The reports under the **Repository** menu enable you to view changes within a base dictionary over a specified time period, to compare two dictionary versions, and to export an entire dictionary to an XML report.

This section includes:

- Date Comparison Report on page 13-14
- Dictionary Comparison Report on page 13-14
- Dictionary Export Report on page 13-15

Date Comparison Report

The Date Comparison Report displays all the changes to a base dictionary between a user-specified start and end date.

To run the Date Comparison Report:

1. Select **Date Comparison** from the **Repository** menu.

2. Fill in the general report specifications. See "Setting Parameters" on page 2-13.

3. Enter a value for **Job Specific** parameters:

   - **Base Dictionary**. Select the dictionary you want to investigate.

   - **Start Date**. The beginning date of the period you want to investigate.

   - **End Date**. The end date of the period you want to investigate.

   - **Include Global VTAs?** If selected, the report will include Global VTA information.

   - **Include Domain Primary Links?** If selected, the report will include information about Domain Primary Links in this dictionary.

   - **Template:** Select the template for the report. If your company has created a custom template, it appears in the list of values. The Oracle Template is the default.

4. Submit the job. See "Running a Job" on page 2-16.

Dictionary Comparison Report

The Dictionary Comparison Report analyzes two versions of the same dictionary, and outputs any differences in their terms and relations. For term records that have been added, deleted, or updated, this report provides all of the term's information from the TMS_DICT_CONTENTS table. For relation records that have been added or deleted, the report pulls relation information from TMS_DICT_RELATIONS.

You can only run this report for two versions of the same dictionary: either a base dictionary and one of its virtual dictionaries, or two virtual dictionaries that share the

same base dictionary. Regardless of the order in which you enter the dictionaries, TMS orders the dictionaries by cut-off date, with a base dictionary always appearing as the second dictionary in the report.

The lists of values you use to define the **First Dictionary** and **Second Dictionary** in the comparison are not restricted to allow only versions of the same dictionary. If you select virtual dictionaries that are not based on the same base dictionary, this report returns an error.

To run the Dictionary Comparison Report:

1. From the **Repository** menu, select **Dictionary Comparison**.

2. Fill in the general report specifications. See "Setting Parameters" on page 2-13.

3. Enter a value for each of the **Job Specific** parameters:

   ■ First Dictionary

   ■ Second Dictionary

   ■ **Template:** Select the template for the report. If your company has created a custom template, it appears in the list of values. The Oracle Template is the default.

   Choose the dictionaries you want to compare from the **First Dictionary** and **Second Dictionary** lists. Even though the lists display all of the dictionaries in the database, be sure to choose dictionaries that share the same base dictionary.

4. Submit the job. See "Running a Job" on page 2-16.

Dictionary Export Report

The Dictionary Export Report generates a rendition of an entire TMS base dictionary or virtual dictionary displaying dictionary contents and relations.

To run the Dictionary Export report:

1. Select **Dictionary Export** from the **Repository** menu.

2. Fill in the general report specifications. See "Setting Parameters" on page 2-13.

3. Fill in the job-specific specifications. All are required except the cut-off date of a base dictionary:

   ■ **First Dictionary**. Select the **Base** dictionary or Virtual dictionary.

   ■ **Include VTAs?** If **Yes**, VTAs will be included in the Dictionary Export Report.

4. Submit the job. See "Running a Job" on page 2-16.

14

Using the HTML Browser

The TMS HTML Browser enables you to search and browse a read-only display of the Oracle Thesaurus Management System (TMS) Repository data, and search for dictionary terms within the source data in these records in the database. Because this tool works in a browser window without requiring a plug-in, and may not require user names or passwords to view dictionary data, it provides an easy way to publish this data for use throughout your organization.

When TMS is fully integrated with an external system, you can use the HTML Browser to search for source data from that external system. See "Searching for Source Data" on page 14-29.

See also "TMS HTML Browser Administration" on page 3-46.

The following topics explain how the HTML Browser works, and how to use it:

- Overview on page 14-1
- HTML Browser Map on page 14-3
- Choosing a Type of Search on page 14-4
- Searching for Terminology Data on page 14-5
- Searching for Terms Using Filter Dictionaries on page 14-11
- Searching for Verbatim Term Assignments (VTAs and VTIs) on page 14-16
- Viewing VTA and Omission Status Information on page 14-22
- Viewing External System Information for Omissions on page 14-26
- Browsing a Dictionary Hierarchy on page 14-26
- Searching for Source Data on page 14-29
- Examining a Term's Details and History on page 14-32
- Examining a Relation's Details and History on page 14-36
- Examining a Record's Informative Notes on page 14-38

Overview

This section describes which tasks you can perform in the HTML Browser, how it structures information in a Web browser window, and how to edit the list of dictionaries available for browsing.

Available Repository Data and Verbatim Terms

This section describes which dictionaries you can search and browse in HTML Browser windows.

User Roles

Any user with the RXCLIN_READ role can browse and search for dictionary data. Oracle Clinical users with the RXCLIN roles (such as RXCLIN_READ and RXCLIN_ MOD) have access to dictionary data, but cannot search for source data without a TMS role. See "Available External System Data" on page 14-2.

Accessible Dictionaries

The HTML Browser can display Repository data, including verbatim terms, for all of a database's active dictionaries that have the **Accessible to Light Browser?** option selected. If the dictionary is accessible and has a verbatim term level, you can use it to search for verbatim term data.

You can change the **Accessible to Light Browser?** setting for any dictionary by selecting that box for the dictionary in the Define Dictionaries window of TMS. You may want to leave this option deselected for dictionaries that you want to keep secure.

As long as a dictionary is accessible, you can browse or search it in the HTML Browser. This includes base and virtual dictionaries, and strong and dynamic dictionaries. See Chapter 6, "Defining and Loading Dictionaries" for an overview of the types of dictionaries you can define in TMS, and what characteristics distinguish them.

Autoquerying Dictionaries

When you autoquery a dictionary in the HTML Browser, it displays all of the terms in the top level of that dictionary. Because only strong dictionaries contain top levels, autoquerying is only possible for this type of dictionary.

You can make a dictionary eligible for autoquery by selecting the **Autoqueried in Light Browser?** box for that dictionary in the Define Dictionaries window.

For instructions on autoquerying, see "Browsing in the Terminology Data Tree Structure" on page 14-28.

Searching Across Dictionaries and Domains

You can browse for Repository data within one domain (including the Global domain) or across all domains, and from a single dictionary or across all accessible dictionaries.

Available External System Data

Source data searches scan the data in one external system or all integrated systems for the source term that you enter. The HTML Browser keeps external system data secure during these searches. If a record contains the source term for which you searched, but you do not have sufficient privileges to browse that record in the external system, the Browser will not allow you to view the record either.

The Source Data Search feature only appears in the HTML Browser when:

- You log in as a user with one of the following TMS roles: TMS_CLASSIFY_PRIV, TMS_RECLASSIFY_PRIV, TMS_APPROVE_PRIV, or TMS_RESEARCH_PRIV. You may want to grant an external system user the TMS_RESEARCH_PRIV role if you prefer to deny them access to TMS, but grant them access to Source Data Searches in the HTML Browser.

- You connect to a database in which TMS is fully integrated with an external system. Oracle Clinical and Oracle Adverse Event Reporting System are fully integrated automatically, but you can fully integrate with other external systems as well.

Basic and Complex Searches

All advanced search pages in the HTML Browser include **Use Context Search** boxes, which control whether the browser performs a Basic or Context Search. The difference is significant: Basic Searches can use only two of the special character operators, which enable you to introduce fuzzy logic or language properties in your search.

The two allowable special characters in Basic Searches are the wildcards. The single character wildcard (an underscore: _) matches any single character when inserted in a query. The multi-character wildcard (a percent symbol: %) matches one or more characters.

In addition, the syntax for including wildcards is different in Basic and Context Searches. Assume you want to search for all dictionary terms that begin with the word "ankle" and end with the word "swelling." This search would yield three matches: "ankle swelling", "ankle really swelling" and "ankles are swelling."

Basic Search syntax follows standard SQL queries. Inserting a multi-character wildcard (%) directly between the starting and ending strings returns all three matches:

```
ankle%swelling
```

Context Searches treat multi-character wildcards differently. You must insert a space after the wildcard to retrieve the same results:

```
ankle% swelling
```

HTML Browser Map

Figure 14–1, "Map of HTML Browser Screens with Links" shows which TMS HTML Browser screens have links to which other screens.

The letters A and B represent additional links that didn't fit in the diagram. You can go from A to A in the direction of the arrows, and from B to B in the direction of the arrows.

Figure 14–1   Map of HTML Browser Screens with Links
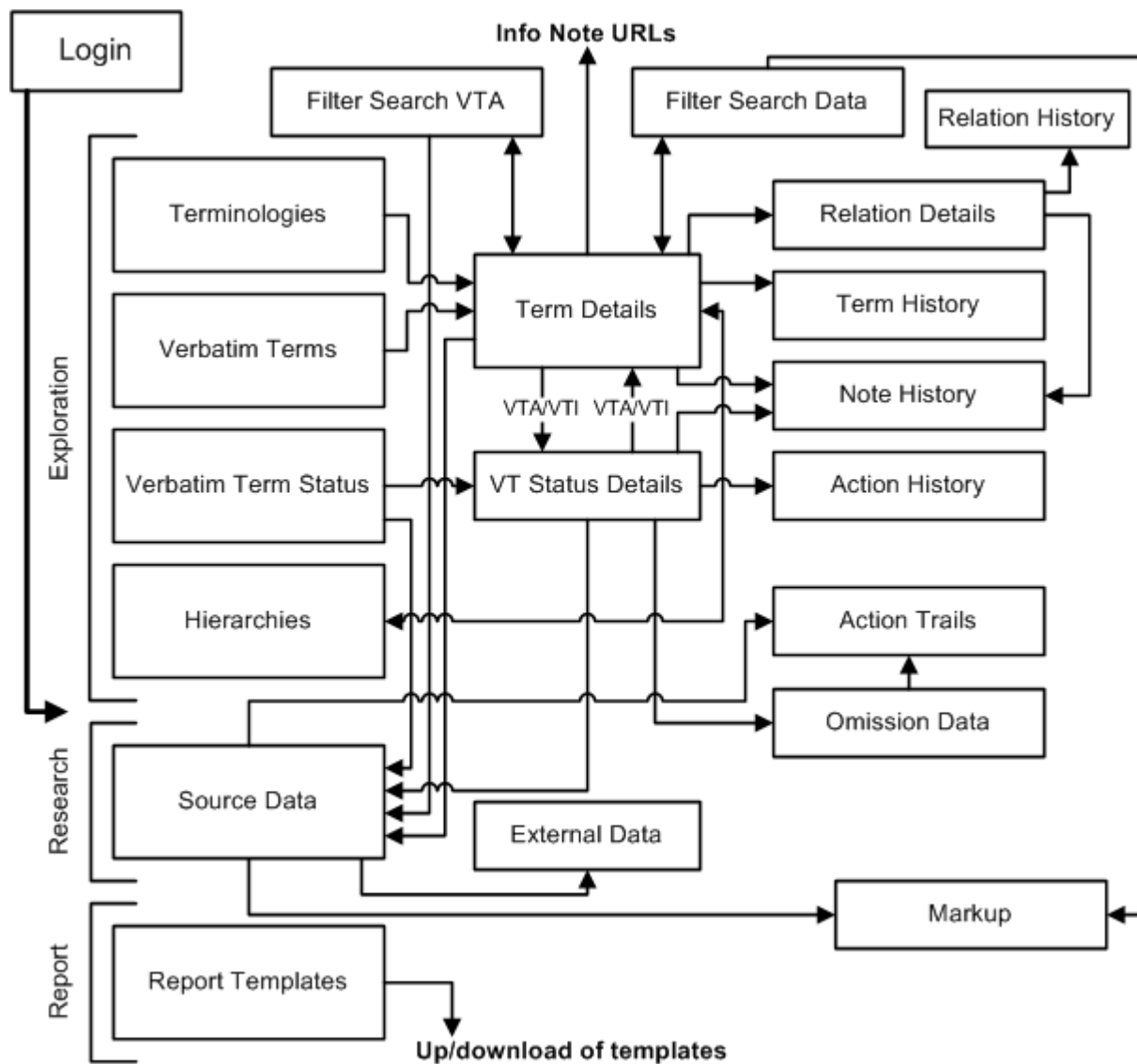
Choosing a Type of Search

The HTML Browser divides all searches into two main categories: Exploration Tab Searches scan dictionary data in the TMS Repository, while Research Tab Searches use external source system data.

Exploration Tab Searches

The **Exploration** tab (Figure 14–2) contains links to four types of search: Terminologies, Verbatim Term Assignment, Verbatim Term Status, and Hierarchies.

Figure 14–2   Exploration Tab Searches
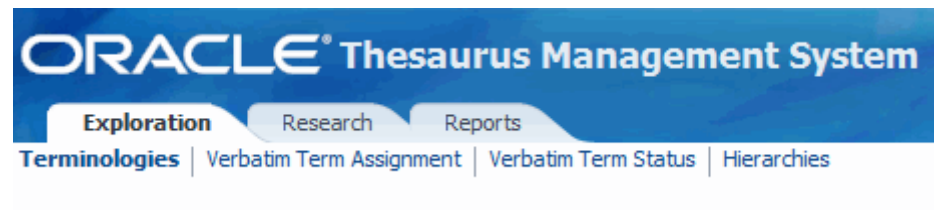
Terminologies

Terminology Searches enable you to search for TMS dictionary data in any accessible dictionary in the database, or across all dictionaries. See "Searching for Terminology Data" on page 14-5.

Verbatim Term Assignment

Verbatim Term Assignment Searches scan only the verbatim term levels of dictionaries, and return information about the verbatim term itself, dictionary terms to which it is classified, and specifics about the VTA or VTI. See "Searching for Verbatim Term Assignments (VTAs and VTIs)" on page 14-16.

Verbatim Term Status

Verbatim Term Status searches scan the verbatim term levels of one dictionary (or all dictionaries in the database, if you choose that option) for your search criteria. For each record returned, the browser lists the verbatim term.

Hierarchies

Hierarchy Browsing let you examine a dictionary's terms from its top level, by using the Terminology Data Tree Structure. See "Browsing a Dictionary Hierarchy" on page 14-26.

Research Tab Searches

The **Research** tab contains a link to access **Source Data** search.

Data Searches enable you to search in any external system that is fully integrated with TMS for Clinical, Adverse Event, or other data. See "Searching for Source Data" on page 14-29.

The **Source Data** tab (under the **Research** tab) appears only if the TMS database you are browsing is fully integrated with an external system. Oracle Clinical and Oracle Adverse Event Reporting System are fully integrated when you install them, and you can install other external systems in full integration as well.

Searching for Terminology Data

Terminology Searches are searches for dictionary terms. You can perform these searches on very broad data sets—across all domains and dictionaries in the database, for example—or within one domain and one dictionary level.

To search for dictionary terms, start with the following steps:

1. Begin a Simple or Advanced Terminology Search

2. Analyze the Results of a Terminology Search

3. Select a Term

To browse a dictionary from the top level (known as autoquerying), perform the steps described in "Browsing in the Terminology Data Tree Structure" on page 14-28.

Like the other windows in the HTML Browser, the Terminology Search page is dynamic: when you choose a setting in one field, it may update the choices available in other fields in the page. For example, if you choose **English** from the **Language** list, the page refreshes and only lists English dictionaries in the **Terminology** list.

Using Special Characters in Searching

Simple Terminology Searches return all the dictionary terms whose Term Names contain the entire text string that you specify. For example, a Simple Search for `ache` returns the terms "backache" and "ache in back" but not the term "aching back."

You can also include special characters as operators in your search string:

- **Underscore** (_). The underscore character represents exactly one unspecified character.

- **Per Cent** (%). The per cent character represents one or more unspecified characters.

- **Dollar Sign** ($). The dollar sign prompts TMS to do a *language stemming* search to find terms with the same linguistic root.

- **Question Mark** (?). The question mark prompts TMS to do a *fuzzy logic* search to find records that nearly match your query parameters.

- **Exclamation Point** (!). The exclamation point prompts TMS to do a *soundex* search to find words that sound similar. You can use soundex searches to find words that are spelled differently but sound the same.

- **Curly Brackets** ({}) serve as escape sequences to enable you to include special characters in your query that would normally be handled as logical operators. Searching for `{10%}` retrieves records with that search string, and the browser does not use the per cent sign as an operator.

Perform a Simple Terminology Search

Simple Searches allow you to search for the term's name within one dictionary, one domain, or across all dictionaries or domains. You must perform an advanced search if:

- You want to search for Repository data using details other than the term's name.

- You want to perform a more complex search than a single text string.

- You want to search for terms within specific dictionary levels.

If your Terminology Search satisfies any of these advanced criteria, proceed to "Perform an Advanced Terminology Search" on page 14-7.

> **Note:** Simple Search finds both VTAs and VTIs that satisfy the criteria. To query for just one or the other, use Advanced Search and specify the Type as a criterion.

To perform a simple terminology search, do the following:

1. If you are not in the Terminology Search page, click the **Terminology** link in the **Exploration** tab.

2. If the **Language** list appears, choose the language for the dictionary (or dictionaries) you want to search. (The **Language** field does not appear if all of the accessible dictionaries in this database are in the same language.)

   When you select a language, the Terminology Search page refreshes, and narrows the list in the **Terminology** list to dictionaries in that language.

3. From the **Terminology** list, select:

   - **All Terminologies** to include all accessible dictionaries of the current language in your search.

   - One of the listed dictionaries, to focus your search within that dictionary's data.

     If you select a filter dictionary such as MedDRA SMQs, the system will search for terms in that filter dictionary. You can then go to the Details window to see base dictionary terms to which the filter term has a named relation; see "Searching for Terms Using Filter Dictionaries" on page 14-11.

   Your dictionary choice restricts the values in the **Domain** list to the subset of domains that contain the selected dictionary.

4. From the **Domain** list, choose a domain in which to focus your search, or **All Domains** to search in every domain that has this dictionary assigned to it.

5. If a Custom Layout exists for the selected dictionary, the browser includes the **Layout** list. Choose either the Default Layout or a Custom Layout.

   Layouts determine which data columns are displayed for each record in the search results. Custom Layouts are dictionary-specific, and allow you to specify a different subset of data columns. See "Defining HTML Layouts" on page 7-2 for more about Custom Layout creation.

6. Click **Search**.

The Terminology Search window displays the terms that match your search criteria in the lower part of the window. You can now choose a dictionary term to investigate. See "Analyze the Results of a Terminology Search" on page 14-9.

Perform an Advanced Terminology Search

Advanced Terminology Searches provide more flexibility than Simple Searches, because they enable you to:

- Use context searches to broaden your query when exact naming matches are not sufficient.

- Focus your search to one or more dictionary levels.

- Search for dictionary data using criteria other than the Term Name. These include the Code, Alternate Code, Dictionary Content ID, the term's Type and Subtype, Approval Status, and the User who introduced the term to the database.

To start an Advanced Terminology Search, open the Terminology Search page, then click the **Advanced Search** button. The HTML Browser opens the Terminologies: Advanced Search page, which is split into three sections: Terminology, Term, and Direction.

The process of creating an Advanced Terminology Search includes the following steps:

1. Determine the Terminology Data Set

2. Choose Term Query Criteria

3. Include Related Dictionary Data in the Search

Step 1: Determine the Terminology Data Set

The **Terminology** section of the Advanced Search page lets you choose which Repository data you want to include in the search. Figure 14–3 shows the fields in this section, with their default choices listed.

Figure 14–3   Terminology Section of the Advanced Terminology Search Page

You can choose the **Language** for the candidate dictionaries, and the **Dictionary**, **Domain**, **Dictionary Levels**, and **Data Currency Status** for the search.

To determine the Terminology Data Set:

1. If the **Language** list appears, choose the language for the dictionary (or dictionaries) you want to search. (The **Language** field does not appear on this page if all of the accessible dictionaries in this database are in the same language.)

When you select a language, the Terminology Search page refreshes, and narrows the list in the **Terminology** to dictionaries in that language.

2. From the **Terminology** list, select:

- **All Terminologies** to include all accessible dictionaries of the current language in your search.

- The name of the dictionary to restrict your search to a single dictionary.

  If you select a filter dictionary such as MedDRA SMQs, the system will search for terms in that filter dictionary. You can then go to the Details window to see base dictionary terms to which the filter term has a named relation; see "Searching for Terms Using Filter Dictionaries" on page 14-11.

Because choosing a dictionary changes the available information for your search, the HTML Browser refreshes and updates the following fields:

- The **Domain** list will only contain domains that contain the selected dictionary.

- The **Levels** field will populate with the level names in the selected dictionary.

- If you choose a virtual dictionary, the browser hides the **Scope** field. This change occurs because virtual dictionaries by definition show a base dictionary at a certain point in time; the browser displays this **cut-off date** next to the **Terminology** field when you choose the dictionary.

- If a Custom Layout exists for the selected dictionary, the browser adds the **Layout** list to the page.

3. From the **Domain** list, choose a domain in which to search, the **Global Domain**, or **All Domains** to search across every domain.

4. Choose a data scope for this search. You can include only **Current** data, **All** data (both current and expired), or **Date**.

   When you choose **Date**, the browser adds a field that enables you to enter a cut-off date. The Terminology Search will display terms and relations that existed on the date you specify.

   You can also choose a cut-off date graphically, by clicking the **Calendar** button and navigating to a date.

5. If you selected one dictionary for this search, you can restrict your search to one or more dictionary levels. Shift-click to highlight contiguous levels, or Ctrl-click for non-continuous levels. To deselect a selected level, Ctrl-click on it.

6. Choose either the Default Layout or a Custom Layout.

   Layouts determine which data columns are displayed for each record in the search results. Custom Layouts are dictionary-specific, and allow you to specify a different subset of data columns. See "Defining HTML Layouts" on page 7-2 for more about Custom Layout creation.

7. Click **Search** to execute the search, or proceed to "Step 2: Choose Term Query Criteria" to further define your Advanced Terminology Search.

Step 2: Choose Term Query Criteria

The fields in the **Term** section enable you to query for almost any column in a Term Record. The Tip text under the **Term** field lists several context operators you may want to include in your search; select the **Use Context Search?** box as well to perform a context search. Select a **Term Type**: VTA or VTI (for nonunique dictionary terms).

Step 3: Include Related Dictionary Data in the Search

You can also expand Terminology Searches so that, in addition to the dictionary term that matches your query, the search also includes all related dictionary terms above or below the matching term in the dictionary hierarchy.

The **Direction** field enables you to use the dictionary hierarchy to include related dictionary terms. Choose **Up** to include parent terms and other terms upward in the dictionary hierarchy; choose **Down** to include child terms and other terms downward in the dictionary hierarchy; or choose **Up/Down** to include terms in both directions.

Analyze the Results of a Terminology Search

When you complete either type of Terminology Search, Simple or Advanced, the browser displays the matches in the bottom of the window.

This section includes:

- Presentation of Results in the Terminology Search Window on page 14-10
- Navigating Through Results on page 14-10
- Refining Your Search on page 14-10

Presentation of Results in the Terminology Search Window

Figure 14–4 shows some of the results of a Simple Search. This search for the string "headache" uses the Default Layout, and includes all dictionaries and domains.

Figure 14–4   Terminology Search Results Using Default Layout

Results

Term	Approved	Terminology	Level	Code	Domain	Alt.code	Id
Headache	Yes	MedDRA Primary Path	Preferred Terms	10019211	GLOBAL	6.0	2207242
Headache	Yes	SNOMED	Descriptions	41990019	GLOBAL	PT	1984101
Headache	Yes	ICD-CM	Disease or Procedure Level 2	784.0	GLOBAL	69	167564
HEADACHE	Yes	OPA Dictionary	Verbatim Term		OPADOM		7415898
Headache	Yes	OPA Dictionary	Symptom Level		GLOBAL		7415888
Headache	Yes	WHO-ART	Preferred Terms	W0109	GLOBAL	337	56812
HEADACHE	Yes	COSTART	Preferred Terms		GLOBAL	224	187142

Two variables control the information that the HTML Browser returns in the Results screen:

Columns   The columns that the Terminology Search page displays for each record are determined by which layout you select when you enter your search parameters. The search shown in Figure 14–4 used the Default Layout, which returns each term's **Term Name**, **Approval Status**, **Dictionary Level**, **Code Number**, **Alternate Code Number**, and **TMS Internal ID Number**. If you choose **All Terminologies** or **All Domains** in your search, the page includes columns for each record's dictionary or domain as well.

Rows   Each search window in the HTML Browser returns no more than 25 rows per page. You cannot change this row-per-page setting, but you can change the maximum number of rows that a query can find by updating the OPA setting OPA_UIX_MAX_ROWS. See "Customizing Defaults in TMS Windows Using TMS Settings" on page 3-17 for more information on using TMS Settings to control some default values in the HTML Browser.

Navigating Through Results

You can navigate through the results of any kind of search using the scrollbars at the right of the screen. They provide continuous scrolling through the entire set of results retrieved by the Search.

Sorting the columns of the Results page can also help you navigate to a term more quickly. You can sort on any of the columns that you include in your layout.

Refining Your Search

If your search did not yield any of the records you wanted, you may want to edit the search criteria at the top of the page, then execute this new search. Note that when you update any of these lists, the change causes the Web browser to refresh the page, but this refresh does not requery the database for new results. When you complete your changes to the search criteria, you must click the **Search** button to execute the search.

Select a Term

When you find the term you want to use from the records returned in your Repository search, you can view more detailed information about that term or browse its place in the dictionary hierarchy.

Example 14–1 Selecting a Term

When you use a Default Layout, each record that a Terminology Search returns is presented like the example below. This record shows the Lowest Level Term "Back ache," from the MedDRA Primary Path Dictionary in the Global Domain.

Term	Approved	Terminology	Level	Code	Domain	Alt.code	Id
Back ache	Yes	MedDRA Primary Path	Lowest Level Terms	10003978	GLOBAL	2.1	1555373

Click the hyperlinked Term Name to access the Term Details window, which displays three types of data about a dictionary term: all of the detailed information in the database about that Term Record, its derived path in its dictionary, and any related terms. See "Using the Term Details Window" on page 14-32.

Searching for Terms Using Filter Dictionaries

If your company uses filter dictionaries such as MedDRA SMQs, you can use the filter dictionary to find related dictionary terms, VTAs, and source terms. In the Term Details window you can see the relations between filter terms and base dictionary terms.

To begin, do the following:

1. Go to the **Terminologies** window in the **Exploration** tab.

2. Enter a query for a filter dictionary term; specify a filter dictionary in the **Dictionary** field or specify a filter dictionary term in the **Term** field. Or do an **Advanced Search**; see "Perform an Advanced Terminology Search" on page 14-7.

3. Click **Search**. The system displays terms in the **Results** section that satisfy the search criteria.

4. Click the term in the **Results** section. The Term Details window opens. Expand the **Term Details** node (**>**) to see the following filter dictionary-related information:

   - **Informative Notes**. If there are Informative Notes associated with the term you see them at the bottom of the Term Details section. You can see the text of all Informative Notes associated with the term. For Informative Notes of type Algorithm you see the actual algorithm.

     You can click on the icon for each Informative Note to see its history. If changes have been made to the note over time, you see each version of the note.

   - **SMQ category**. The term **Status** field displays the SMQ category for the term. The category is used in search algorithms.

   The **Related Terms** section displays all terms to which the filter dictionary term has a named relation. The terms' dictionary and dictionary level are displayed in the **Level** column.

   See "Using the Term Details Window" on page 14-32 for more information.

You can link to other windows from the Term Details Window:

- **Related VTAs**. Click the **Verbatim Term Assignment** link and enter a query to see VTAs related to the term that meet your search criteria; see "Performing a Simple VTA Filter Search" on page 14-12 and "Performing an Advanced VTA Filter Search" on page 14-13.

- **Related Source Terms**. Click the **Source Data** link and enter a query to see source terms related to the term that meet your search criteria; see "Performing a Simple

Source Data Filter Search" on page 14-14 and "Performing an Advanced Source Data Search" on page 14-15.

- **Hierarchy Search**. Expand the Term Details node (+) and click the Term Hierarchy link; see "Browsing a Dictionary Hierarchy" on page 14-26.

- **Term History**. Expand the Term Details node (+) and click the Term History link; see "Examining a Term's Details and History" on page 14-32.

- **Relation Details**. Click on a link in the Relations column to see the details of that relation; see "Using the Relation Details Window" on page 14-37. The Status of the relation indicates the SMQ Category of the relation, which is used in SMQ algorithms.

Performing a Simple VTA Filter Search

Use this window to search for VTAs related to Filter Dictionary terms (SMQ terms).

To Reach This Window   After you have queried for a filter dictionary term in the Terminologies window (see the previous section), click a filter dictionary term, then click **Verbatim Term Assignment**.

Searches in this window return VTAs that satisfy the search criteria you enter here and are related to the filter dictionary term you selected in the Term Details window. To perform a search do the following:

1. Select a **Domain** in which to search, or select **All Domains**. **Global** is also an option.

2. Select a **Direction** to use the dictionary hierarchy to include related verbatim term assignments (VTAs) in the search results:

   - **None**. No hierarchical search is conducted; the search retrieves only VTAs with a direct relation to the selected term.

   - **Up**. The search retrieves parent terms and other terms upward in the dictionary hierarchy; in the case of SMQs, this option does not make sense because there will never be VTAs in levels above SMQ terms.

   - **Down**. The search retrieves child terms and other terms downward in the dictionary hierarchy.

   - **Up/Down**. Retrieves related terms both above and below the selected term in the hierarchy; in the case of SMQs, this option does not make sense because there will never be VTAs in levels above SMQ terms.

   For example, if you select the SMQ term *Hepatic disorders (SMQ)* and search with Direction set to None the search retrieves no records because *Hepatic disorders (SMQ)* has no direct relations to any MedDRA terms. However, if you search with Direction set to Down it will retrieve many terms because its children do have relations with MedDRA terms, many of which may be VTAs.

3. Select a **Filter Relationship**. For MedDRA SMQs the choices are as your company defined them: Broad Scope, Narrow Scope, or All named relations, or the equivalent.

4. Click **Search**. The system displays the retrieved VTAs in the **Results** section; see "Viewing VTA Filter Search Results" on page 14-13.

Performing an Advanced VTA Filter Search

Advanced Search allows you to use additional search criteria. Specify a value for one or more of the following criteria:

- Select a **Domain** in which to search, or select **All Domains**. **Global** is also an option.

- Select a **Direction** to use the dictionary hierarchy to include related verbatim term assignments (VTAs) in the search results:

  - **None**. No hierarchical search is conducted; the search retrieves only VTAs with a direct relation to the selected term.

  - **Up**. The search retrieves parent terms and other terms upward in the dictionary hierarchy; in the case of SMQs, this option does not make sense because there will never be VTAs in levels above SMQ terms.

  - **Down**. The search retrieves child terms and other terms downward in the dictionary hierarchy.

  - **Up/Down**. Retrieves related terms both above and below the selected term in the hierarchy; in the case of SMQs, this option does not make sense because there will never be VTAs in levels above SMQ terms.

  For example, if you select the SMQ term *Hepatic disorders (SMQ)* and search with Direction set to None the search retrieves no records because *Hepatic disorders (SMQ)* has no direct relations to any MedDRA terms. However, if you search with Direction set to Down it will retrieve many terms because its children do have relations with MedDRA terms, many of which may be VTAs.

- Select a **Filter Relationship**. For MedDRA SMQs the choices are as your company defined them: Broad Scope, Narrow Scope, or All named relations, or the equivalent.

- Select a MedDRA SMQ or other filter dictionary **Status**.

- Select an **Approval** status for the VTAs: Approved, Not Approved, or All.

- Select a **VTA Subtype**: Misspelled, Accepted, or All.

- Click **Search**. The system displays the retrieved VTAs in the **Results** section; see "Viewing VTA Filter Search Results" on page 14-13.

Viewing VTA Filter Search Results

For each term retrieved by either a simple or advanced VTA filter search, the system displays the following information:

- **Verbatim Term** name.

- Link to **Source Data**. See "Performing a Simple Source Data Filter Search" on page 14-14 and "Performing an Advanced Source Data Search" on page 14-15.

- **Approved**. If Y, the VTA is approved, if N, it is not approved.

- **Dictionary Term**. The dictionary term to which the verbatim term is assigned. Use this link to see dictionary term details; see "Using the Term Details Window" on page 14-32.

- **Code**. The unique ID of the dictionary term to which the verbatim term is assigned.

- **Domain**. The VTA's domain, if the VTA is specific to a domain, or GLOBAL if it is global.

> **Note:** Virtual dictionaries cannot have global VTAs.

- **Terminology**. The dictionary that contains the VTA's dictionary term.

- **Level**. The dictionary level of the VTA's dictionary term.

Performing a Simple Source Data Filter Search

Use this window to search for source terms related to Filter Dictionary terms (SMQ terms). After you have queried for a filter dictionary term (see the previous section), click **Source Data** and enter a query here. The search returns source terms that satisfy the search criteria you enter here and are related to the term you selected in the Term Details window.

To view source data for VTAs that are related to a filter dictionary term, do the following:

1. Select an **Application**; the external source data system that collected the source terms you want to view.

2. Select a **Group**. Groups control which external system columns you return for each external system data record. In Oracle Clinical Data Searches, for example, you can view the Project, Study, and Patient for each record. See "Setting Up External System Drill-down Queries" on page 5-36 for instructions about how to define groups.

3. Select a **Domain** or All Domains.

4. Select a **Direction** to use the dictionary hierarchy to include related verbatim term assignments (VTAs) in the search results:

   - **None**. No hierarchical search is conducted; the search retrieves only VTAs with a direct relation to the selected term.

   - **Up**. The search retrieves parent terms and other terms upward in the dictionary hierarchy; in the case of SMQs, this option does not make sense because there will never be VTAs in levels above SMQ terms.

   - **Down**. The search retrieves child terms and other terms downward in the dictionary hierarchy.

   - **Up/Down**. Retrieves related terms both above and below the selected term in the hierarchy; in the case of SMQs, this option does not make sense because there will never be VTAs in levels above SMQ terms.

   For example, if you select the SMQ term *Hepatic disorders (SMQ)* and search with Direction set to None the search retrieves no records because *Hepatic disorders (SMQ)* has no direct relations to any MedDRA terms. However, if you search with Direction set to Down it will retrieve many terms because its children do have relations with MedDRA terms, many of which may be VTAs.

5. Select a **Filter Relationship**. For MedDRA SMQs the choices are as your company defined them: Broad Scope, Narrow Scope, or All named relations; or the equivalent.

   Alternatively, if the filter dictionary term is associated with an algorithm Informative Note, you can conduct a search using that algorithm by selecting: **@ SMQ Algorithm**.

6. Click **Search**. The **VTA Source Data** window appears; see "Viewing Source Data for a Verbatim Term Record" on page 14-20.

Performing an Advanced Source Data Search

Use this window to search for source terms related to Filter Dictionary terms (SMQ terms). After you have queried for a filter dictionary term (see the previous section), click **Source Data**, then click **Advanced Search** and enter a query here. The search returns source terms that satisfy the search criteria you enter here and are related to the term you selected in the Term Details window.

Advanced Search allows you to use additional search criteria.

1. Select an **Application**; the external source data system that collected the source terms you want to view.

2. Select a **Group**. Groups control which external system columns you return for each external system data record. In Oracle Clinical Data Searches, for example, you can view the Project, Study, and Patient for each record. See "Setting Up External System Drill-down Queries" on page 5-36 for instructions about how to define groups.

3. Select a **Domain** or All Domains.

4. Select a **Direction** to use the dictionary hierarchy to include related verbatim term assignments (VTAs) in the search results:

    ■   **None**. No hierarchical search is conducted; the search retrieves only VTAs with a direct relation to the selected term.

    ■   **Up**. The search retrieves parent terms and other terms upward in the dictionary hierarchy; in the case of SMQs, this option does not make sense because there will never be VTAs in levels above SMQ terms.

    ■   **Down**. The search retrieves child terms and other terms downward in the dictionary hierarchy.

    ■   **Up/Down**. Retrieves related terms both above and below the selected term in the hierarchy; in the case of SMQs, this option does not make sense because there will never be VTAs in levels above SMQ terms.

    For example, if you select the SMQ term *Hepatic disorders (SMQ)* and search with Direction set to None the search retrieves no records because *Hepatic disorders (SMQ)* has no direct relations to any MedDRA terms. However, if you search with Direction set to Down it will retrieve many terms because its children do have relations with MedDRA terms, many of which may be VTAs.

5. Select a **Filter Relationship**. For MedDRA SMQs the choices are as your company defined them: Broad Scope, Narrow Scope, or All named relations, or the equivalent.

6. Select a **Status**. TMS stores the MedDRA SMQ term category in this field.

7. Select an **Approval** status for the VTAs: Approved, Not Approved, or All.

8. Select a **VTA Subtype**: Misspelled, Accepted, or All.

9. Enter values in the **Source Data** section. TMS displays the external system source data fields defined for the application you selected. In each field you can enter a specific value to narrow the search.

    For example, if you selected Oracle Clinical as the application, you can enter a specific study to retrieve source terms collected in that study only.

10. Click **Search**. The **VTA Source Data** window appears; see "Viewing Source Data for a Verbatim Term Record" on page 14-20.

Searching for Verbatim Term Assignments (VTAs and VTIs)

Verbatim Term Assignment (VTA) Searches scan only the verbatim term levels of dictionaries, and return information about the verbatim term itself, dictionary terms to which it is classified, and specifics about the VTA.

To search for verbatim terms:

1. In the Exploration tab, select Verbatim Term Assignment, then choose a simple or advanced search. See Perform a Simple Terminology Search or Perform an Advanced Terminology Search.

2. Analyze the Results of a Verbatim Term Search

3. Select a Verbatim Term Record

After you choose the term you want to examine, you can perform any of the following tasks with that verbatim term record:

- Browsing a Dictionary Hierarchy on page 14-26

- Examining a Term's Details and History on page 14-32

- Examining a Relation's Details and History on page 14-36

Perform a Simple Verbatim Term Search

Simple Searches return all the verbatim terms whose Term Names contain the entire text string that you specify. For example, a Simple Search for ache returns the terms "backache" and "ache in back" but not the term "aching back." You can also include in your search string any of the text operators available in the *inter*Media text option; see "Using Special Characters in Searching" on page 14-6.

You can perform Simple Searches for data within one dictionary and domain, or expand the search across all domains and accessible dictionaries in the Repository. If you need to search for verbatim term data using details other than the term's name, or you want to perform more complex searches than just one text string, follow the instructions in "Analyze the Results of a Verbatim Term Search" on page 14-19.

Like the other windows in the HTML Browser, the Terminology Search page is dynamic: when you choose a setting in one field, it may update the choices available in other fields in the page. For example, if you choose **English** from the **Language** list, the page refreshes and only lists English dictionaries in the **Terminology** list.

To perform a Simple Verbatim Term Search:

1. If you are not in the Verbatim Terms Search page, click the **Verbatim Term Assignment** tab.

2. From the **Language** list, choose the language for the dictionary (or dictionaries) you want to search. When you select a language, the Search page narrows the list in the **Terminology** list to dictionaries in that language.

3. From the **Terminology** list, select:

    - **All Terminologies** to include all accessible dictionaries of the current language in your search.

    - The name of the dictionary to restrict your search to a single dictionary.

    When you choose a specific dictionary, the Search page may change in several ways. Your dictionary choice can also effect which domains are listed in the **Domain** field. The Terminology Search restricts the values in this list to those domains that contain the selected dictionary.

4. From the **Domain** list, choose a single domain to focus your search, or **All Domains** to search across every domain that includes the selected dictionary or dictionaries.

5. Click **Search**.

The Verbatim Term Search window displays the terms that match your search criteria in the lower part of the window. See "Analyze the Results of a Verbatim Term Search" on page 14-19 for the next step.

Perform an Advanced Verbatim Term Search

Advanced Verbatim Term searches allow you more flexibility and power than Simple Searches, because they allow you to search for:

- Current or retired verbatim terms

- All verbatim terms that classify to a particular dictionary term

- Verbatim terms that match external system criteria

To start an Advanced Verbatim Term Search, open the Verbatim Term Search page, then click the **Advanced Search** button. The Advanced Search page contains the following sections:

- Terminology on page 14-17

- Verbatim Term on page 14-17

- Relations on page 14-18

- External System on page 14-19

Terminology

The Terminology selections restrict the Candidate Data Set for your Verbatim Term Search according to dictionary (labelled "Terminology"), domain, and data currency.

Verbatim Term

The Verbatim Term selections enable you to focus your search using the following fields relating to verbatim terms. The following selections can refine your Verbatim Term Search:

Use Context Search. If selected, the HTML Browser will use the Context Server Index for your search. See "Querying in Windows" on page 2-9 for an overview of the

differences between simple and context searches. This page includes a tip with the common context search operators, such as Fuzzy and Soundex.

Use VT as DT. If selected, the search returns only direct matches, where the verbatim term (VT) and dictionary term (DT) match exactly.

Only Inconsistent. If selected, the browser will only return verbatim terms that are inconsistently classified in different domains.

Search ID. If a search object created this VTA or VTI, you can search for it using the values in this list.

Approval. The approval status of the VTA. Choose **Approved**, **Not Approved**, or **All**.

User. The user who classified the VTA or VTI.

Database. The external system database from which the source term originates.

Relations

Settings in the **Relations** section enable you to include a wider range of verbatim terms in your search, based on dictionary structure.

Searching for verbatim terms under a Start Level enables you to include all the verbatim terms that derive up to a particular verbatim term. If you search for "nerv%" in MedDRA and specify System Order Class (SOC) as the Start Level, the browser returns all verbatim terms that derive up to the "Nervous system disorders." If you include a Reverse at Level setting as well, the system derives the terms that share the common parent at the level specified. You can also focus the verbatim terms returned in this search to primary relations only by selecting the **Only Primary Rels** box.

Figure 14–5   Start Level Example
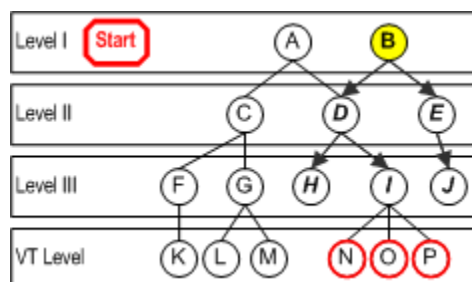
The sample dictionary in Figure 14–5 illustrates a Verbatim Term Search with a Start Level. If a query uses Level I as the start level, and the query criteria match Term B only, the resulting verbatim Term Records would be the VT records whose primary derivable path goes up to Term B. The End Level for all VT searches is always the VT level. In this example, the search would return the VT Level terms N, O, and P.

Figure 14–6   Reverse at Level Example
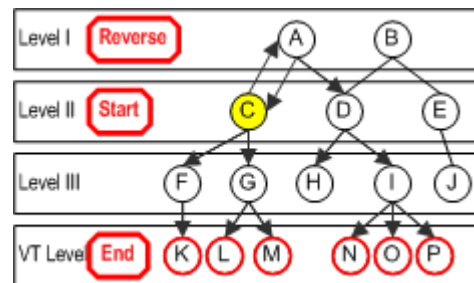
For the same sample dictionary, Figure 14–6 illustrates how the browser conducts a Reverse at Level search for verbatim term data. Assume a Verbatim Term Search uses Level II as the Start Level and Level I as the Reverse Level. If Term C is the only match for the query criteria, the browser derives Term C's parent in the Reverse Level (Term A), then returns all of the verbatim term level records whose Level I derivable parent is Term A. Effectively, the search in this example finds all of the verbatim terms that derive up to term C and any of term C's sibling terms (in this case, it is only sibling is term D).

Selecting the **Only Primary Rels** box excludes all secondary relations from the Verbatim Term Search. Viewing just primary paths information can clarify the classification information for you, but could focus the search more than you need.

External System

You can focus a Verbatim Term Search according to which external system data is included in the verbatim term. When you choose an external system from the list (Oracle Clinical, in this case), the page refreshes and includes that external system's columns.

Analyze the Results of a Verbatim Term Search

When you complete either type of Repository Search, Simple or Advanced, the browser displays the matches at the bottom of the window.

This section includes:

- Presentation of Results in the Verbatim Term Search Window on page 14-19

- Refining Your Search on page 14-20

Presentation of Results in the Verbatim Term Search Window

Figure 14–7 shows the results of a Simple Search of the MedDRA Primary Path Dictionary across all domains for the string "pain".

Figure 14–7   Results in the Verbatim Term Search Window

Verbatim Term	Source	Approved	Dictionary Term	Code	Domain	Terminology	Level
HEADACHE	DATA 🔍	Yes	Headache		OPADOM	OPA Dictiona...	Symptom Level

By default, each search window in the HTML Browser returns no more than 25 rows per page. You cannot change this rows-per-page setting, but you can change the maximum number of records the HTML Browser can retrieve; to do so, update the variable OPA_UIX_MAX_ROWS in the TMS Settings window. See "Customizing Defaults in TMS Windows Using TMS Settings" on page 3-17 for more information on using TMS Settings to control some default values in the HTML Browser.

You cannot use an HTML Layout to change the set of columns displayed for Verbatim Term Searches. All Verbatim Term Searches use the columns displayed in Figure 14–7.

Refining Your Search

If your search did not yield any of the records you wanted, or produced too many matches, you can edit your search criteria to create a more inclusive or specific search. To refine your search, change any of your choices for the dictionary or domain, or enter a new search string in the **Search** field, then click the **Search** button.

> **Note:** When you update any of these lists, the change causes the Web browser to refresh the page, but this refresh does not requery the database for new results. When you complete your changes to the search criteria, you must click the **Search** button to execute the search.

Select a Verbatim Term Record

When you find the verbatim term you want to examine from the records returned in your Verbatim Term Search, you can either view more detailed information about that term or browse its derivable path or paths in the dictionary hierarchy.

Example 14–2   Selecting a Verbatim Term

Verbatim Term Searches return records in the format below. This example shows the verbatim term "left ear pain" classified to the dictionary term "Ear ache." This classification applies in the global domain for the MedDRA Primary Path Dictionary.

Verbatim Term	Source	Approved	Dictionary Term	Code	Domain	Terminology	Level
SORE THROAT	DATA 🔍	Yes	Pancreatitis due to biliary obstruction	10057010	MEDDRADOM	MedDRA Primary Path	Lowest Level Terms

Clicking either hyperlinked **Term Name** (the verbatim term or the dictionary term to which it is classified) launches the Term Details window for that term. The Term Details window displays three types of data about a term: all of the detailed information in the database about that Term Record, its derived path in its dictionary, and any related terms. See "Using the Term Details Window" on page 14-32.

Viewing Source Data for a Verbatim Term Record

Each verbatim Term Record includes a **Source Data** icon under the Source heading. Clicking this icon launches the Classified Source Data window, which you can use to view the original source data from a particular external system. Reading through the

external system source data can provide information about the source term, such as the clinical trial project and study from which it arose.

To view a verbatim term's external system source data:

1. From the Verbatim Term Search window, click the **Source Data** icon for the verbatim Term Record you want to investigate.

   The HTML Browser loads the Classified Source Data page, with two fields displayed: **Application** and **Database**.

2. From the **Application** list, choose the external system you want to investigate.

3. Choose the **Database** from which you want to retrieve external system data.

4. Click **Search**. The browser loads external system records that contain the term you selected, and to which you have access in the external system.

Markup	Project	Study	Investigator	Visit	Patient	Document Number	DCM	Discrepancy
📄	TMSPROJII7	TMSSTDYII7	TMS_INVII7	BASELINE 1	1117	TMS7II7	OPADCMII7	

Application: Oracle Clinical   Database: [   ]   Search

To view an external system record, click the **Markup** icon to the left of the Project column. The Source Data page for that record loads in the browser; see "Viewing a Source Data Record with Markup" on page 14-21 for more information.

Some external system information is displayed as hyperlinks. These details have been generated by external system views or functions, and the hyperlinks launch a page that provides more information about the external system. See "Viewing Data from an External Drill-down Function or View" on page 14-22.

Viewing a Source Data Record with Markup

Markup reformats the presentation of a source data record so that the selected source term is hyperlinked, and appears prominently in the page (Figure 14–8).

Figure 14–8 Sample Source Data Record

Viewing Data from an External Drill-down Function or View

Drill-down information enables you to see more about the program, study, or project generating the data record that contains your source term. Clicking any of these hyperlinks launches either the Source Term Data view or Data Function page.

Viewing VTA and Omission Status Information

The Verbatim Term Status window allows you to search for verbatim terms, including omissions and VTAs, by their status. Using Advanced Search you can also search for particular verbatim term occurrences by external system values and by assignee. See:

- Performing a Simple Verbatim Term Status Search on page 14-22
- Performing an Advanced Verbatim Term Status Search on page 14-23

From the search results, through one or more links, you can see additional information, including the following:

- Viewing Verbatim Term Status History Details on page 14-24
- Viewing a Term's Action History on page 14-24
- Viewing the Action History of an Omission Occurrence on page 14-25

Performing a Simple Verbatim Term Status Search

To perform a Simple Verbatim Term Status search:

1. If you are not in the Verbatim Terms Search page, click the **Verbatim Term Status** tab.

2. If there are multiple languages available, choose the language for the dictionary (or dictionaries) you want to search from the **Language** list. When you select a language, the Search page narrows the selection in the **Terminology** list to dictionaries in that language.

3. From the **Terminology** list, select:

   - **All Terminologies** to include all accessible dictionaries of the current language in your search.

   - The name of the dictionary to restrict your search to a single dictionary.

   If you choose a virtual dictionary, the system displays its base dictionary name and its cut-off date and time.

   When you choose a specific dictionary, the Search page may change in several ways. Your dictionary choice can also effect which domains are listed in the **Domain** field. The Terminology Search restricts the values in this list to those domains that contain the selected dictionary.

4. From the **Domain** list, choose a single domain to focus your search, or **All Domains** to search across every domain that includes the selected dictionary or dictionaries.

5. From the Status list, choose a single status to focus your search or **Any Status**.

6. If you wish, enter a specific Term for which to search, or use special characters; see "Searching for Verbatim Term Assignments (VTAs and VTIs)" on page 14-16 for information on using special characters.

7. Click **Search**.

In the **Results** section of the window the system displays the terms that match your search criteria, with each term's current status and substatus, domain name, the user name of the person or process that performed the action that led to the term's current status, and the user to whom the term is assigned, if any.

The verbatim term name is a link. Click it to view the status history of the term; see "Viewing Verbatim Term Status History Details" on page 14-24.

Performing an Advanced Verbatim Term Status Search

Advanced verbatim term status searches allow you to specify:

- Substatuses as well as statuses

- The person to whom a term is currently assigned

- External system criteria

To perform an advanced verbatim term status search, enter values in the following fields to serve as search criteria and click **Search**:

Language If multiple languages are available, choose one from the list.

Terminology Choose a terminology from the list.

Domain Choose a domain from the list (choices are limited by the terminology you chose.

Status Leave at the default value of Any Status or choose a specific status as a search criterion; see "Term Statuses" on page 10-19.

Substatus Leave at the default value of Any Substatus or choose a specific substatus as a search criterion; see "Term Statuses" on page 10-19.

Term Do one of the following:

- Leave blank to retrieve all terms that fit the other search criteria.

- Enter the name of a specific term to retrieve that term only.

- Use special characters with a character string; see "Perform a Simple Terminology Search" on page 14-6 for information.

Created By   leave blank or enter the user name of the person or process that created the term or terms you are searching for.

Assignee   Leave blank or enter the user name of the person to which the term is currently assigned.

External System   From the list, choose the external system where the term was collected. When you choose an external system, TMS displays additional fields appropriate for that system. For example, you can enter values in any of these fields to further narrow the search.

Viewing Verbatim Term Status History Details

To reach this window, click the term name in the Verbatim Term Status window; see "Performing a Simple Verbatim Term Status Search" on page 14-22.

This window displays the following information for a single term:

- a complete history of the statuses that have been assigned to a term, with the most recent at the top

- the user to whom the term is currently assigned, if it is assigned

- the complete history of Informative Note that have been assigned to the term, if any, with the most recent at the top

There are several links to additional information from this window:

- **Term Details**. Click the verbatim term name to see the Term Details window; see "Using the Term Details Window" on page 14-32.

- **Omission Data**. If the term is an omission, click the Omission Data icon to see external system information associated with the verbatim term; see"Browsing a Dictionary Hierarchy" on page 14-26.

- **Source Data**. If the term is not an omission, click the Source Data icon to see external system information associated with the verbatim term; see "Viewing Source Data for a Verbatim Term Record" on page 14-20.

- **Action Details**. If the term has ever had an Action assigned to it, click the Action text for additional details; see "Viewing a Term's Action History" on page 14-24.

- **Dictionary Term Details**. If the term is a VTA (is assigned to a dictionary term), click the dictionary term to see additional information on the dictionary term; see "Using the Term Details Window" on page 14-32.

- **Info Note History**. Click the Informative Note icon to see the history of Informative Notes applied to this term; see "Examining a Record's Informative Notes" on page 14-38.

Viewing a Term's Action History

In this window you can see the complete history of actions applied to a term, including the following information for each Action. This window shows every

change to an Action—including, for example, a change to the Action's text—even if the change does not result in a change in status for the term.

- **Action Text**. The text that appears to the user assigned to the task in TMS, or to the external system, depending on the Action type.

- **Action Type**. Internal, Answerable, or Unanswerable; see "Action Types" on page 7-6 for information.

- **Action Ref Type**. If the Action is of type Internal, it must reference an Action of type Answerable or Unanswerable. This is the Action it references. (If the Action is type Answerable or Unanswerable, this field does not appear.)

- **Creation Time**. The date and time the Action was assigned to the term.

- **Created By**. The user name of the person who assigned the Action.

- **Valid Until**. The expiration date of the Action. If the Action is currently valid and has no set expiration date, Valid Until is set to 15-Aug-3501 00:00:00.

To reach this window, do the following:

1. Search for a verbatim term in the Verbatim Term Status Search window or the Verbatim Term Assignment window.

2. In the Search Results section, click the verbatim term name to see the Verbatim Term Status Details window.

3. If the term has had an Action assigned, click the Action text.

Viewing the Action History of an Omission Occurrence

This window displays the complete Action history associated with a single occurrence of a verbatim term omission (an unclassified source term). Because the information is for a single occurrence, the system can provide information not available in the Action History window, including the omission status and information on which system owns the term. The system displays the following Action-related information:

- **Verbatim Term**. The term collected in the external source data system.

- **Action**. The Action name.

- **Action App. Owner**. The name of the application that currently owns the omission; either TMS or the external system where the term was collected.

- **Action Owner**. The user name of the person who applied the Action to the term.

- **Omission Status**. The status TMS has assigned to this particular occurrence of the omission. This is the status that is sent to the external system.

- **Action Text**. The text that appears to the user assigned to the task in TMS, or to the external system, depending on the Action type.

- **Action Type**. Internal, Answerable, or Unanswerable; see "Action Types" on page 7-6 for information.

- **Creation Time**. The date and time the Action was assigned to the term.

- **Valid Until**. The expiration date of the Action. If the Action is currently valid and has no set expiration date, Valid Until is set to 15-Aug-3501 00:00:00.

To reach this window, do the following:

1. Search for a verbatim term omission in the Verbatim Term Status Search window.

2. In the Search Results section, click the verbatim term name to see the Verbatim Term Status Details window.

3. Click the omissions icon. The Omission Data window appears.

4. Click the icon in the Action Text column, if any.

Viewing External System Information for Omissions

To view the source system information associated with each occurrence of a verbatim term omission, choose the external system from the list and click **Search**. The system displays the term information appropriate for that external system.

To reach this window, do the following:

1. Query a verbatim term in the Verbatim Term Status page.

2. Click on the term name in the results, it should be a record with the status of VTO - Verbatim Term Omission. The system opens the Term Status Details page.

3. Click the omission icon displayed near the Omission Data field.

Browsing a Dictionary Hierarchy

This section provides tips and strategies for browsing through data in the Terminology Data Tree Structure window. You can start a browsing session in the following ways:

- In the **Exploration** tab, click **Hierarchies**, then choose a **Language** and **Dictionary** to explore. This strategy starts a browsing session at the top level of a dictionary.

  You can only browse the top level of a dictionary if that dictionary is eligible for autoquery. For instructions on checking if a dictionary is eligible, see "Autoquerying Dictionaries" on page 14-2.

- Perform a Terminology Search or Verbatim Term Search, choose a term, expand its Details node, then click the **Term Hierarchy** link. The Terminology Data Tree Structure window will open and display the selected term in focus.

This section discusses the following Actions you can take in the Terminology Data Tree Structure window:

- Introduction to Browsing the Dictionary Hierarchy on page 14-26

- Starting a Browsing Session from the Hierarchies Link on page 14-27

- Browsing in the Terminology Data Tree Structure on page 14-28

- Expanding a Branch One Level Using the Arrow Keys on page 14-28

- Expanding a Branch in the Dictionary on page 14-28

- Reversing the Orientation of the Tree Window on page 14-28

Introduction to Browsing the Dictionary Hierarchy

The example in Figure 14–9 shows the Terminology Data Tree Structure interface for selected terms in a strong dictionary, MedDRA Primary Path.

Figure 14–9   Overview of the Structure of the Terminology Data Tree Structure Window

Example 14–3   Explanations of Annotations in Figure 14–9

1. In order to set the focus on a term, click on the row that contains it or use the Ctrl key to select the respective row.

2. The arrow button expands or contracts a single branch of the tree. In this example, "Vascular disorders", "Vascular injuries" and "Vascular injury" are expanded, while "Venous and vena caval injuries" is contracted.

3. The Term Name of each record is hyperlinked to the Term Details page.

4. Clicking the reverse button switches the orientation of the Tree Structure window. If the tree structure is organized to display the child terms of the in-focus term (the one on the highlighted row), you can reverse direction to see the parent terms of any term you select in the window.

There are also three more buttons that you can use, located on the table header:

■   When you have a term highlighted on a row, click this button: to go up one level in the tree structure.

■   To go to the top of the structure, click this button: .

■   When you want to explore a multi-layered structured, you can choose to see the highlighted term as the first row in the table by clicking this button: . If you wish to go back to the previous view, click the blue arrow that will appear next to the term to have an overall view of the structure and choose the desired level of detail:

Starting a Browsing Session from the Hierarchies Link
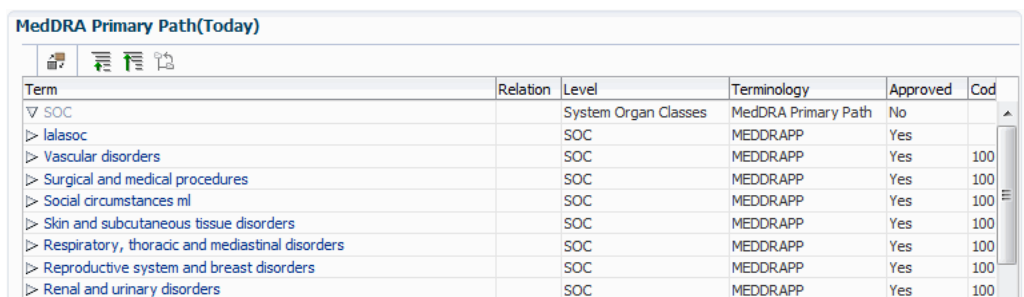
Clicking the **Hierarchies** link on the **Exploration** tab enables you to start browsing a dictionary's data from its top level. To start browsing the top level of a dictionary:

1. From the **Exploration** tab, click **Hierarchies**. The Hierarchy page opens.

2. If the **Language** list appears, choose the language of the dictionary you want to browse. The **Terminology** list populates with dictionaries in that language.

   The **Language** list is not included when all accessible dictionaries in the database are in the same language.

3. Choose a dictionary from the **Terminology** list, and click **Search**. The HTML Browser launches the Terminology Data Tree Structure; see "Browsing in the Terminology Data Tree Structure" on page 14-28.

Browsing in the Terminology Data Tree Structure

When you browse dictionary hierarchy, the Terminology Data Tree Structure window loads all of the terms from that dictionary's highest level. Figure 14–10 displays part of the tree from a session browsing the top level of the MedDRA Primary Path Dictionary.

Figure 14–10   Several Records in the Terminology Data Tree Structure

Each of the rows in this example are terms on the SOC level, the top level of this dictionary. You can use this output as a starting point for further searches into the hierarchy.

Expanding a Branch One Level Using the Arrow Keys

You can expand and contract branches up or down one level in the Repository tree by clicking the arrow keys next to each expandable term.

Expanding a Branch in the Dictionary

You can expand the tree to one node at a time to view all data under one term in the dictionary hierarchy.

Reversing the Orientation of the Tree Window

You can orient your view of the data in the Terminology Data Tree Structure window to browsing up or down the dictionary hierarchy. Browsing up the hierarchy reveals a term's derivable path; browsing down shows its child relations.

Reversing the orientation also changes which term is in focus, because you must choose a dictionary term around which you change the direction.

To reverse the orientation, choose row of the term you want to be in focus, then click the reverse icon. The Repository tree refreshes, with the newly selected term now in focus.

Launching the Term Details Window

The Term Names for each record in the Terminology Data Tree Structure are hyperlinked to the Term Details window. The Term Details window includes all of the information in the TMS database regarding the selected term, including derived path and all of the term's relations.

Searching for Source Data

Source Data Searches scan an external system for records that contain your search parameters. You can only search for data in external systems that are fully integrated with TMS, and that have at least one external system query defined in TMS (see "Setting Up External System Drill-down Queries" on page 5-36).

The Source Data Search feature can provide an easy method of finding which patients in a clinical trial took a particular drug, or reporting a specific adverse event.

Security

The Source Data Search feature is only available to TMS users with one of the following privileges: TMS_APPROVE_PRIV, TMS_CLASSIFY_PRIV, TMS_RECLASSIFY_PRIV, or TMS_RESEARCH_PRIV. If you connect to the HTML Browser as a user without any of these privileges, or your HTML Browser is configured for Auto-login, Data Searches will not be available.

Source Data Searches only return records from studies to which you have access in the external system. If you search for terms from a particular Oracle Clinical Project, and you only have access to three studies within that project, the HTML Browser only returns matches from those three studies. For more details on clinical data security in the HTML Browser, see "Available External System Data" on page 14-2.

Starting a Source Data Search

Source Data Searches follow this general procedure:

1. Choose a Simple or Advanced Search. See "Starting a Simple Search for Source Data" or "Starting an Advanced Search for Source Data".

2. Analyzing Source Data Search Results

Starting a Simple Search for Source Data

To search for external system data records that contain a particular text string:

1. If you are not in the Source Data search window, click the **Source Data** tab under the **Research** tab.

   The Source Data Search window opens (Figure 14–11).

Figure 14–11 Source Data Search Window

2. Choose a **Language** in which you want to search for data.

3. Choose an external system from the **Application** list. The Source Data Search page refreshes, updating the **Group** list.

4. Choose a **Group**.

   Groups control which external system columns you return for each external system data record. In Oracle Clinical Data Searches, for example, you can view the Project, Study, and Patient for each record. See "Setting Up External System Drill-down Queries" on page 5-36 for instructions about how to define groups.

5. Enter the text for which you want to search, then click **Search**.

The Source Data Search page refreshes, and returns any matches in the bottom part of the page. See "Analyzing Source Data Search Results" on page 14-31 for information about handling the results of a Source Data Search.

Starting an Advanced Search for Source Data

The Advanced Source Data Search window contains three sections. Choose a language for your search, then make your selections in each of these areas:

Application

You can focus a Source Data Search according to its detailed external system data. When you choose an external system from the **Application** list (Oracle Clinical, in this
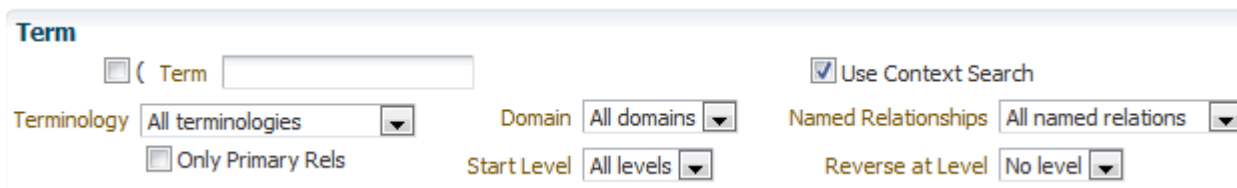
case), the page refreshes and includes that external system's columns and the databases where it is installed.

Verbatim Terms

This section enables you to focus your source data queries according to detailed information about the verbatim term or the dictionary term to which it classifies.

Term

For each term that you want to include in the search, you can specify the search keywords, dictionary and levels within that dictionary, and domain. In addition, the Advanced Search page allows you to specify whether you want to perform a simple or context search, or use a direction to include hierarchically related dictionary terms in the search.

Entering the Query Criteria

To start an Advanced Search for source data:

1. If you are not in the Source Data Search window, click the **Source Data** tab under the **Research** tab.

   The Source Data Search window opens.

2. Click the **Advanced Search** button.

   The Source Data: Advanced Search window opens.

3. Choose a **Language** for the Source Term Search.

4. Enter search criteria in any or all of the Application, Verbatim Terms or Term sections.

5. Click **Search**.

The Source Data Search page refreshes, and returns any matches in the bottom part of the page. The data displayed depends on the external system and the integration defined in TMS.

Analyzing Source Data Search Results

The presentation of source data results depends on which external system group you used for the search. The group determines which columns are displayed for each record.

In the example below, the Source Data Search for Oracle Clinical data used a group that included Project, Study, and Patient Data.

Results			
Markup	Study	Patient	Visit
📄	TMSSTUDY_OPADIC	P1	VISIT 1

To examine the patient data in Markup mode, click the **Markup** icon for the external system record you want to investigate. For information about navigating through an external system record, see "Viewing a Source Data Record" on page 14-32.

Viewing a Source Data Record

Once you find a source data record you want to investigate from your results in the Source Data Search page, click its **Markup** icon. If you have access to this record in the external system, the browser loads it in the Source Data page. If you do not have access, the browser loads an error message.

See "Defining Views and Functions in the External System" on page 5-33 for information on defining an HTML plug-in function.

Examining a Term's Details and History

Two windows provide more information about Term Records: the Term Details Window and the Term History Window. To examine the Informative Notes attributed to a Term Record, see "Examining a Record's Informative Notes" on page 14-38.

Using the Term Details Window

The Term Details window displays, in separate sections, a Term Record's entire detailed record, its derived path in the dictionary, its indication assignments, and all of its relations.

Term (Currency Status)

Figure 14–12 shows the Term part of the Term Details window.

Figure 14–12   Term Section of the Term Details Window

The Term heading displays the term's currency in parentheses. In the example in Figure 14–12, the term "Headache histamine" is current, so the value appears as "Today." When you browse an expired record, this heading includes the term's **Expiration Date** instead.

The **Hierarchy** link launches the Terminology Data tree structure, where you can browse the selected term in its position in the dictionary hierarchy. See "Browsing a Dictionary Hierarchy" on page 14-26.

The **History** link launches the Term History window, which displays the details for this term for each change in its history. See "Using the Term History Window" on page 14-35.

The **Term Details** section also includes any Informative Notes you define for the dictionaries and terms in the Repository and, if the term is a VTI, any auxiliary information associated with it.

Paths

For terms in strong dictionaries, the Term Details window includes a section for the current term's derived paths in this domain. This section does not appear for terms in weak dictionaries.

If a term has more than one derived path, the upper part of this section displays each path and its type (primary or secondary). Select the option to the left of the path you want to investigate.

The **Path Details** section shows the terms and their levels in the selected path. When you select a new path by clicking on its row, this section refreshes with the terms and levels in the newly selected derived path.

Paths

Select a path to see its details below

Path	Type
Nervous system disorders> Headaches> Migraine headaches> Migraine>	Primary
Vascular disorders> Vascular disorders NEC> Cerebrovascular and spinal vascular disorders NEC> Migraine>	Secondary

Path Details

Term	Level	Code
▽ Vascular disorders	System Organ Classes	10047065
▽ Vascular disorders NEC	High Level Group Terms	10047066
▽ Cerebrovascular and spinal vascular disorders NEC	High Level Terms	10008193
▽ Migraine	Preferred Terms	10027599
Migraine NOS	Lowest Level Terms	10027605

Indication Assignments

If a term has relations to higher level terms based on the indication for which the drug is being taken, the system displays the assignments. For example, the drug **Salbutamol** may be classified first to **Selective beta-2-adrenoreceptor agonists** and then to either **Adrenergics, Inhalants** or **Adrenergics for System Use**.

Paths

Select a path to see its details below

Path
RESPIRATORY SYSTEM> DRUGS FOR OBSTRUCTIVE AIRWAY DISEASES> ADRENERGICS, INHALANTS> Selective beta-2-adrenoreceptor agonists> Salbutamol>
RESPIRATORY SYSTEM> DRUGS FOR OBSTRUCTIVE AIRWAY DISEASES> ADRENERGICS FOR SYSTEMIC USE> Selective beta-2-adrenoreceptor agonists> Salbutamol>

Path Details

Term
▽ RESPIRATORY SYSTEM
▽ DRUGS FOR OBSTRUCTIVE AIRWAY DISEASES
▽ ADRENERGICS, INHALANTS
▽ Selective beta-2-adrenoreceptor agonists
▽ Salbutamol

Related Terms

The Related Terms part of the Term Details window lists any terms in any dictionary that contain relations to "Migraine." In this example, there are several related terms, and because all of them are in the same dictionary as the currently selected term, the browser does not display a Dictionary column in the results.

Related terms

Level	Term	Relation	Domain	Related term	Level
PT	Migraine	Strong		PAIN (SSC)	SSC
PT	Migraine	Strong		Migraine NOS	LLT
PT	Migraine	Strong		Fulgurating migraine	LLT
PT	Migraine	Strong		Migrainous neuralgia	LLT
PT	Migraine	Strong		Migrainous headache	LLT
PT	Migraine	Strong		Migraine, unspecified, with intractable migraine, so stated	LLT
HLT	Migraine headaches	Strong		Migraine	PT
PT	Migraine	Strong		Migraine, unspecified	LLT
HLT	Cerebrovascular and spinal vascular disorders NEC	Strong		Migraine	PT

Each row includes the type of relation that connects the terms. The Relation Name is hyperlinked to the Relation Details window; see "Using the Relation Details Window" on page 14-37 for more information.

There are two types of relation that this section displays. Strong relations connect terms in strong dictionaries, where each term belongs to a dictionary level, and

relations are created between terms in different dictionary levels. The Relation is listed as "Strong" when this occurs.

In weak (or "dynamic") dictionaries, all terms reside in the same dictionary level, and the named relations of type Standard describe the structure of the dictionary. The following example shows the **Related Terms** section when Narrow Scope and Broad Scope relations occur.

Level	Term	Relation	Domain	Related term
MEDDRASMQ-SMQ1	Acute pancreatitis (SMQ)	Narrow Scope		Pancreatitis NOS
MEDDRASMQ-SMQ1	Acute pancreatitis (SMQ)	Narrow Scope		Pancreatitis acute on chronic
MEDDRASMQ-SMQ1	Acute pancreatitis (SMQ)	Narrow Scope		Pancreatitis
MEDDRASMQ-SMQ1	Acute pancreatitis (SMQ)	Narrow Scope		Pancreatitis due to biliary obstruction
MEDDRASMQ-SMQ1	Acute pancreatitis (SMQ)	Broad Scope		Abdominal distension
MEDDRASMQ-SMQ1	Acute pancreatitis (SMQ)	Broad Scope		Abdominal pain upper
MEDDRASMQ-SMQ1	Acute pancreatitis (SMQ)	Broad Scope		Ascites

The **Level** field is blank for all of these records. Terms in weak dictionaries all reside on the same dictionary level, so the level is irrelevant.

Related Release Label Terms

The Related Release Label Terms section displays terms that have a named relation of type Release Label (RL) with the dictionary term you selected. However, depending on the way the named relation is defined, the selected term may appear in the Related Term column instead of the Term column.

For example, if you query for and select the term "Hepatobiliary and spleen infections" and look in the Related Release Label Terms section, you see "Hepatobiliary and spleen infections" because the two terms with RL named relations to it have a "merge" relation defined as "is merged with" so that the older terms must be displayed on the Term side and the term they "are merged with" on the Related Term side, so that the relationship is correct when read left to right.

Using the Term History Window

The Term History window has two components: the **Version** section displays a term's details for each stage of its history in the Repository, and the **Relations** section shows all relations that this term has had throughout its history. You can use the Term History window to check, for example, when a term was approved in this domain.

Version

Figure 14–13 shows the history of the MedDRA term "Parkinson's dis." The Term Valid headings identify the time periods over which the selected term details apply:

The upper row reflects the current status of the term, because it is valid until August 15, 3501. All current records in TMS have this date as the end timestamp.

The lower row reflects the status of the term from the date it was loaded into TMS (March 15, 1999), until the term was changed on March 6, 2002.

Figure 14–13 Version Section of the Term History Window

Version

Term valid from 06-Mar-2002 00:00:00 To 15-Aug-3501 00:00:00

Term	Parkinson's dis	Level	**Lowest Level Terms**
Terminology	**MedDRA Primary Path**	Approved	**No**
Code	**10034004**	Deleted by	
Id	**1564559**	Status	
Created by	**tms**	Term subtype	**External**
Domain	**GLOBAL**	Comment text	
Term type	**Dictionary Term**		
Language			

Term valid from 15-Mar-1999 00:00:00 To 06-Mar-2002 00:00:00

Term	Parkinson's dis	Level	**Lowest Level Terms**
Terminology	**MedDRA Primary Path**	Approved	**Yes**
Code	**10034004**	Deleted by	
Id	**1564559**	Status	
Created by	**tms**	Term subtype	**External**
Domain	**GLOBAL**	Comment text	
Term type	**Dictionary Term**		
Language			

By checking the differences between term details in the two sections, you can determine which changes were made on March 6, 2002. In this case, one term detail changed: the **Approval Status** was changed from **Yes** to **No**.

Relations

Relation Records in the Term History window include the creation and expiration dates for the relation, listed respectively as Creation Time and Valid Until in the headings. For verbatim terms, you can use this information to see the verbatim term's classification history.

Relations

Term	Level	Relation	Creation time	Valid until	Domain
KR SMQ I	SMQ I	Narrow Scope	26-Jul-2005 02:27:02	08-Sep-2005 06:47:25	GLOBAL
Retroauricular pain	Lowest Level Terms	Strong	24-Mar-2004 04:48:30	15-Aug-3501 00:00:00	GLOBAL
Frequent headaches	Lowest Level Terms	Strong	22-May-2003 10:55:54	15-Aug-3501 00:00:00	GLOBAL
Throbbing headache	Lowest Level Terms	Strong	22-May-2003 10:55:54	15-Aug-3501 00:00:00	GLOBAL
Intermittent headache	Lowest Level Terms	Strong	22-May-2003 10:55:54	15-Aug-3501 00:00:00	GLOBAL
Headache NOS aggravated	Lowest Level Terms	Strong	22-May-2003 10:55:54	15-Aug-3501 00:00:00	GLOBAL
Headache NOS	Lowest Level Terms	Strong	22-May-2003 10:55:54	15-Aug-3501 00:00:00	GLOBAL
PAIN (SSC)	Special Search Categories	Strong	22-May-2003 10:55:54	15-Aug-3501 00:00:00	GLOBAL
Headache (excl migraine) a...	Lowest Level Terms	Strong	22-May-2003 10:55:54	15-Aug-3501 00:00:00	GLOBAL
Drug-induced headache	Lowest Level Terms	Strong	22-May-2003 10:55:54	15-Aug-3501 00:00:00	GLOBAL
Chronic headaches	Lowest Level Terms	Strong	22-May-2003 10:55:54	15-Aug-3501 00:00:00	GLOBAL
Headache discomfort	Lowest Level Terms	Strong	22-May-2003 10:55:54	15-Aug-3501 00:00:00	GLOBAL
Premenstrual headache	Lowest Level Terms	Strong	22-May-2003 10:55:54	15-Aug-3501 00:00:00	GLOBAL
Headaches	High Level Group Terms	Strong	22-May-2003 10:55:54	15-Aug-3501 00:00:00	GLOBAL

Examining a Relation's Details and History

This section discusses the two windows that provide more information about Relation Records.

Using the Relation Details Window

The Relation Details window (see Figure 14–14) provides all of the information in the database about the relation itself, and details about the terms that this relation connects.

Figure 14–14   Relation Details Window

For relations in strong dictionaries, the **Relation** field displays the word "Strong." In dynamic dictionaries, the Relation Details window shows the Short Name of the named relation being used; the relation in Figure 14–14 uses the named relation "Broad Scope."

You can view the history of this relation by clicking the **History** button, which appears to the left of the Relation Name. See "Using the Relation History Window" on page 14-37.

Using the Relation History Window

The Relation History window displays a relation's details for each stage of its history in the database. You can use the different groupings of records in this window to see when certain aspects of a relation changed.

Figure 14–15, "Relation History Window" shows the history of the same relation under consideration in Figure 14–14, "Relation Details Window": Other newborn respiratory problems - Broad Scope - Respiratory disorder neonatal.

Figure 14–15 Relation History Window

In this example, only two records appear in the Relation History window, so this relation has only changed once since it was initially created.

Examining a Record's Informative Notes

The HTML Browser's Note History pages enable you to see which Informative Notes have been defined for a Term or Relation Record. Informative Notes can provide more detail about records in the Repository, such as the reasons why a term's status changed from Approved to Nonapproved.

To browse an Informative Note's history, navigate to the Term Details or Relation Details page, then click the Info Note History icon:

The Note History window provides information about the Informative Note at each stage of its history in the database, and the timespan during which each set of details was effective.

A

Sample Dictionary Definitions

This section describes sample or practice dictionary definitions in Oracle Thesaurus Management System (TMS) that correspond to the sample load scripts you can download from My Oracle Support from *Sample Maintenance Scripts for MedDRA, MedDRA Primary Path, MedDRA SMQ, WHO-Drug Formats B, B2, and C, and SNOMED* (article ID 258975.1) and *TMS CLASS Dictionary Loading Scripts* (article ID 428204.1); see "Searching by Article ID" on page xxii.

- Practice Dictionary CLS on page A-1

- Sample Medical Dictionary on page A-5

- Sample Medical Dictionary with a Primary Path on page A-7

- Sample Medical SMQ Dictionary on page A-11

- Sample Drug Dictionary on page A-13

- Sample Drug Dictionary with Format C on page A-17

- Sample SNOMED Dictionary on page A-24

> **Note:** The short names of the dictionary levels and of the dictionary itself must be exactly the same in the load script and in the dictionary definition in the TMS user interface.

Practice Dictionary CLS

The dictionary definition and loading process is complex, and varies depending on the dictionary and the needs of the company. You can use CLS as a practice dictionary. All the control, data, and SQL files you need are provided in a zip file on the My Oracle Support website. You can use these files as examples when you create your own dictionaries.

- CLS Structure Diagram on page A-1

- CLS Level Definitions on page A-2

- CLS Level Relations Definitions on page A-3

CLS Structure Diagram

The CLS structure diagram in Figure A–1 corresponds to the Level Definitions in "CLS Level Definitions" on page A-2 and Level Relations Definitions in "CLS Level Relations Definitions" on page A-3.

Figure A–1   CLS Structure Diagram
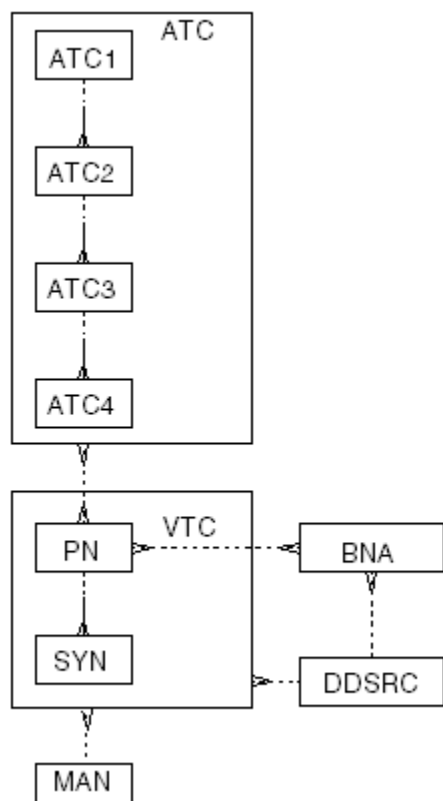
CLS Level Definitions

See "Defining the Dictionary Levels" on page 6-20 for instructions on defining dictionary levels using the Define Dictionaries window in TMS.

Table A–1   CLS Level Definitions

Short Name	Name	Classification Level?	Report Value?	Level Order	Level Type	Term Unique?	Group Short Name
ATC	Anatomical-Therapeutic-Chemical Classification Group				Group Level		
ATC1	ATC 1		X		Level		ATC
ATC2	ATC 2		X		Level		ATC
ATC3	ATC 3		X		Level		ATC
ATC4	ATC 4		X		Level		ATC
VTC	Verbatim Term Classification Group	X			Group Level	X	
PN	Preferred Name		X		Level		VTC
SYN	Synonym		X	1	Level		VTC
BNA	Drug Constituents			2	Level		
DDSRC	Source of Drug			1	Level		
MAN	Manufacturers			2	Level		

CLS Level Relations Definitions

See "Defining Relations Between Levels (Strong Dictionaries Only)" on page 6-22 for instructions on defining Level Relations in the Define Dictionaries window.

Table A–2   CLS Level Relations Definitions

Setting	Parent Level	Child Level
Level Name	Anatomical-Therapeutic-Chemical-Classification Group	ATC 1
Relation Type: Top Sublevel		
Short Name	ATC	ATC1
Mandatory?		
Many Cardinality?		
Primary Link?		n/a
Derivable?		n/a
Level Name	ATC 1	ATC 2
Relation Type: Level		
Short Name	ATC1	ATC2
Mandatory?		X
Many Cardinality?		X
Primary Link?		n/a
Derivable?	X	n/a
Level Name	ATC 2	ATC 3
Relation Type: Level		
Short Name	ATC2	ATC3
Mandatory?		X
Many Cardinality?		X
Primary Link?		n/a
Derivable?	X	n/a
Level Name	ATC 3	ATC 4
Relation Type: Level		
Short Name	ATC3	ATC4
Mandatory?		X
Many Cardinality?		X
Primary Link?		n/a
Derivable?	X	n/a
Level Name	Anat.-Therap.-Chem.-Classification Group	Preferred Name
Relation Type: Level		
Short Name	ATC	PN
Mandatory?		
Many Cardinality?	X	X

Table A–2   (Cont.) CLS Level Relations Definitions

Setting	Parent Level	Child Level
Primary Link?	X	n/a
Derivable?	X	n/a
Level Name	Preferred Name	Synonym
Relation Type: Level		
Short Name	PN	SYN
Mandatory?		X
Many Cardinality?		X
Primary Link?		n/a
Derivable?	X	n/a
Level Name	Preferred Name	Drug Constituents
Relation Type: Level		
Short Name	PN	BNA
Mandatory?		
Many Cardinality?	X	X
Primary Link?		n/a
Derivable?		n/a
Level Name	Drug Constituents	Source of Drug
Relation Type: Level		
Short Name	BNA	DDSRC
Mandatory?		
Many Cardinality?	X	
Primary Link?		n/a
Derivable?		n/a
Level Name	Verbatim Term Classification Group	Preferred Name
Relation Type: Top Sublevel		
Short Name	VTC	PN
Mandatory?		
Many Cardinality?		
Primary Link?		n/a
Derivable?		n/a
Level Name	Verbatim Term Classification Group	Manufacturers
Relation Type: Level		
Short Name	VTC	MAN
Mandatory?		
Many Cardinality?	X	
Primary Link?		n/a
Derivable?		n/a

Table A–2   (Cont.) CLS Level Relations Definitions

Setting	Parent Level	Child Level
Level Name	Verbatim Term Classification Group	Source of Drug
Relation Type: Level		
Short Name	VTC	DDSRC
Mandatory?		
Many Cardinality?	X	
Primary Link?		n/a
Derivable?		n/a

Sample Medical Dictionary

This section provides a structure diagram of the sample medical dictionary, and the dictionary level and level relations definitions that are required to create a dictionary with this structure.

- Sample Medical Dictionary Structure on page A-5

- Sample Medical Dictionary Level Definitions on page A-6

- Sample Medical Dictionary Level Relations Definitions on page A-6

Sample Medical Dictionary Structure

The sample medical dictionary structure in Figure A–2 corresponds to the definitions provided in "Sample Medical Dictionary Level Definitions" on page A-6 and "Sample Medical Dictionary Level Relations Definitions" on page A-6.

Figure A–2   Sample Medical Dictionary Structure
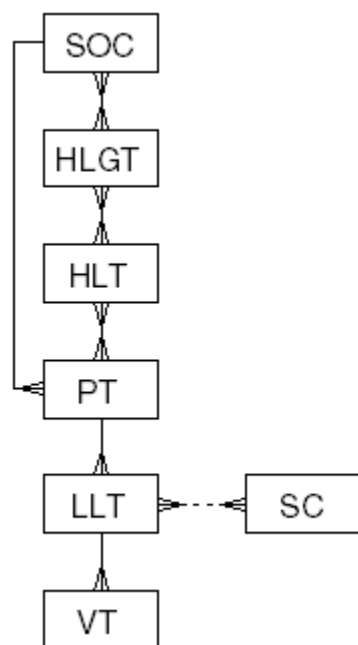
Sample Medical Dictionary Level Definitions

See "Defining the Dictionary Levels" on page 6-20 for instructions on defining dictionary levels using the Define Dictionaries window in TMS.

Table A–3   Sample Medical Dictionary Level Definitions

Short Name	Name	Classifica-tion Level?	Report Value	Level Order	Level Type	Term Unique?	Group Short Name
SOC	System Organ Class		X		Level		
HLGT	High Level Group Term				Level		
HLT	High Level Term				Level		
PT	Preferred Term		X		Level		
LLT	Lowest Level Term	X	X	1	Level		
SC	Special Category			2	Level		

Sample Medical Dictionary Level Relations Definitions

See "Defining Relations Between Levels (Strong Dictionaries Only)" on page 6-22 for instructions on defining Level Relations in the Define Dictionaries window.

Table A–4   Sample Medical Dictionary Level Relations Definitions

Setting	Parent Level	Child Level
Level Name	System Organ Class	High Level Group Term
Relation Type: Level		
Short Name	SOC	HLGT
Mandatory?	X	X
Many Cardinality?	X	X
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?		n/a
Level Name	High Level Group Term	High Level Term
Relation Type: Level		
Short Name	HLGT	HLT
Mandatory?	X	X
Many Cardinality?	X	X
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?		n/a
Level Name	High Level Term	Preferred Term
Relation Type: Level		
Short Name	HLT	PT
Mandatory?	X	X

Table A–4   (Cont.) Sample Medical Dictionary Level Relations Definitions

Setting	Parent Level	Child Level
Many Cardinality?	X	X
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?		n/a
Level Name	Preferred Term	Lowest Level Term
Relation Type: Level		
Short Name	PT	LLT
Mandatory?	X	X
Many Cardinality?		X
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?	X	n/a
Level Name	System Organ Class	Preferred Term
Relation Type: Level		
Short Name	SOC	PT
Mandatory?	X	X
Many Cardinality?		X
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?	X	n/a
Level Name	Preferred Term	Special Category
Relation Type: Level		
Short Name	PT	SC
Mandatory?		
Many Cardinality?	X	X
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?		n/a

Sample Medical Dictionary with a Primary Path

This section provides a structure diagram of the sample medical dictionary with a primary path, and the dictionary level and level relations definitions that are required to create a dictionary with this structure.

-

-

-

Sample Medical Dictionary with a Primary Path Structure

The sample medical dictionary structure in Figure A–3 corresponds to the definitions provided in "Sample Medical Dictionary with a Primary Path Dictionary Level Definitions" on page A-8 and "Sample Medical Dictionary with a Primary Path Level Relations Definitions" on page A-9.

Figure A–3   Sample Medical Dictionary with Primary Path Structure
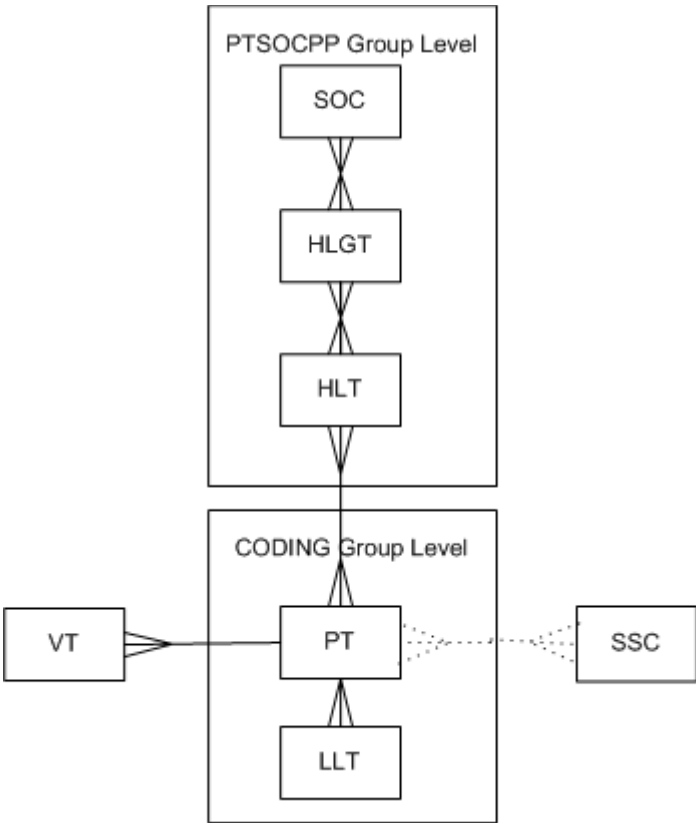
Sample Medical Dictionary with a Primary Path Dictionary Level Definitions

See "Defining the Dictionary Levels" on page 6-20 for instructions on defining dictionary levels using the Define Dictionaries window in TMS.

Table A–5   Sample Medical Dictionary with Primary Path Dictionary Level Definitions

Short Name	Name	Classifica-tion Level?	Report Value	Level Order	Level Type	Term Unique?	Group Short Name
PTSOCPP	Primary Path PT->SOC			0	Group Level	None	
SOC	System Organ Classes		X	0	Level	Level	PTSOCPP
HLGT	High Level Group Terms		X	0	Level	Level	PTSOCPP
HLT	High Level Terms		X	0	Level	Level	PTSOCPP
CODING	Coding Group	X		0	Group Level	Level	

Table A–5   (Cont.) Sample Medical Dictionary with Primary Path Dictionary Level Definitions

Short Name	Name	Classifica-tion Level?	Report Value	Level Order	Level Type	Term Unique?	Group Short Name
PT	Preferred Terms		X	100	Level	None	CODING
LLT	Lowest Level Terms		X	0	Level	Non	CODING
SC	Special Search Categories			1	Level	Level	
VT	Verbatim Term			500	Verbatim Level	Level	

Sample Medical Dictionary with a Primary Path Level Relations Definitions

See "Defining Relations Between Levels (Strong Dictionaries Only)" on page 6-22 for instructions on defining Level Relations in the Define Dictionaries window.

Table A–6   Sample Medical Dictionary with Primary Path Level Relations

Setting	Parent Level	Child Level
Level Name	Primary Path PT->SOC	System Organ Class
Relation Type: Level		
Short Name	PTSOCPP	SOC
Mandatory?		
Many Cardinality?		
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?		n/a
Level Name	System Organ Class	High Level Group Term
Relation Type: Level		
Short Name	SOC	HLGT
Mandatory?	X	X
Many Cardinality?	X	X
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?		n/a
Level Name	High Level Group Term	High Level Term
Relation Type: Level		
Short Name	HLGT	HLT
Mandatory?	X	X
Many Cardinality?	X	X
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?		n/a
Level Name	High Level Term	Preferred Term

Table A–6   (Cont.) Sample Medical Dictionary with Primary Path Level Relations

Setting	Parent Level	Child Level
Relation Type: Level		
Short Name	HLT	PT
Mandatory?	X	X
Many Cardinality?	X	X
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?		n/a
Level Name	Preferred Term	Lowest Level Term
Relation Type: Level		
Short Name	PT	LLT
Mandatory?	X	X
Many Cardinality?		X
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?	X	n/a
Level Name	Primary Path PT->SOC	Preferred Terms
Relation Type: Level		
Short Name	PTSOCPP	PT
Mandatory?	X	X
Many Cardinality?	X	X
Primary Link?		n/a
Primary Path Link?	X	n/a
Derivable?	X	n/a
Level Name	System Organ Class	Preferred Terms
Relation Type: Level		
Short Name	SOC	PT
Mandatory?	X	X
Many Cardinality?		X
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?	X	n/a
Level Name	Preferred Terms	Special Categories
Relation Type: Level		
Short Name	PT	SC
Mandatory?		
Many Cardinality?	X	X
Primary Link?		n/a

Table A–6   (Cont.) Sample Medical Dictionary with Primary Path Level Relations

Setting	Parent Level	Child Level
Primary Path Link?		n/a
Derivable?		n/a
Level Name	Coding Group	Preferred Terms
Relation Type: Level		
Short Name	CODING	PT
Mandatory?		
Many Cardinality?		
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?		n/a
Level Name	Coding Group	Verbatim Term
Relation Type: Verbatim Level		
Short Name	CODING	SC
Mandatory?		X
Many Cardinality?		X
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?	X	n/a

Sample Medical SMQ Dictionary

This section provides a structure diagram of the sample medical SMQ dictionary, and the dictionary level and level relations definitions that are required to create a dictionary with this structure.

- Sample Medical SMQ Dictionary Structure on page A-11

- Sample Medical SMQ Dictionary Level Definitions on page A-12

- Sample Medical SMQ Dictionary Level Relation Definitions on page A-12

Sample Medical SMQ Dictionary Structure

The sample medical dictionary structure in Figure A–4 corresponds to the definitions provided in "Sample Medical SMQ Dictionary Level Definitions" on page A-12 and "Sample Medical SMQ Dictionary Level Relation Definitions" on page A-12.

Term Uniqueness Enforced? is set to N (unchecked) at the dictionary level.

Figure A–4   Sample Medical SMQ Dictionary Structure
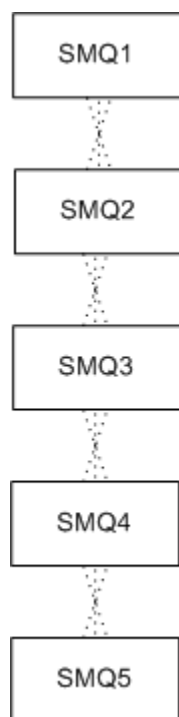
Sample Medical SMQ Dictionary Level Definitions

The following table provides the Level Definitions for the sample medical SMQ dictionary. See "Defining the Dictionary Levels" on page 6-20 for instructions on defining dictionary levels using the Define Dictionaries window in TMS.

Table A–7   Sample Medical SMQ Dictionary Level Definitions

Short Name	Name	Classifica- tion Level?	Report Value	Level Order	Level Type	Term Unique?	Group Short Name
SMQ1	SMQ I			0	Level	None	
SMQ2	SMQ II			0	Level	None	
SMQ3	SMQ III			0	Level	None	
SMQ4	SMQ IV			0	Level	None	
SMQ5	SMQ V			0	Level	None	

Sample Medical SMQ Dictionary Level Relation Definitions

The following table provides the Level Relations Definitions for the sample medical SMQ dictionary. See "Defining Relations Between Levels (Strong Dictionaries Only)" on page 6-22 for instructions on defining Level Relations in the Define Dictionaries window.

Table A–8   Sample Medical SMQ Dictionary Level Relation Definitions

Setting	Parent Level	Child Level
Level Name	SMQ I	SMQ II

Table A–8   (Cont.) Sample Medical SMQ Dictionary Level Relation Definitions

| Setting	Parent Level	Child Level
Relation Type: Level		
Short Name	SMQ1	SMQ2
Mandatory?		
Many Cardinality?	X	X
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?		n/a
Level Name	SMQ II	SMQ III
Relation Type: Level		
Short Name	SMQ2	SMQ3
Mandatory?		
Many Cardinality?	X	X
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?		n/a
Level Name	SMQ III	SMQ IV
Relation Type: Level		
Short Name	SMQ3	SMQ4
Mandatory?		
Many Cardinality?	X	X
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?		n/a
Level Name	SMQ IV	SMQ V
Relation Type: Level		
Short Name	SMQ4	SMQ5
Mandatory?		
Many Cardinality?	X	X
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?		n/a

Sample Drug Dictionary

This section provides a structural diagram of the sample drug dictionary, and the dictionary level and Level Relations Definitions that are required to create a dictionary with this structure.

This dictionary is the same as CLS except that no Primary Link is defined between the ATC Group level and the Preferred Name level, and therefore you cannot derive terms from any of the sublevels within the ATC Group.

- Sample Drug Dictionary Structure on page A-14
- Sample Drug Dictionary Level Definitions on page A-14
- Sample Drug Dictionary Level Relations Definitions on page A-15

Sample Drug Dictionary Structure

The sample drug dictionary structure in Figure A–5 corresponds to the definitions provided in "Sample Drug Dictionary Level Definitions" on page A-14 and "Sample Drug Dictionary Level Relations Definitions" on page A-15.

Figure A–5   Sample Drug Dictionary Structure
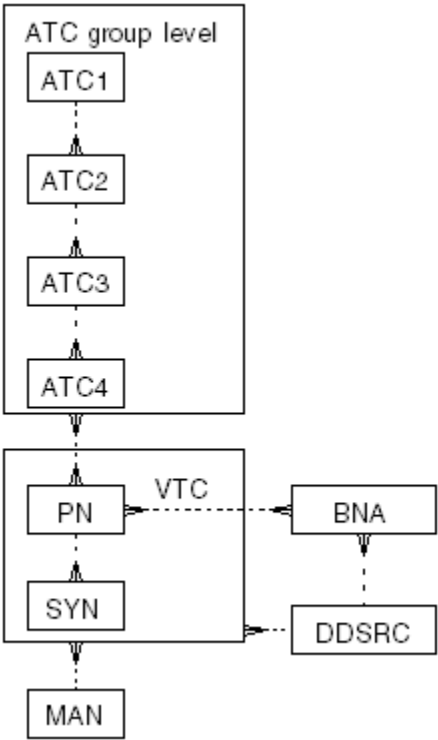
Sample Drug Dictionary Level Definitions

See "Defining the Dictionary Levels" on page 6-20 for instructions on defining dictionary levels using the Define Dictionaries window in TMS.

Table A–9   Sample Drug Dictionary Level Definitions

Short Name	Name	Classifica-tion Level?	Report Value?	Level Order	Level Type	Term Unique?	Group Short Name
ATC	Anatomical-Therapeutic-Chemical Classification Group				Group Level		
ATC1	ATC 1				Level		ATC

Table A–9   (Cont.) Sample Drug Dictionary Level Definitions

Short Name	Name	Classifica-tion Level?	Report Value?	Level Order	Level Type	Term Unique?	Group Short Name
ATC2	ATC 2				Level		ATC
ATC3	ATC 3				Level		ATC
ATC4	ATC 4				Level		ATC
VTC	Verbatim Term Classification Group	X			Group Level	X	
PN	Preferred Name		X		Level		VTC
SYN	Synonym		X	1	Level		VTC
BNA	Drug Constituents			2	Level		
DDSRC	Source of Drug			1	Level		
MAN	Manufacturers			2	Level		

Sample Drug Dictionary Level Relations Definitions

See for instructions on defining Level Relations in the Define Dictionaries window.

Table A–10   Sample Drug Dictionary Level Relations Definitions

Setting	Parent Level	Child Level
Level Name	Anatomical-Therapeutic-Chemical-Classification Group	ATC 1
Relation Type: Top Sublevel		
Short Name	ATC	ATC1
Mandatory?		
Many Cardinality?		
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?		n/a
Level Name	ATC 1	ATC 2
Relation Type: Level		
Short Name	ATC1	ATC2
Mandatory?		X
Many Cardinality?		X
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?		n/a
Level Name	ATC 2	ATC 3
Relation Type: Level		
Short Name	ATC2	ATC3

Table A–10 (Cont.) Sample Drug Dictionary Level Relations Definitions

Setting	Parent Level	Child Level
Mandatory?		X
Many Cardinality?		X
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?		n/a
Level Name	ATC 3	ATC 4
Relation Type: Level		
Short Name	ATC3	ATC4
Mandatory?		X
Many Cardinality?		X
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?		n/a
Level Name	Anat.-Therap.-Chem.-Classification Group	Preferred Name
Relation Type: Level		
Short Name	ATC	PN
Mandatory?		
Many Cardinality?	X	X
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?		n/a
Level Name	Preferred Name	Synonym
Relation Type: Level		
Short Name	PN	SYN
Mandatory?		X
Many Cardinality?		X
Primary Link?		n/a
Derivable?	X	n/a
Level Name	Preferred Name	Drug Constituents
Relation Type: Level		
Short Name	PN	BNA
Mandatory?		
Many Cardinality?	X	X
Primary Link?		n/a
Derivable?		n/a
Primary Path Link?		n/a
Level Name	Drug Constituents	Source of Drug

Table A–10   (Cont.) Sample Drug Dictionary Level Relations Definitions

Setting	Parent Level	Child Level
Relation Type: Level		
Short Name	BNA	DDSRC
Mandatory?		
Many Cardinality?	X	
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?		n/a
Level Name	Verbatim Term Classification Group	Preferred Name
Relation Type: Top Sublevel		
Short Name	VTC	PN
Mandatory?		
Many Cardinality?		
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?		n/a
Level Name	Verbatim Term Classification Group	Manufacturers
Relation Type: Level		
Short Name	VTC	MAN
Mandatory?		
Many Cardinality?	X	
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?		n/a
Level Name	Verbatim Term Classification Group	Source of Drug
Relation Type: Level		
Short Name	VTC	DDSRC
Mandatory?		
Many Cardinality?	X	
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?		n/a

Sample Drug Dictionary with Format C

This section provides a structural diagram of the sample drug format C dictionary, and the dictionary level and Level Relations Definitions that are required to create a dictionary with this structure.

- Sample Drug Dictionary Structure on page A-14

Sample Drug Dictionary with Format C Structure

The sample drug dictionary structure in Figure A–5 corresponds to the definitions provided in "Sample Drug Dictionary Level Definitions" on page A-14 and "Sample Drug Dictionary Level Relations Definitions" on page A-15.

Figure A–6   Sample Drug Dictionary with Format C Structure
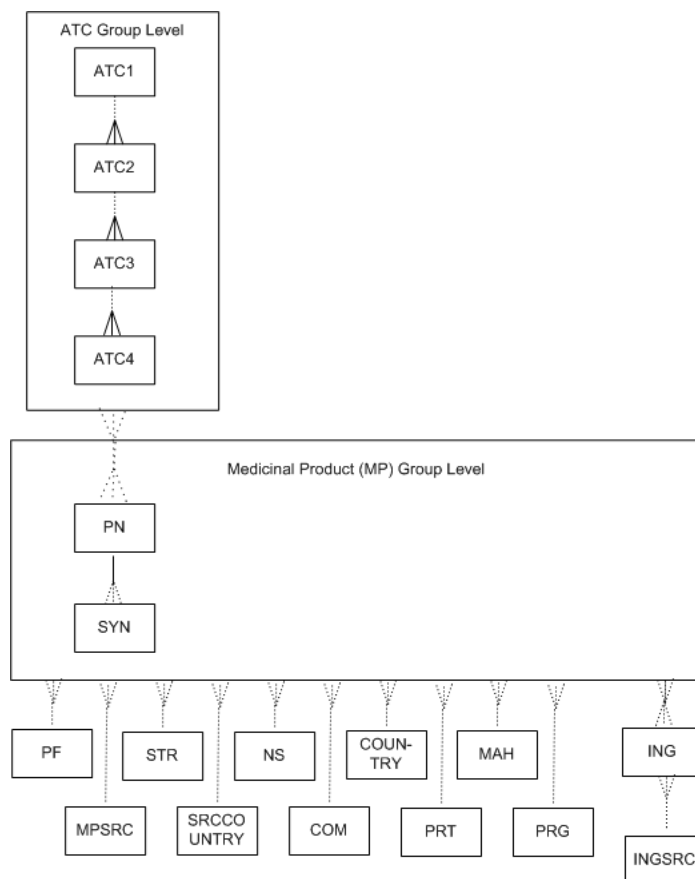
Sample Drug Dictionary with Format C Level Definitions

See "Defining the Dictionary Levels" on page 6-20 for instructions on defining dictionary levels using the Define Dictionaries window in TMS.

Table A–11   Sample Drug Dictionary with Format C Level Definitions

Short Name	Name	Classification Level?	Report Value?	Level Order	Level Type	Term Unique?	Group Short Name
ATC	Anatomical-Therapeutic-Chemical Classification Group			0	Group Level	None	
ATC1	ATC 1		X	0	Level	None	ATC

Table A–11   (Cont.) Sample Drug Dictionary with Format C Level Definitions

Short Name	Name	Classifi-cation Level?	Report Value?	Level Order	Level Type	Term Unique?	Group Short Name
ATC2	ATC 2		X	0	Level	None	ATC
ATC3	ATC 3		X	0	Level	None	ATC
ATC4	ATC 4		X	0	Level	None	ATC
MP	Medicinal Product Group	X			Group Level	None	
PN	Preferred Name		X	10	Level	None	MP
SYN	Synonym		X	0	Level	None	MP
PF	Pharmaceutical Form			20	Level	None	
STR	Strength			30	Level	None	
NS	Name Specifier			40	Level	None	
COUNTRY	Country			50	Level	None	
MAH	Marketing Authorization Holder			60	Level	None	
ING	Ingredient			70	Level	None	
INGSRC	Ingredient Reference			0	Level	None	
MPSRC	Reference			80	Level	None	
SRCCOUNTRY	Source Country			90	Level	None	
COM	Company			100	Level	None	
PRT	Product Type			110	Level	None	
PRG	Product Group			120	Level	None	

Sample Drug Dictionary Level Relations Definitions

See for instructions on defining Level Relations in the Define Dictionaries window.

Table A–12   Sample Drug Dictionary Level Relations Definitions

Setting	Parent Level	Child Level
Level Name	Anatomical-Therapeutic-Chemical-Classification Group	ATC 1
Relation Type: Top Sublevel		
Short Name	ATC	ATC1
Mandatory?		
Many Cardinality?		
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?		n/a

Table A–12   (Cont.) Sample Drug Dictionary Level Relations Definitions

Setting	Parent Level	Child Level
Level Name	ATC 1	ATC 2
Relation Type: Level		
Short Name	ATC1	ATC2
Mandatory?		X
Many Cardinality?		X
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?		n/a
Level Name	ATC 2	ATC 3
Relation Type: Level		
Short Name	ATC2	ATC3
Mandatory?		X
Many Cardinality?		X
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?		n/a
Level Name	ATC 3	ATC 4
Relation Type: Level		
Short Name	ATC3	ATC4
Mandatory?		X
Many Cardinality?		X
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?		n/a
Level Name	Anat.-Therap.-Chem.-Classification Group	Preferred Name
Relation Type: Level		
Short Name	ATC	PN
Mandatory?		
Many Cardinality?	X	X
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?	X	n/a
Level Name	Preferred Name	Synonym
Relation Type: Level		
Short Name	PN	SYN
Mandatory?	X	X
Many Cardinality?		X

Table A–12   (Cont.) Sample Drug Dictionary Level Relations Definitions

Setting	Parent Level	Child Level
Primary Link?		n/a
Derivable?	X	n/a
Level Name	Medicinal Product Group	Preferred Name
Relation Type: Level		
Short Name	MP	PN
Mandatory?		
Many Cardinality?		X
Primary Link?		n/a
Derivable?		n/a
Primary Path Link?		n/a
Level Name	Medicinal Product Group	Pharmaceutical Form
Relation Type: Top Sublevel		
Short Name	MP	PF
Mandatory?		
Many Cardinality?	X	
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?		n/a
Level Name	Medicinal Product Group	Strength
Relation Type: Level		
Short Name	MP	STR
Mandatory?		
Many Cardinality?	X	
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?		n/a
Level Name	Medicinal Product Group	Name Specifier
Relation Type: Level		
Short Name	MP	NS
Mandatory?		
Many Cardinality?	X	
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?		n/a
Level Name	Medicinal Product Group	Country
Relation Type: Level		
Short Name	MP	COUNTRY

Table A–12   (Cont.) Sample Drug Dictionary Level Relations Definitions

Setting	Parent Level	Child Level
Mandatory?		
Many Cardinality?	X	
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?		n/a
Level Name	Medicinal Product Group	Marketing Authorization Holder
Relation Type: Level		
Short Name	MP	MAH
Mandatory?		
Many Cardinality?	X	
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?		n/a
Level Name	Medicinal Product Group	Ingredient
Relation Type: Level		
Short Name	MP	ING
Mandatory?		
Many Cardinality?		X
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?		n/a
Level Name	Ingredient	Ingredient Reference
Relation Type: Level		
Short Name	ING	INGSRC
Mandatory?		
Many Cardinality?	X	
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?		n/a
Level Name	Medicinal Product Group	Reference
Relation Type: Level		
Short Name	MP	MPSRC
Mandatory?		
Many Cardinality?	X	
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?		n/a

Table A–12   (Cont.) Sample Drug Dictionary Level Relations Definitions

Setting	Parent Level	Child Level
Level Name	Medicinal Product Group	Source Country
Relation Type: Level		
Short Name	MP	SRCCOUNTRY
Mandatory?		
Many Cardinality?	X	
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?		n/a
Level Name	Medicinal Product Group	Company
Relation Type: Level		
Short Name	MP	COM
Mandatory?		
Many Cardinality?	X	
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?		n/a
Level Name	Medicinal Product Group	Product Type
Relation Type: Level		
Short Name	MP	PRT
Mandatory?		
Many Cardinality?	X	
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?		n/a
Level Name	Medicinal Product Group	Product Group
Relation Type: Level		
Short Name	MP	PRG
Mandatory?		
Many Cardinality?	X	
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?		n/a

WHO-Drug C Data Loaded as TMS Informative Notes

Some WHO-Drug Format C tables, which are loaded into TMS as dictionary levels, contain too many attributes, or columns, to fit into tables tms_dict_contents and tms_dict_relations. In these cases the sample scripts load certain column data as Informative Notes associated with terms that have data in the column.

All mappings are detailed in the sample script code.

Table A–13   WHO-Drug C Data Handled as Informative Notes in TMS

WHO-Drug C Table/ TMS Dictionary Level	WHO-Drug C Table Column	Informative Note Attribute
Medicinal Product	AUTHNUM	Marketing Authorization Number
Medicinal Product	AUTHDATE	Marketing Authorization Date
Medicinal Product	WITHDATE	Marketing Auth Withdrawal Date
Medicinal Product and Substance	REFYEAR	Year of Reference
Medicinal Product and Ingredient	CREATEDATE	Create Date
Medicinal Product	CHGDATE	Date Changed

Sample SNOMED Dictionary

This section provides a structural diagram of the sample SNOMED dictionary, and the dictionary level and Level Relations Definitions that are required to create a dictionary with this structure.

- Sample SNOMED Dictionary Structure on page A-24
- Sample SNOMED Dictionary Level Definitions on page A-24
- Sample SNOMED Dictionary Level Relations on page A-25

Sample SNOMED Dictionary Structure

The sample SNOMED dictionary structure in Figure A–7 corresponds to the definitions provided in "Sample SNOMED Dictionary Level Definitions" on page A-24 and "Sample SNOMED Dictionary Level Relations" on page A-25.

Figure A–7   Sample SNOMED Dictionary Structure
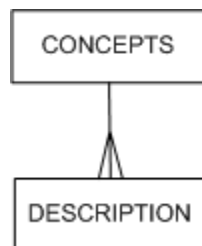
Sample SNOMED Dictionary Level Definitions

See "Defining the Dictionary Levels" on page 6-20 for instructions on defining dictionary levels using the Define Dictionaries window in TMS.

Table A–14   Sample SNOMED Dictionary Level Definitions

Short Name	Name	Classification Level?	Report Value?	Level Order	Level Type	Term Unique?	Group Short Name
CONCEPT	Concept			0	Level	None	
DESC	Description			0	Level	None	

Sample SNOMED Dictionary Level Relations

See "Defining Relations Between Levels (Strong Dictionaries Only)" on page 6-22 for instructions on defining Level Relations in the Define Dictionaries window.

Table A–15   Sample SNOMED Dictionary Level Relations

Setting	Parent Level	Child Level
Level Name	Concept	Description
Relation Type: Level		
Short Name	CONCEPT	DESC
Mandatory?	X	X
Many Cardinality?		X
Primary Link?		n/a
Primary Path Link?		n/a
Derivable?		n/a

B

Disconnected System Integration XML Files

This appendix contains information about the XML files used to exchange data between the sponsor and CRO sites using Oracle Thesaurus Management System (TMS) Disconnected System Integration (DSI). For more information about DSI, see "Using TMS with Disconnected Systems" on page 5-3.

This appendix contains the following sections:

- XML File Names on page B-1

- XML File Structure on page B-1

- Key to XML File Tags on page B-3

- Validation of Non-TMS XML Files on page B-5

XML File Names

An export job extracts data into one or more XML files that it creates in the Output directory. DSI avoids naming conflicts by generating the XML file name at extraction time. The naming convention for files exported from a CRO site is:

database_global_name||*x_area*||*database_sequence_number*.xml

Names for files exported from the sponsor site follow the same convention, except that they also contain the prefix SPN_.

In both cases, all characters must be in uppercase.

In the Database Global Name portion, any periods (.) are replaced by underscores (_). For an explanation of X Areas, see "Defining X Areas" on page 5-9.

If any errors are encountered when the file is imported, the error file will appear with the same name in the error directory.

XML File Structure

The XML file has two sections: the metadata definition, which is used to map the metadata of two disconnected systems (see "Defining XML Tag Mappings" on page 5-10), and the actual source data. For information about the tags, see Table B–1, " XML Section Tags" on page B-4 and Table B–2, " XML Data and Metadata Tags" on page B-4.

Example B–1   Source Site XML Sample Structure

```
<?xml version="1.0" encoding="UTF-8" ?>
<TMSDSI>
  <META_DATA>
```

```
        <INTEGRATION_KEY>OCL</INTEGRATION_KEY>
        <INSTANCE_NAME>SUN5X13.US.ORACLE.COM</INSTANCE_NAME>
        <X_AREA>1</X_AREA>
        <X_AREA_NAME>TMSLEARN</X_AREA_NAME>
        <GMT_DIFF>-10:00</GMT_DIFF>
    </META_DATA>
    <VT>
     <VT_ROW>
        <DICTIONARY_SHORT_NAME>MEDDRAPP</DICTIONARY_SHORT_NAME>
        <DOMAIN_NAME>ALLDICTIONARIES</DOMAIN_NAME>
        <VERBATIM_TERM>SORE SORE MUSCLE PAIN</VERBATIM_TERM>
        <VT_NOTE>
         <VT_NOTE_ROW>
            <LABEL_TEXT_UPPER>CODER COMMENT</LABEL_TEXT_UPPER>
            <OCCURRENCE_ID>1</OCCURRENCE_ID>
            <DEF_DETAIL_VALUE>another test of special chars %^$#@!~`&*()_
-=+|\}{[];:'"<>,.?/</DEF_DETAIL_VALUE>
            <CREATED_BY LOCAL_FLAG="Y">TMS</CREATED_BY>
         </VT_NOTE_ROW>
        </VT_NOTE>
     </VT_ROW>
    </VT>
     <SOURCE_DATA_DEL />
     <SOURCE_DATA>
      <SOURCE_DATA_ROW>
        <DICTIONARY_SHORT_NAME>MEDDRAPP</DICTIONARY_SHORT_NAME>
        <DOMAIN_NAME>ALLDICTIONARIES</DOMAIN_NAME>
        <VERBATIM_TERM>CORN BEEF 1</VERBATIM_TERM>
        <SOURCE_TERM_ID>300000005</SOURCE_TERM_ID>
        <OCCURRENCE_ID>0</OCCURRENCE_ID>
        <EXT_VALUE_1>TMSPROJ</EXT_VALUE_1>
        <EXT_VALUE_2>TMSSTUDY2</EXT_VALUE_2>
        <EXT_VALUE_3>M_INV1</EXT_VALUE_3>
        <EXT_VALUE_4>VISIT2</EXT_VALUE_4>
        <EXT_VALUE_5>101</EXT_VALUE_5>
        <EXT_VALUE_6>KEN804</EXT_VALUE_6>
        <EXT_VALUE_7>OPA</EXT_VALUE_7>
        <EXT_VALUE_8>66201</EXT_VALUE_8>
        <OMISSION_STATUS>TMS IN PROGRESS</OMISSION_STATUS>
        <ACTION_TEXT>Can you clarify please?</ACTION_TEXT>
        <ACTION_OWNER LOCAL_FLAG="Y">OLS$JSMITH</ACTION_OWNER>
      </SOURCE_DATA_ROW>
     </SOURCE_DATA>
</TMSDSI>
```

Example B–2   Sponsor Site XML Sample Structure

```
<?xml version="1.0" encoding="UTF-8" ?>
<TMSDSI>
 <META_DATA>
    <INTEGRATION_KEY>OCL</INTEGRATION_KEY>
    <INSTANCE_NAME>EXAMPLE.US.ORACLE.COM</INSTANCE_NAME>
  <X_AREA>1</X_AREA>
    <X_AREA_NAME>LEARN</X_AREA_NAME>
  <GMT_DIFF>-04:00</GMT_DIFF>
 </META_DATA>
 <VTA_DEL />
 <VTI_DEL>
 <VTI_DEL_ROW><DICTIONARY_SHORT_NAME>WHOC2</DICTIONARY_SHORT_NAME><DOMAIN_
NAME>DOM1</DOMAIN_NAME><VERBATIM_TERM>BACK HEADPAIN</VERBATIM_TERM><SOURCE_TERM_
```

```
ID>36838008</SOURCE_TERM_ID></VTI_DEL_ROW>
 </VTI_DEL>
 <VTA>
 <VTA_ROW><DICTIONARY_SHORT_NAME>WHOC2</DICTIONARY_SHORT_NAME><DOMAIN_
NAME>DOM1</DOMAIN_NAME><VERBATIM_TERM>TERM141</VERBATIM_TERM><VT_SUBTYPE>AC</VT_
SUBTYPE><DICTIONARY_TERM>PT1</DICTIONARY_TERM><APPROVED_FLAG>Y</APPROVED_
FLAG><VTA_GLOBAL_FLAG>N</VTA_GLOBAL_FLAG><VTA_CREATED_BY LOCAL_
FLAG="Y">USER1</VTA_CREATED_BY></VTA_ROW>
 </VTA>
 <VTI>
<VTI_ROW><DICTIONARY_SHORT_NAME>WHOC2</DICTIONARY_SHORT_NAME><DOMAIN_
NAME>DOM1</DOMAIN_NAME><VERBATIM_TERM>TERM71</VERBATIM_TERM><SOURCE_TERM_
ID>36838005</SOURCE_TERM_ID><OCCURRENCE_ID>2</OCCURRENCE_ID><VT_SUBTYPE>AC</VT_
SUBTYPE><DICTIONARY_TERM>PTNONUQ1</DICTIONARY_TERM><DICT_CONTENT_
CODE>NONUQ3</DICT_CONTENT_CODE><VTI_CREATED_BY LOCAL_FLAG="Y">USER1</VTI_CREATED_
BY></VTI_ROW>
 </VTI>
 <VT>
  <VT_ROW>
    <DICTIONARY_SHORT_NAME>MEDDRAPP</DICTIONARY_SHORT_NAME>
    <DOMAIN_NAME>ALLDICTIONARIES</DOMAIN_NAME>
    <VERBATIM_TERM>CORN BEEF 1</VERBATIM_TERM>
   <VT_NOTE>
    <VT_NOTE_ROW>
        <LABEL_TEXT_UPPER>CODER COMMENT</LABEL_TEXT_UPPER>
        <OCCURRENCE_ID>1</OCCURRENCE_ID>
        <DEF_DETAIL_VALUE>SUN5X12 What is this "thing" with corn beef?</DEF_
DETAIL_VALUE>
      <CREATED_BY LOCAL_FLAG="Y">TMS</CREATED_BY>
     </VT_NOTE_ROW>
   </VT_NOTE>
  </VT_ROW>
 <VT_ROW>
      <DICTIONARY_SHORT_NAME>MEDDRAPP</DICTIONARY_SHORT_NAME>
      <DOMAIN_NAME>ALLDICTIONARIES</DOMAIN_NAME>
      <VERBATIM_TERM>SORE SORE MUSCLE PAIN</VERBATIM_TERM>
   <VT_NOTE>
    <VT_NOTE_ROW>
      <LABEL_TEXT_UPPER>CODER COMMENT</LABEL_TEXT_UPPER>
      <OCCURRENCE_ID>1</OCCURRENCE_ID>
      <DEF_DETAIL_VALUE>From SUN5X12 %^$#@!~`&*()_-=+|\}{[];:'"<>,.?/</DEF_
DETAIL_VALUE>
      <CREATED_BY LOCAL_FLAG="Y">TMS</CREATED_BY>
     </VT_NOTE_ROW>
   </VT_NOTE>
  </VT_ROW>
 </VT>
 <SOURCE_DATA />
</TMSDSI>
```

Key to XML File Tags

The following tables provide an explanation of the tags used in the required XML file structure.

Table B–1   XML Section Tags

Tag	Description
`<TMSDSI>`	File data to be processed by DSI.
`<META_DATA>`	Metadata information that applies to all data in the XML file.
`<SOURCE_DATA>`	Source data to be processed.
`<SOURCE_DATA_ROW>`	Individual source data rows.
`<SOURCE_DATA_DEL>`	Source data deletes to be processed.
`<SOURCE_DATA_DEL_ROW>`	Individual source data rows to be deleted.
`<VTA>`	Verbatim Term Assignments.
`<VTA_ROW>`	Verbatim Term Assignments Rows.
`<VTA_NOTE>`	Workflow Informative Notes for VTA.
`<VTA_NOTE_ROW>`	Workflow Informative Note row for VTA.
`<VTA_DEL>`	Declassified Verbatim Term Assignments.
`<VTA_DEL_ROW>`	Declassified Verbatim Term Assignments rows.
`<VTI>`	Verbatim Term Individuals
`<VTI_ROW>`	Verbatim Term Individual Rows.
`<VTI_DEL>`	Deleted Verbatim Term Individuals.
`<VTI_DEL_ROW>`	Deleted Verbatim Term Individual Rows.

Table B–2   XML Data and Metadata Tags

Tag	Description	Data Type (length)	Mand-atory?	Default Value
`<INTEGRATION_KEY>`	Source Data System Name	Varchar2(15)	Y	N/A
`<INSTANCE_NAME>`	Source Database Name	Varchar2(93)	Y	N/A
`<X_AREA>`	Batch Processing Unit	Number(10)	Y	0
`<X_AREA_NAME>`	Name; e.g. `Study123`	Varchar2(65)	Y	N/A
`<GMT_DIFF>`	DB's GMT Time Difference	Varchar2(6)	N	N/A
`<DICTIONARY_SHORT_NAME>`	Dictionary Short Name	Varchar2(10)	Y	N/A
`<DOMAIN_NAME>`	Domain Name	Varchar2(65)	Y	N/A
`<VERBATIM_TERM>`	Collected Verbatim Term	Varchar2(300)	Y	N/A
`<VTA_GLOBAL_FLAG>`	Is the VTA Global or not?	Varchar2(1)	N	N/A
`<VTA_CREATED_BY>`	The VTA Creator	Varchar2(30)	N	N/A
`<VT_SUBTYPE>`	MS (Misspelled) or AC (Accepted)	Varchar2(15)	N	N/A
`VTI_CREATED_BY`	The VTI Creator	Varchar2(30)	N	N/A
`VT_TYPE`	VTA or VTI	Varchar2(15)	N	N/A
`DICT_CONTENT_CODE`	DICT_CONTENT_CODE of the Dictionary term	Varchar2(30)	N	N/A
`<DICTIONARY_TERM>`	Dictionary Term	Varchar2(300)	N	N/A

Table B–2   (Cont.) XML Data and Metadata Tags

Tag	Description	Data Type (length)	Mandatory?	Default Value
`<LABEL_TEXT_UPPER>`	Informative Note Label	Varchar2(30)	N	N/A
`<ALL_VERSION_FLAG>`	Does the Informative Note apply to all versions?	Varchar2(1)	N	N/A
`<CREATED_BY>`	Informative Note Creator	Varchar2(30)	N	N/A
`<DEF_DETAIL_VALUE>`	Informative Note Text	CLOB	N	N/A
`<STATUS>`	Informative Note Status	Varchar2(15)	N	N/A
`<SOURCE_TERM_ID>`	Source Data ID (PK)	Number(15)	Y	N/A
`<OCCURRENCE_ID>`	Source Data ID (PK)	Number(10)	Y	0
`<SOURCE_TERM_ALT_KEY>`	Alternative Source Data Key	Varchar2(60)	N	N/A
`<EXT_VALUE_1>`	User Defined Data Value	Varchar2(30)	N	N/A
`<EXT_VALUE_2>`	User Defined Data Value	Varchar2(30)	N	N/A
`<EXT_VALUE_3>`	User Defined Data Value	Varchar2(30)	N	N/A
`<EXT_VALUE_4>`	User Defined Data Value	Varchar2(30)	N	N/A
`<EXT_VALUE_5>`	User Defined Data Value	Varchar2(30)	N	N/A
`<EXT_VALUE_6>`	User Defined Data Value	Varchar2(30)	N	N/A
`<EXT_VALUE_7>`	User Defined Data Value	Varchar2(30)	N	N/A
`<EXT_VALUE_8>`	User Defined Data Value	Varchar2(30)	N	N/A
`<OMISSION_STATUS>`	Omission Status (`TMS_IN_PROGRESS`, `TMS_EVALUATION` are reserved)	Varchar2(15)	N	N/A
`<ACTION_OWNER>`	Person to whom the Action is assigned	Varchar2(30)	N	N/A
`<ACTION_TEXT>`	Action Text	Varchar2 (2000)	N	N/A
`<APPROVED_FLAG>`	VTA Approved Flag	Varchar2(1)	Y	N

Validation of Non-TMS XML Files

CROs using a system other than TMS for dictionary coding should validate their XML files before sending them to the sponsor. Sponsors working with CROs operating without TMS should provide them with the following information about how to do this:

1. Modify the XML file you have created for export to the sponsor so that its TMSDSI element (at the top of the file) references the .xsd file. (You must remove this reference when you send the XML file to the sponsor.) For example:

```
<TMSDSI

  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

 xsi:noNamespaceSchemaLocation="dsi_sponsor.xsd">
```

2. Put the .xsd file into the same directory as the XML file.

3. From that directory, if you are running on an Oracle database, invoke the Oracle tool oraxml to validate the XML file against the XML schema:

```
oraxml -schema xmlfilename.xml
```

This tool is included with Oracle Database. You must install the XML Toolkit. See the Oracle Database documentation for more information.

If you are not running on an Oracle database, you can find a tool on the Internet for verifying XML files against XML schemas.

4. If there are any errors, you must fix them. The sponsor cannot process the file if its XML structure is invalid for the TMS XML schema.

Example B–3   Sponsor Schema Validation Text

```
<?xml version="1.0" encoding="UTF-8"?>
 <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      TMS schema to validate DSI XML Files.
      Source data Import (Sponsor Extract)
      Copyright (c) 2004 Oracle Corporation. All rights reserved.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:element name="ACTION_OWNER" type="xsd:string"/>
  <xsd:element name="ACTION_TEXT" type="xsd:string"/>
  <xsd:element name="APPROVED_FLAG" type="xsd:string"/>
  <xsd:element name="CREATED_BY" type="xsd:string"/>
  <xsd:element name="DEF_DETAIL_VALUE"  type="xsd:string"/>
  <xsd:element name="DICTIONARY_SHORT_NAME" type="xsd:string"/>
  <xsd:element name="DICTIONARY_TERM" type="xsd:string"/>
  <xsd:element name="ERROR_ROW" type="xsd:string"/>
  <xsd:element name="DOMAIN_NAME" type="xsd:string"/>
  <xsd:element name="LABEL_TEXT_UPPER" type="xsd:string"/>
  <xsd:element name="EXT_VALUE_1" type="xsd:string"/>
  <xsd:element name="EXT_VALUE_2" type="xsd:string"/>
  <xsd:element name="EXT_VALUE_3" type="xsd:string"/>
  <xsd:element name="EXT_VALUE_4" type="xsd:string"/>
  <xsd:element name="EXT_VALUE_5" type="xsd:string"/>
  <xsd:element name="EXT_VALUE_6" type="xsd:string"/>
  <xsd:element name="EXT_VALUE_7" type="xsd:string"/>
  <xsd:element name="EXT_VALUE_8" type="xsd:string"/>
  <xsd:element name="GMT_DIFF" type="xsd:string"/>
  <xsd:element name="INSTANCE_NAME" type="xsd:string"/>
  <xsd:element name="INTEGRATION_KEY" type="xsd:string"/>
  <xsd:element name="OCCURRENCE_ID" type="xsd:integer"/>
  <xsd:element name="OMISSION_STATUS" type="xsd:string"/>
  <xsd:element name="SOURCE_TERM_ALT_KEY" type="xsd:string"/>
  <xsd:element name="SOURCE_TERM_ID" type="xsd:integer"/>
  <xsd:element name="STATUS"  type="xsd:string"/>
  <xsd:element name="VERBATIM_TERM" type="xsd:string"/>
  <xsd:element name="VT_SUBTYPE" type="xsd:string"/>
  <xsd:element name="VTA_CREATED_BY" type="xsd:string"/>
  <xsd:element name="VTI_CREATED_BY" type="xsd:string"/>
  <xsd:element name="DICT_CONTENT_CODE" type="xsd:string"/>
  <xsd:element name="VTA_GLOBAL_FLAG" type="xsd:string"/>
  <xsd:element name="X_AREA" type="xsd:integer"/>
  <xsd:element name="X_AREA_NAME" type="xsd:string"/>
  <xsd:element name="WARNING_ROW" type="xsd:string"/>
  <xsd:attribute name="LOCAL_FLAG" type="xsd:string"/>
  <xsd:element name="TMSDSI">
    <xsd:complexType>
      <xsd:sequence>
```

```
      <xsd:element name="META_DATA" type="MetaDataType"/>
      <xsd:element name="ACCOUNTS" type="AccountRowType" minOccurs="0"/>
      <xsd:element name="VTA_DEL" type="VtaDelType" minOccurs="0"/>
      <xsd:element name="VTI_DEL" type="VtiDelType" minOccurs="0"/>
      <xsd:element name="VTA" type="VtaType" minOccurs="0"/>
      <xsd:element name="VTI" type="VtiType" minOccurs="0"/>
      <xsd:element name="VT" type="VtType" minOccurs="0"/>
      <xsd:element name="SOURCE_DATA" type="SourceDataType" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:complexType name="MetaDataType">
  <xsd:sequence>
    <xsd:element name="INTEGRATION_KEY"/>
    <xsd:element name="INSTANCE_NAME"/>
    <xsd:element name="X_AREA"/>
    <xsd:element name="X_AREA_NAME"/>
    <xsd:element name="GMT_DIFF"/>
    <xsd:element name="ERROR" maxOccurs="unbounded" minOccurs="0"
type="ErrorRowType"/>
    <xsd:element name="WARNING" maxOccurs="unbounded" minOccurs="0"
type="WarningRowType"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="SourceDataRowType">
  <xsd:sequence maxOccurs="1" minOccurs="0">
    <xsd:element name="SOURCE_TERM_ID"/>
    <xsd:element name="OCCURRENCE_ID"/>
    <xsd:element name="OMISSION_STATUS" minOccurs="0"/>
    <xsd:element name="ACTION_TEXT" minOccurs="0"/>
    <xsd:element name="ACTION_OWNER" minOccurs="0">
      <xsd:complexType>
        <xsd:simpleContent>
          <xsd:extension base="xsd:string">
            <xsd:attribute ref="LOCAL_FLAG"/>
          </xsd:extension>
        </xsd:simpleContent>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="ERROR" maxOccurs="unbounded" minOccurs="0"
type="ErrorRowType"/>
    <xsd:element name="WARNING" maxOccurs="unbounded" minOccurs="0"
type="WarningRowType"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="SourceDataType">
  <xsd:sequence maxOccurs="unbounded" minOccurs="0">
    <xsd:element name="SOURCE_DATA_ROW" type="SourceDataRowType"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="VtaDelRowType">
  <xsd:sequence maxOccurs="1" minOccurs="0">
    <xsd:element name="DICTIONARY_SHORT_NAME"/>
    <xsd:element name="DOMAIN_NAME"/>
    <xsd:element name="VERBATIM_TERM"/>
    <xsd:element name="VTA_GLOBAL_FLAG"/>
    <xsd:element name="VTA_MODIFIED_BY">
      <xsd:complexType>
        <xsd:simpleContent>
          <xsd:extension base="xsd:string">
```

```
                    <xsd:attribute ref="LOCAL_FLAG"/>
                  </xsd:extension>
                </xsd:simpleContent>
              </xsd:complexType>
            </xsd:element>
            <xsd:element name="ERROR" maxOccurs="unbounded" minOccurs="0"
type="ErrorRowType"/>
            <xsd:element name="WARNING" maxOccurs="unbounded" minOccurs="0"
type="WarningRowType"/>
          </xsd:sequence>
        </xsd:complexType>
        <xsd:complexType name="VtiDelRowType">
          <xsd:sequence maxOccurs="1" minOccurs="0">
            <xsd:element name="DICTIONARY_SHORT_NAME"/>
            <xsd:element name="DOMAIN_NAME"/>
            <xsd:element name="VERBATIM_TERM"/>
            <xsd:element name="SOURCE_TERM_ID"/>
            <xsd:element name="ERROR" maxOccurs="unbounded" minOccurs="0"
type="ErrorRowType"/>
            <xsd:element name="WARNING" maxOccurs="unbounded" minOccurs="0"
type="WarningRowType"/>
          </xsd:sequence>
        </xsd:complexType>
        <xsd:complexType name="VtaDelType">
          <xsd:sequence maxOccurs="unbounded" minOccurs="0">
            <xsd:element name="VTA_DEL_ROW" type="VtaDelRowType"/>
          </xsd:sequence>
        </xsd:complexType>
        <xsd:complexType name="VtiDelType">
          <xsd:sequence maxOccurs="unbounded" minOccurs="0">
            <xsd:element name="VTI_DEL_ROW" type="VtiDelRowType"/>
          </xsd:sequence>
        </xsd:complexType>
        <xsd:complexType name="VtaRowType">
          <xsd:sequence minOccurs="0">
            <xsd:element name="DICTIONARY_SHORT_NAME"/>
            <xsd:element name="DOMAIN_NAME"/>
            <xsd:element name="VERBATIM_TERM"/>
            <xsd:element name="VT_SUBTYPE" minOccurs="0"/>
            <xsd:element name="DICTIONARY_TERM" minOccurs="0"/>
            <xsd:element name="APPROVED_FLAG" minOccurs="0"/>
            <xsd:element name="VTA_GLOBAL_FLAG" minOccurs="0"/>
            <xsd:element name="VTA_CREATED_BY" minOccurs="0">
              <xsd:complexType>
                <xsd:simpleContent>
                  <xsd:extension base="xsd:string">
                    <xsd:attribute ref="LOCAL_FLAG"/>
                  </xsd:extension>
                </xsd:simpleContent>
              </xsd:complexType>
            </xsd:element>
            <xsd:element name="ERROR" maxOccurs="unbounded" minOccurs="0"
type="ErrorRowType"/>
            <xsd:element name="WARNING" maxOccurs="unbounded" minOccurs="0"
type="WarningRowType"/>
          </xsd:sequence>
        </xsd:complexType>
        <xsd:complexType name="VtiRowType">
          <xsd:sequence minOccurs="0">
            <xsd:element name="DICTIONARY_SHORT_NAME"/>
```

```
        <xsd:element name="DOMAIN_NAME"/>
        <xsd:element name="VERBATIM_TERM"/>
        <xsd:element name="SOURCE_TERM_ID"/>
        <xsd:element name="OCCURRENCE_ID"/>
        <xsd:element name="VT_SUBTYPE" minOccurs="0"/>
        <xsd:element name="DICTIONARY_TERM" minOccurs="0"/>
        <xsd:element name="DICT_CONTENT_CODE" minOccurs="0"/>
        <xsd:element name="VTI_CREATED_BY" minOccurs="0">
          <xsd:complexType>
            <xsd:simpleContent>
              <xsd:extension base="xsd:string">
                <xsd:attribute ref="LOCAL_FLAG"/>
              </xsd:extension>
            </xsd:simpleContent>
          </xsd:complexType>
        </xsd:element>
        <xsd:element name="ERROR" maxOccurs="unbounded" minOccurs="0"
type="ErrorRowType"/>
        <xsd:element name="WARNING" maxOccurs="unbounded" minOccurs="0"
type="WarningRowType"/>
      </xsd:sequence>
    </xsd:complexType>
    <xsd:complexType name="VtaType">
      <xsd:sequence maxOccurs="unbounded" minOccurs="0">
        <xsd:element name="VTA_ROW" type="VtaRowType"/>
      </xsd:sequence>
    </xsd:complexType>
    <xsd:complexType name="VtiType">
      <xsd:sequence maxOccurs="unbounded" minOccurs="0">
        <xsd:element name="VTI_ROW" type="VtiRowType"/>
      </xsd:sequence>
    </xsd:complexType>
    <xsd:complexType name="VtNoteRowType">
      <xsd:sequence maxOccurs="unbounded" minOccurs="0">
        <xsd:element name="LABEL_TEXT_UPPER"/>
        <xsd:element name="OCCURRENCE_ID"/>
        <xsd:element name="DEF_DETAIL_VALUE" minOccurs="0"/>
        <xsd:element name="CREATED_BY">
          <xsd:complexType>
            <xsd:simpleContent>
              <xsd:extension base="xsd:string">
                <xsd:attribute ref="LOCAL_FLAG"/>
              </xsd:extension>
            </xsd:simpleContent>
          </xsd:complexType>
        </xsd:element>
        <xsd:element name="ERROR" maxOccurs="unbounded" minOccurs="0"
type="ErrorRowType"/>
        <xsd:element name="WARNING" maxOccurs="unbounded" minOccurs="0"
type="WarningRowType"/>
      </xsd:sequence>
    </xsd:complexType>
    <xsd:complexType name="VtNoteType">
      <xsd:sequence maxOccurs="unbounded" minOccurs="0">
        <xsd:element name="VT_NOTE_ROW" type="VtNoteRowType"/>
      </xsd:sequence>
    </xsd:complexType>
    <xsd:complexType name="VtRowType">
      <xsd:sequence  maxOccurs="1" minOccurs="1">
        <xsd:element name="DICTIONARY_SHORT_NAME"/>
```

```
        <xsd:element name="DOMAIN_NAME"/>
        <xsd:element name="VERBATIM_TERM"/>
        <xsd:element name="VT_NOTE" type="VtNoteType"/>
        <xsd:element name="ERROR" maxOccurs="unbounded" minOccurs="0"
type="ErrorRowType"/>
        <xsd:element name="WARNING" maxOccurs="unbounded" minOccurs="0"
type="WarningRowType"/>
      </xsd:sequence>
  </xsd:complexType>
  <xsd:complexType name="VtType">
    <xsd:sequence maxOccurs="unbounded" minOccurs="0">
      <xsd:element name="VT_ROW" type="VtRowType"/>
    </xsd:sequence>
  </xsd:complexType>
  <xsd:complexType name="ErrorRowType">
    <xsd:sequence minOccurs="0">
      <xsd:element name="ERROR_ROW"/>
    </xsd:sequence>
  </xsd:complexType>
  <xsd:complexType name="WarningRowType">
    <xsd:sequence minOccurs="0">
      <xsd:element name="WARNING_ROW"/>
    </xsd:sequence>
  </xsd:complexType>
  <xsd:complexType name="AccountRowType">
    <xsd:sequence maxOccurs="unbounded" minOccurs="0">
      <xsd:element name="OPA_ACCOUNT">
        <xsd:complexType mixed="true">
          <xsd:attribute name="FIRST_NAME" use="required"/>
          <xsd:attribute name="LAST_NAME" use="required"/>
          <xsd:attribute name="ENTRY_TS" use="required"/>
          <xsd:attribute name="END_TS" use="required"/>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:schema>
```

Example B–4   Source Schema Validation Text

```
<?xml version="1.0" encoding="UTF-8"?>
 <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      TMS schema to validate DSI XML Files.
      Source data Export (Sponsor Import)
      Copyright (c) 2004 Oracle Corporation. All rights reserved.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:element name="ACTION_OWNER" type="xsd:string"/>
  <xsd:element name="ACTION_TEXT" type="xsd:string"/>
  <xsd:element name="APPROVED_FLAG" type="xsd:string"/>
  <xsd:element name="CREATED_BY" type="xsd:string"/>
  <xsd:element name="DEF_DETAIL_VALUE"  type="xsd:string"/>
  <xsd:element name="DICTIONARY_SHORT_NAME" type="xsd:string"/>
  <xsd:element name="DICTIONARY_TERM" type="xsd:string"/>
  <xsd:element name="ERROR_ROW" type="xsd:string"/>
  <xsd:element name="DOMAIN_NAME" type="xsd:string"/>
  <xsd:element name="LABEL_TEXT_UPPER" type="xsd:string"/>
  <xsd:element name="EXT_VALUE_1" type="xsd:string"/>
```

```
<xsd:element name="EXT_VALUE_2" type="xsd:string"/>
<xsd:element name="EXT_VALUE_3" type="xsd:string"/>
<xsd:element name="EXT_VALUE_4" type="xsd:string"/>
<xsd:element name="EXT_VALUE_5" type="xsd:string"/>
<xsd:element name="EXT_VALUE_6" type="xsd:string"/>
<xsd:element name="EXT_VALUE_7" type="xsd:string"/>
<xsd:element name="EXT_VALUE_8" type="xsd:string"/>
<xsd:element name="GMT_DIFF" type="xsd:string"/>
<xsd:element name="INSTANCE_NAME" type="xsd:string"/>
<xsd:element name="INTEGRATION_KEY" type="xsd:string"/>
<xsd:element name="OCCURRENCE_ID" type="xsd:integer"/>
<xsd:element name="OMISSION_STATUS" type="xsd:string"/>
<xsd:element name="SOURCE_TERM_ALT_KEY" type="xsd:string"/>
<xsd:element name="SOURCE_TERM_ID" type="xsd:integer"/>
<xsd:element name="STATUS"  type="xsd:string"/>
<xsd:element name="VERBATIM_TERM" type="xsd:string"/>
<xsd:element name="VT_SUBTYPE" type="xsd:string"/>
<xsd:element name="VT_TYPE" type="xsd:string"/>
<xsd:element name="DICT_CONTENT_CODE" type="xsd:string"/>
<xsd:element name="VTA_CREATED_BY" type="xsd:string"/>
<xsd:element name="VTI_CREATED_BY" type="xsd:string"/>
<xsd:element name="VTA_GLOBAL_FLAG" type="xsd:string"/>
<xsd:element name="X_AREA" type="xsd:integer"/>
<xsd:element name="X_AREA_NAME" type="xsd:string"/>
<xsd:element name="WARNING_ROW" type="xsd:string"/>
<xsd:attribute name="LOCAL_FLAG" type="xsd:string"/>
<xsd:element name="TMSDSI">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="META_DATA" type="MetaDataType"/>
      <xsd:element name="ACCOUNTS" type="AccountRowType" minOccurs="0"/>
      <xsd:element name="VT" type="VtType" minOccurs="0"/>
      <xsd:element name="SOURCE_DATA_DEL" type="SourceDataDelType"
minOccurs="0"/>
      <xsd:element name="SOURCE_DATA" type="SourceDataType" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:complexType name="MetaDataType">
  <xsd:sequence>
    <xsd:element name="INTEGRATION_KEY"/>
    <xsd:element name="INSTANCE_NAME"/>
    <xsd:element name="X_AREA"/>
    <xsd:element name="X_AREA_NAME"/>
    <xsd:element name="GMT_DIFF"/>
    <xsd:element name="ERROR" maxOccurs="unbounded" minOccurs="0"
type="ErrorRowType"/>
    <xsd:element name="WARNING" maxOccurs="unbounded" minOccurs="0"
type="WarningRowType"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="SourceDataRowType">
  <xsd:sequence maxOccurs="1" minOccurs="1">
    <xsd:element name="DICTIONARY_SHORT_NAME"/>
    <xsd:element name="DOMAIN_NAME"/>
    <xsd:element name="VERBATIM_TERM"/>
    <xsd:element name="DICTIONARY_TERM" minOccurs="0"/>
    <xsd:element name="DICT_CONTENT_CODE" minOccurs="0"/>
    <xsd:element name="VTA_GLOBAL_FLAG" minOccurs="0"/>
    <xsd:element name="VTA_CREATED_BY" minOccurs="0">
```

```xml
                    <xsd:complexType>
                      <xsd:simpleContent>
                        <xsd:extension base="xsd:string">
                          <xsd:attribute ref="LOCAL_FLAG"/>
                        </xsd:extension>
                      </xsd:simpleContent>
                    </xsd:complexType>
                  </xsd:element>
                  <xsd:element name="VT_SUBTYPE" minOccurs="0"/>
                  <xsd:element name="VT_TYPE" minOccurs="0"/>
                  <xsd:element name="SOURCE_TERM_ID"/>
                  <xsd:element name="OCCURRENCE_ID"/>
                  <xsd:element name="SOURCE_TERM_ALT_KEY" minOccurs="0"/>
                  <xsd:element name="EXT_VALUE_1" minOccurs="0"/>
                  <xsd:element name="EXT_VALUE_2" minOccurs="0"/>
                  <xsd:element name="EXT_VALUE_3" minOccurs="0"/>
                  <xsd:element name="EXT_VALUE_4" minOccurs="0"/>
                  <xsd:element name="EXT_VALUE_5" minOccurs="0"/>
                  <xsd:element name="EXT_VALUE_6" minOccurs="0"/>
                  <xsd:element name="EXT_VALUE_7" minOccurs="0"/>
                  <xsd:element name="EXT_VALUE_8" minOccurs="0"/>
                  <xsd:element name="APPROVED_FLAG" minOccurs="0"/>
                  <xsd:element name="OMISSION_STATUS" minOccurs="0"/>
                  <xsd:element name="ACTION_TEXT" minOccurs="0"/>
                  <xsd:element name="ACTION_OWNER" minOccurs="0">
                    <xsd:complexType>
                      <xsd:simpleContent>
                        <xsd:extension base="xsd:string">
                          <xsd:attribute ref="LOCAL_FLAG"/>
                        </xsd:extension>
                      </xsd:simpleContent>
                    </xsd:complexType>
                  </xsd:element>
                  <xsd:element name="ERROR" maxOccurs="unbounded" minOccurs="0"
        type="ErrorRowType"/>
                  <xsd:element name="WARNING" maxOccurs="unbounded" minOccurs="0"
        type="WarningRowType"/>
                </xsd:sequence>
              </xsd:complexType>
              <xsd:complexType name="SourceDataType">
                <xsd:sequence maxOccurs="unbounded" minOccurs="0">
                  <xsd:element name="SOURCE_DATA_ROW" type="SourceDataRowType"/>
                </xsd:sequence>
              </xsd:complexType>
              <xsd:complexType name="SourceDataDelRowType">
                <xsd:sequence maxOccurs="1" minOccurs="1">
                  <xsd:element name="SOURCE_TERM_ID"/>
                  <xsd:element name="OCCURRENCE_ID"/>
                </xsd:sequence>
              </xsd:complexType>
              <xsd:complexType name="SourceDataDelType">
                <xsd:sequence maxOccurs="unbounded" minOccurs="0">
                  <xsd:element name="SOURCE_DATA_DEL_ROW" type="SourceDataDelRowType"/>
                </xsd:sequence>
              </xsd:complexType>
              <xsd:complexType name="VtNoteRowType">
                <xsd:sequence maxOccurs="unbounded" minOccurs="0">
                  <xsd:element name="LABEL_TEXT_UPPER"/>
                  <xsd:element name="OCCURRENCE_ID"/>
                  <xsd:element name="DEF_DETAIL_VALUE" minOccurs="0"/>
```

```
            <xsd:element name="CREATED_BY">
              <xsd:complexType>
                <xsd:simpleContent>
                  <xsd:extension base="xsd:string">
                    <xsd:attribute ref="LOCAL_FLAG"/>
                  </xsd:extension>
                </xsd:simpleContent>
              </xsd:complexType>
            </xsd:element>
            <xsd:element name="ERROR" maxOccurs="unbounded" minOccurs="0"
type="ErrorRowType"/>
            <xsd:element name="WARNING" maxOccurs="unbounded" minOccurs="0"
type="WarningRowType"/>
          </xsd:sequence>
        </xsd:complexType>
        <xsd:complexType name="VtNoteType">
          <xsd:sequence maxOccurs="unbounded" minOccurs="0">
            <xsd:element name="VT_NOTE_ROW" type="VtNoteRowType"/>
          </xsd:sequence>
        </xsd:complexType>
        <xsd:complexType name="VtRowType">
          <xsd:sequence  maxOccurs="1" minOccurs="1">
            <xsd:element name="DICTIONARY_SHORT_NAME"/>
            <xsd:element name="DOMAIN_NAME"/>
            <xsd:element name="VERBATIM_TERM"/>
            <xsd:element name="VT_NOTE" type="VtNoteType"/>
            <xsd:element name="ERROR" maxOccurs="unbounded" minOccurs="0"
type="ErrorRowType"/>
            <xsd:element name="WARNING" maxOccurs="unbounded" minOccurs="0"
type="WarningRowType"/>
          </xsd:sequence>
        </xsd:complexType>
        <xsd:complexType name="VtType">
          <xsd:sequence maxOccurs="unbounded" minOccurs="0">
            <xsd:element name="VT_ROW" type="VtRowType"/>
          </xsd:sequence>
        </xsd:complexType>
        <xsd:complexType name="ErrorRowType">
          <xsd:sequence minOccurs="0" maxOccurs="unbounded" >
            <xsd:element name="ERROR_ROW"/>
          </xsd:sequence>
        </xsd:complexType>
        <xsd:complexType name="WarningRowType">
          <xsd:sequence minOccurs="0" maxOccurs="unbounded" >
            <xsd:element name="WARNING_ROW"/>
          </xsd:sequence>
        </xsd:complexType>
        <xsd:complexType name="AccountRowType">
          <xsd:sequence minOccurs="0" maxOccurs="unbounded">
            <xsd:element name="OPA_ACCOUNT">
              <xsd:complexType mixed="true">
                <xsd:attribute name="FIRST_NAME" use="required"/>
                <xsd:attribute name="LAST_NAME" use="required"/>
                <xsd:attribute name="ENTRY_TS" use="required"/>
                <xsd:attribute name="END_TS" use="required"/>
              </xsd:complexType>
            </xsd:element>
          </xsd:sequence>
        </xsd:complexType>
</xsd:schema>
```

C

GUI Quick Reference

Most of the instructions in this guide describe how to perform Oracle Thesaurus Management System (TMS) functions by choosing menu paths in TMS windows. There are, however, many functions that you can execute in more than one way: in addition to the menu path, TMS provides a button or a keyboard shortcut for the same function.

Each quick reference section describes the functions that you can perform in TMS and, where applicable, the alternate methods you can use to execute that function. They are arranged in alphabetical order by function. Because TMS uses two sets of icons (one for standard TMS windows and another for batch job and report windows), there are two quick reference sections:

- Quick Reference for TMS Windows on page C-1

- Quick Reference for Batch Job and Reports Windows on page C-3

Only icons that are different from standard TMS windows are included in the latter section. For more information on the TMS graphic user interface, see Chapter 2, "Common Functions."

Quick Reference for TMS Windows

The button icons in the following table are enlarged for clarity.

Table C–1   Buttons in TMS Windows

Button	Function	Keyboard	Menu
None	Arrange icons	None	From Window menu, select **Arrange Icons**.
	Cancel query; select again to exit application (with warning)	Ctrl+Q (cancels query and exits TMS with warning)	From **Query** menu, select **Cancel Query**.
None	Cascade window	None	From **Window** menu, select **Cascade**.
	Clear record	Ctrl+U	From **Record** menu, select **Clear Record**.
None	Close all drill-down windows	None	From **Navigate** menu, select **Close All Drill Down**.
	Close current window	Ctrl+F4	From **File** menu, select **Close**.

Table C–1   (Cont.) Buttons in TMS Windows

Button	Function	Keyboard	Menu
	Collapse all	None	From **Navigate** menu, select **Collapse All**.
	Collapse node	None	From **Navigate** menu, select **Collapse Node**.
None	Copy	Ctrl+C	From **Edit** menu, select **Copy**.
None	Count query hits	Shift+F2	From **Query** menu, select **Count Hits**.
None	Cut	Ctrl+X	From **Edit** menu, select **Cut**.
None	Delete record	Shift+F6	From **Record** menu, select **Delete**.
None	Display error	Shift+F1	From **Help** menu, select **Display Error**.
	Drill down[1]	None	From **Navigate** menu, select **Drill Down**.
None	Duplicate record	F4	From **Record** menu, select **Duplicate**.
	Edit field	Ctrl+E	From **Edit** menu, select **Edit Field**.
	Execute query	F8	From **Query** menu, select **Execute Query**.
	Exit application (with warning) except in query mode where first click ends query, second click exits application	Alt+F4	From **File** menu, select **Exit**.
	Expand all	None	From **Navigate** menu, select **Expand All**.
	Expand node	None	From **Navigate** menu, select **Expand Node**.
	First record	None	From **Navigate** menu, select **First Record**.
	Help	F1	From **Help** menu, select **Help**.
	Insert record	F6	From **Record** menu, select **Insert**.
	Last record	None	From **Navigate** menu, select **Last Record**.
	List of values	F9	From **Edit** menu, select **List Values**.

Table C–1   (Cont.) Buttons in TMS Windows

Button	Function	Keyboard	Menu
	Menu	None	From **File** menu, select Menu.
	Multi/single record display (toggle)	None	From **Navigate** menu, select **Single/Multi Record**.
	Next record	Shift+Down arrow	From **Navigate** menu, select **Next Record**.
	Open query	F7	From **Query** menu, select **Open Query**.
None	Paste	Ctrl+V	From **Edit** menu, select **Paste**.
	Previous record	Shift+Up arrow	From **Navigate** menu, select **Previous**.
	Print	Shift+F8	From **File** menu, select **Print**.
None	Print contents[2]	None	From **File** menu, select **Print Contents**.
None	Refresh tree	None	From **Navigate** menu, select **Refresh Tree**.
	Save	F10	From **File** menu, select **Save**.
None	Show environment	None	From **Help** menu, select **Show Environment**.
None	Show keys	Ctrl+F1	From **Help** menu, select **Show Keys**.
	Single/multi record display (toggle)	None	From **Navigate** menu, select **Single/Multi Record**.
None	Tile Horizontally	None	From **Window** menu, select **Tile Horizontally**.
None	Tile Vertically	None	From **Window** menu, select **Tile Vertically**.

[1]   When drill-down information is available, TMS displays field text in blue.
[2]   The Print Contents option has no function in TMS.

Quick Reference for Batch Job and Reports Windows

The button icons in the following table are enlarged for clarity.

Table C–2   Buttons in the Batch Job and Reports Windows

Button	Function	Keyboard	Menu
	Batch job status	None	From **Job** menu, select **Status**.

Table C–2   (Cont.) Buttons in the Batch Job and Reports Windows

Button	Function	Keyboard	Menu
	Collapse all	None	From **Navigate** menu, select **Collapse All**.
	Collapse node	None	From **Navigate** menu, select **Collapse Node**.
	Connect to database	None	From **File** menu, select **Connect**.
	Expand all	None	From **Navigate** menu, select **Expand All**.
	Expand node	None	From **Navigate** menu, select **Expand Node**.
	Retrieve parameter set	None	From **Job** menu, select **Retrieve Parameter Set**.
	Save parameter set	None	From **Job** menu, select **Save Parameter Set**.
	Submit job	None	From **Job** menu, select **Submit**.

Glossary

action

Most commonly, a user-defined message that Oracle Thesaurus Management System (TMS) returns to the external system; see answerable action and Unanswerable Action. In addition, Internal Actions are used for approving external action assignments or for internal communication.

action type

The manner in which TMS assigns an Action; a answerable action is assigned at the first instance of an omission but not in subsequent appearances of an omission, a Unanswerable Action is assigned at every instance of the specified omission, and a discrepancy message is assigned to a specific TMS omission and applies to that verbatim term occurrence only.

activation group

A group of terms that TMS will activate during the same batch job. The group is defined in the Define Activation Groups window, and used during loading, updating, and modifying dictionaries.

answerable action

A message that a user applies to an unprocessable verbatim term requesting a change to the term in the external source data system. TMS sends the action with the verbatim term to the external system. A user in the external system can either change the term as requested or send a message back to TMS that is visible in TMS.

approved VTA

A VTA (verbatim term assignment) that TMS can use for Autoclassification.

autoclassification

Autocoding: the process of searching for exact matches to each new verbatim term occurrence (source term) among both VTAs and dictionary terms in the current domain.

automatic login

A configuration for the HTML Browser that enables users to connect without specifying a user name and password.

base dictionary

A dictionary to which you can add, modify, or delete terms and unapproved VTAs (unlike a virtual dictionary). Base dictionaries are useful for studies that require the most up-to-date dictionary information.

candidate term

When a verbatim term matches a verbatim term previously classified with a Nonapproved VTA, TMS proposes the dictionary term linked by the Nonapproved VTA as the Candidate Term. If another occurrence of the verbatim term occurs in the same domain, TMS allows the user only to accept or reject the Candidate Term. If another occurrence of the verbatim term occurs in a different domain, the user can accept the Candidate Term or manually classify the verbatim term to a different dictionary term. If the verbatim term is linked to different dictionary terms in different domains, and the supplied and predefined *domain match* search object is linked to the dictionary, TMS presents the Candidate Terms that exist in other domains.

cardinality

A characteristic of a relation between two dictionary levels; determines the maximum number of terms (one or many) in a level to which a term in another level may be linked. In TMS this characteristic is represented in the tree structure by the line between related levels: a single line symbolizes Single Cardinality and a branched line symbolizes Many Cardinality.

classify

Create a verbatim term assignment between an omission and a dictionary term.

classification

The process of assigning a VTA to a verbatim term. Depending on the verbatim term, the definitions in the dictionary domain, and the type of searches selected for the dictionary, TMS may accomplish classification automatically or may return an omission requiring user input. Only users with the TMS role tms_classify_priv may perform classification.

classification level

The dictionary level defined as the level in which TMS should search for matches to verbatim terms; verbatim term code level.

company relation

A relation defined by the user and available to all domains.

company term

A term defined by the user and available to all domains.

cross-dictionary link

A relationship that you must define between two dictionaries before you can create relations from a term in one dictionary to a term in the other.

current level

The dictionary level that is selected in a TMS window.

current term

The term that is selected in a TMS window.

cut-off date

The date after which no dictionary terms may be added to a virtual dictionary.

declassification

Deletion of a verbatim term assignment. Declassifying a source term creates an omission.

derivation path

A series of related levels defined as derivable that begins at the classification level and includes all levels from which you want to derive terms. There must be one and only one path from the verbatim term level to derivable levels. Defining levels as derivable makes them part of the derivation path. To actually derive values from derivable levels, you must also define them as reportable.

derived question

The data collection unit set up in the external system to receive the term or other value TMS returns to the external system after classification.

dictionary

A set of terms and relations; usually including a vendor-supplied (*external*) dictionary as its core, plus user-defined company and domain terms.

dictionary data

Terms and relations associated with a particular dictionary, including external terms supplied by a vendor and user-defined company and domain terms.

dictionary term

An external, company, or domain term associated with a particular dictionary, such as MedDRA or WHO-Drug.

dictionary term display procedure

A user-defined procedure used in omission management to show a different term in the derivation path than the classification level term. Selected on a dictionary-wide basis.

discrepancy

In Oracle Clinical or other external systems, an entry into the Discrepancy Management System created by Batch Validation when (in the case of thesaurus discrepancies) a verbatim term does not match a dictionary term or an Approved VTA; if the appropriate flag is set in the external system, requires resolution in TMS. See also omission.

discrepancy message

An Action that you create and assign to a single occurrence of a verbatim term, sent one time only to the external system.

domain

In TMS, a logical group of data assigned for use in a particular study or other work unit, including external dictionaries, company and domain terms, and VTAs.

domain element

The link between an Oracle Clinical study or project and the TMS dictionary and domain it uses for reference.

domain relation

A relation defined by the user and available only within the domain in which it was created.

domain term

A term defined by the user and available only within the domain in which it was created.

Domain VTA

A verbatim term assignment available only within the domain in which it was created.

dynamic dictionary

A dictionary in which the terms do not belong to specified dictionary levels. In TMS, named relations define the linking between terms in dynamic dictionaries.

external dictionary

A vendor-supplied dictionary; a dictionary maintained and released by an organization external to the TMS customer company.

external system

An application that is using TMS as its dictionary repository; for example, Oracle Clinical.

external term

Term supplied by a vendor in an external dictionary such as MedDRA or WHO-Drug.

global language

The primary language in which a study or project is conducted. When you specify Translation Derivation Links between dictionaries, you must choose which dictionary is in the global language and which is in the local language.

Global VTA

A verbatim term assignment (VTA) available to all domains.

group level

A dictionary level that contains other levels as sublevels within it; superlevel.

HTML Browser

A browser-based tool for viewing and searching for TMS Repository data and source data in a read-only format.

HTML layouts

Sets of columns that you can use in the HTML Browser to customize which data is presented in the results for terminology and verbatim term searches. HTML layouts are dictionary-specific.

indication assignment

An indication omission that has been classified; a high-level classification based on the indication for which a drug was taken.

indication omission

Different instances of the same dictionary coding-level term may be classified to different high level terms based on the indication for which the drug was taken by a

patient. In this case, multiple high-level terms are related to a single lower-level dictionary term with no primary path defined. TMS cannot derive the high-level term and creates an *indication omission*. After a user manually classifies the indication omission, TMS derives all current and future occurrences of the same term for the same verbatim term/indication combination.

Informative Note

Data structures that provide more information about one term, relation, VTA, or dictionary as a whole, in the database.

Informative Note Attribute

Database-wide definitions for the Data Type, Attribute Type, and other properties of an Informative Note. All Informative Notes are based upon an Informative Note Attribute definition.

installation

A set of locations that share data.

Internal Action

An Action that is assigned to a verbatim term for the purpose of approving the assignment of the associated base Answerable Action to the same verbatim term. When the Internal Action is approved, TMS sends the base Answerable Action, with any text modifications made by the approver, to the external system. Internal Actions can also be used to communicate internally with other TMS users.

Light Browser

See *HTML Browser*.

list of values

(LOV, <LIST> function) A list of acceptable values for a field, populated by the system into a list or a shortcut window. Access the LOV by clicking on the arrow of the list, an ellipsis (**…**) button, or by pressing F9 while your cursor is in the field.

local language

The secondary language in which a study or project is conducted. When you specify Translation Derivation Links between dictionaries, you must choose which dictionary is in the global language and which is in the local language.

location

A database that shares data with other databases through replication. This can refer to a separate database on the same computer or to a database in a different physical location.

LOV

See list of values.

materialized view

View that provides indirect access to table data by storing the results of a query in a separate schema object. Unlike an ordinary view, which does not take up any storage space or contain any data, a materialized view contains the rows resulting from a query against one or more base tables or views. A materialized view can be stored in the same database as its base table(s) or in a different database.

more information on materialized views is available in the *Oracle9i Database Concepts* manual (part number A96524).

Misspelled VTA

A verbatim term assignment that maps an incorrectly spelled verbatim term to an appropriate dictionary term.

named relation

A type of relation that defines the connection between terms. Named relations are defined on an installation-wide basis, and are used primarily in weak dictionaries.

Nonapproved VTA

A VTA, or classification of a verbatim term to a dictionary term, that has been created but not approved. The VTA must be manually applied to subsequent occurrences of the same verbatim term in the same domain. During manual classification, the system proposes the dictionary term as a candidate term.

occurrence

A single instance of a verbatim term from the external system; for example, a response of a particular patient at a particular visit.

omission

A verbatim term without a link to a dictionary term; if the appropriate flag is set in the external system, requires resolution by TMS. See also *discrepancy*.

parent question

In Oracle Clinical, the question that collects the response, or verbatim term, sent to TMS for classification.

pre-dictionary data

See *preliminary data*.

predict tables

Pre-dictionary tables; TMS tables that hold terms and relations awaiting activation, and terms and relations that have failed activation.

preliminary data

Data in the predict tables; terms and relations awaiting activation, and terms and relations that have failed activation.

preliminary tables

See *predict tables*.

Primary Link

Defines which single term is considered the predominant relation in a higher dictionary level where more than one term may be related to a single term in a lower level; defines the one term in the multiple-term level that should be derived or used for derivation farther up in the dictionary hierarchy.

Primary Path

Defines the derivation path between terms across any number of defined relations. Primary Path Links enable you to specify a Primary Path between a top level term and

the coding-level terms without defining a series of Primary Links or Paths to bridge the levels in between.

production tables

TMS tables that hold active terms and relations.

profile

A catalog of TMS settings values.

Question Set

In Oracle Clinical, defines the data derived from TMS to Oracle Clinical and associates a parent question with one or more derived questions.

reclassification

Changing a verbatim term assignment so that the verbatim term relates to a different dictionary term.

Release Label

The unique label that TMS applies to a particular build, or activation, of a dictionary. The Release Label consists of the Release Label Prefix followed by the build number.

replication

Creation of a read-only copy of specified objects from one database to another.

reportable level

A dictionary level whose terms can be reported to the external system as derived terms. You can report a level's terms only if the level's relation with its child is derivable and the external system is set up to call for derived terms.

search object

Set of algorithms used to search the TMS Repository in order to find matches for verbatim terms during Autoclassification, to find Candidate Terms during manual classification, or to find Repository terms in the Extended Search window.

source term

A term collected in an external system and sent to TMS for classification. A single occurrence of a verbatim term. In Oracle Clinical, a question response.

strong dictionary

A dictionary in which all of the terms are stored in dictionary levels.

sublevel

One of two or more levels grouped into a superlevel.

superlevel

A level containing other levels as sublevels within it; group level.

temporary tables

Tables that hold data from vendor-supplied ASCII files before it is loaded into the preliminary tables.

TMS domain

See *domain*.

Translation Derivation Link

A connection between two dictionaries with the same data and dictionary structure, which enables you, given a term in one dictionary, to find the corresponding record in the other. These links serve to translate dictionary data in studies being conducted in two different languages.

Unanswerable Action

A message that a user applies to an unprocessable verbatim term. TMS sends the action with the verbatim term to the external system. Any messages sent by a user in the external system to TMS are not visible in TMS.

upgrading (dictionary)

Changing an external dictionary's contents when a new version is released.

user profile

The set of profiles that have been assigned to a TMS user, and the hierarchy of Search Orders that enable TMS to determine which profiles to use for defining default values.

verbatim term

Free-form text initially entered or loaded into the external system; source term; external system question response.

verbatim term assignment (VTA)

The mapping between a verbatim term and a dictionary term in a dictionary whose classification level contains unique terms. VTAs can be created either by TMS during Autoclassification, if a direct match or Approved VTA exists, or by a user during manual classification.

verbatim term individual (VTI)

The mapping between a verbatim term and a dictionary term in a dictionary whose classification level contains non-unique terms, created either by TMS during Autoclassification, if a VTI exists for a verbatim term with the same auxiliary information, or by a user during manual classification.

verbatim term code level (VT code level)

See *classification level*.

virtual dictionary

A dictionary containing all the terms and assignments for a base dictionary at a specified point in time, the *cut-off date*. Virtual dictionaries enable you to continue coding verbatim terms for a clinical study against one version of a dictionary while the base dictionary changes.

VT

See *verbatim term*.

VTA

See *verbatim term assignment*.

VTI

See *verbatim term individual*.

VT code level

See *classification level*.

weak dictionary folder

A TMS dictionary that contains one or more dynamic dictionaries. Each level of a weak dictionary folder can be a fully defined dictionary.

X Area

The external system unit used in data exchange with TMS. If the external system is Oracle Clinical, an X Area corresponds to a clinical study, identified by the clinical study ID. For other external systems, the X Area should be a unique number that defines the external system batch processing unit.

Index