Oracle TimesTen In-Memory Database Installation Guide

Release 6.0

B25264-03

For last-minute updates, see the TimesTen release notes.

March 2006
Printed in the United States of America

Contents

2 TimesTen Installation

3 Data Store Upgrades

Index

About this Guide

This document contains all necessary information for installing the Oracle TimesTen® In-Memory Database (TimesTen) Data Manager, Client and Server components.

The TimesTen CD contains a `README.TXT` file that holds the release notes. These notes list product information and late changes to the printed documentation. The release notes are also available in PDF format. The PDF file is named `README.pdf`.

TimesTen documentation

Including this guide, the TimesTen documentation set consists of these documents:

- The *Oracle TimesTen In-Memory Database Operations Guide* provides information on configuring TimesTen and using the ttIsql utility to manage a data store. This guide also provides a basic tutorial for TimesTen.
- The *Oracle TimesTen In-Memory Database Architectural Overview* provides a description of all the available features in TimesTen.
- The *Oracle TimesTen In-Memory Database C Developer's and Reference Guide* and the *Oracle TimesTen In-Memory Database Java Developer's and Reference Guide* provide information on how to use the full set of available features in TimesTen to develop and implement applications that use TimesTen.
- The *Oracle TimesTen In-Memory Database API and SQL Reference Guide* contains a complete reference to all TimesTen utilities, procedures, APIs and other features of TimesTen.
- The *TimesTen to TimesTen Replication Guide*. This guide is for application developers who use and administer TimesTen and for system administrators who configure and manage TimesTen Replication. It provides: Background information to help you understand how TimesTen Replication works. Step-by-step instruction and examples that show how to perform the most commonly needed tasks.
- The *TimesTen Cache Connect to Oracle Guide* describes how to use Cache Connect to Oracle to cache Oracle data in TimesTen. This guide is for developers who use and administer TimesTen for caching Oracle data. It provides information on caching Oracle data in TimesTen data stores. It also describes how to use the Cache

Connect Administrator, a web-based interface for creating cache groups.

TimesTen documentation is available on the product CD-ROM and on the TimesTen web site: `http://www.timesten.com`.

Background reading

For a conceptual overview and JDBC development information, see:

- Hamilton, Cattell, Fisher. *JDBC Database Access with Java*. Reading, MA: Addison Wesley. 1998.

For a Java reference, see:

- Horstmann, Cornell. *Core Java*. Palo Alto, CA: Sun Microsystems Press. 1999.
- For the JDBC API specification, refer to java.sql package in the appropriate Java Platform API Specification.
- If you are working with JDK 1.2, refer to the Java 2 Platform API specification at: `http://java.sun.com/products/jdk/1.2/docs/api/index.html`
- If you are working with JDK 1.3, refer to the Java 2 Platform API specification at: `http://java.sun.com/j2se/1.3/docs/api/index.html`
- If you are working with JDK 1.4, refer to the Java 2 Platform API specification at: `http://java.sun.com/j2se/1.4/docs/api/index.html`
- Siple, Matthew. *The Complete Guide to Java Database Programming: JDBC, ODBC and SQL*. McGraw-Hill. 1997.

An extensive list of books about ODBC and SQL is in the Microsoft ODBC manual included in your developer's kit. In addition to this guide, your developer's kit includes the appropriate ODBC manual for your platform:

– *Microsoft ODBC 3.0 Programmer's Reference and SDK Guide* provides all relevant information on ODBC for Windows developers.

– *Microsoft ODBC 2.0 Programmer's Reference and SDK Guide*, included online in PDF format, provides information on ODBC for UNIX developers.

For a conceptual overview and programming how-to of ODBC, see:

- Sanders, Roger E. *ODBC 3.5 Developer's Guide* (McGraw-Hill Series On Data Warehousing and Data Management); McGraw-Hill. 1999
- Signore, Robert / Stegman, Michael O. / et al. *ODBC Solution: Open Database Connectivity in Distributed Environments*: *Mcgraw-hill Series On Computer Communications*; McGraw Hill. 1995

For a review of SQL, see:

- Melton, Jim and Simon, Alan R. *Understanding the New SQL: A Complete Guide*. San Francisco, CA: Morgan Kaufmann Publishers. 1993.
- Groff, James R. / Weinberg, Paul N. *SQL: The Complete Reference.* McGraw-Hill. /1999

For information on Unicode, see:

- The Unicode Consortium, *The Unicode Standard, Version 4.0*, Addison-Wesley, 2003.
- The Unicode Consortium Home Page at `http://www.unicode.org`

Conventions used in this guide

TimesTen supports multiple platforms. Unless otherwise indicated, the information in this guide applies to all supported platforms. The term Windows refers to Windows 2000, Windows XP and Windows Server 2003. The term UNIX refers to Solaris, Linux, HP-UX, Tru64 and AIX.

TimesTen documentation uses these typographical conventions:

If you see...	It means...
`code font`	Code examples, filenames, and pathnames. For example, the `.odbc.ini`.`ttconnect.ini` file.
`italic code font`	A variable in a code example that you must replace. For example: `Driver=`*`install_dir`*`/lib/libtten.sl` Replace *`install_dir`* with the path of your TimesTen installation directory.

TimesTen documentation uses these conventions in command line examples and descriptions:

If you see...	It means...
fixed width italics	Variable; must be replaced
[ ]	Square brackets indicate that an item in a command line is optional.
{ }	Curly braces indicated that you must choose one of the items separated by a vertical bar (|) in a command line.
|	A vertical bar (or pipe) separates arguments that you may use more than one argument on a single command line.
...	An ellipsis (. . .) after an argument indicates that you may use more than one argument on a single command line.
%	The percent sign indicates the UNIX shell prompt.
#	The number (or pound) sign indicates the UNIX root prompt.

TimesTen documentation uses these variables to identify path, file and user names:

If you see...	It means...
install_dir	The path that represents the directory where the current release of TimesTen is installed.
TTinstance	The instance name for your specific installation of TimesTen. Each installation of TimesTen must be identified at install time with a unique alphanumeric instance name. This name appears in the install path. The instance name “giraffe” is used in examples in this guide.
bits or *bb*	Two digits, either 32 or 64, that represent either the 32-bit or 64-bit operating system.
release or rr	Two digits that represent the first two digits of the current TimesTen release number, with or without a dot. For example, 60 or 5.0 represents TimesTen Release 5.0.
jdk_version	Two digits that represent the version number of the major JDK release. For example 12 for versions of jdk1.2.

`timesten`	A sample name for the TimesTen instance administrator. You can use any legal user name as the TimesTen administrator. On Windows the TimesTen instance administrator must be a member of the Administrators group. Each TimesTen instance can have a unique instance administrator name.
DSN	The data source name.

Technical Support

For information about obtaining technical support for TimesTen products, go to the following Web address:

http://www.oracle.com/support/contact.html

Email: timesten-support_us@oracle.com

1

Access Control and non-root installations

Introduction

Non-root installations

In previous releases of TimesTen, only the user `root` could install TimesTen. This release of TimesTen allows a user to optionally install TimesTen as a user other than `root` on UNIX systems. This feature is not available on Windows systems.

Access Control

In addition, this release of TimesTen optionally allows you to install TimesTen with a layer of security, which throughout the TimesTen documentation set and in the installation scripts is referred to as "Access Control."

The Access Control feature of TimesTen provides an environment of basic control for applications that use the defined privileges. Access Control does not provide definitive security for all processes that might be able to access the data store. For example, this feature does not protect the data store from user processes that may have sufficient privileges to connect to the data store when in memory or that can access files on disk that are associated with the data store, such as log files and checkpoint files.

In TimesTen, privileges are granted on a instance wide-basis. Privileges apply to all data stores in a given TimesTen instance or installation.

Your options for installing TimesTen, now include the following:

Non-root user	Root User
Access Control enabled	Access Control enabled
Access Control not enabled	Access Control not enabled

Limitations of Access Control and non-root installs

General

Regardless of whether you install TimesTen as `root` or as a non-root user, you must decide whether to enable Access Control at the time you install TimesTen. It cannot be enabled or disabled after installation.

If Access Control is to be enabled, installation itself must be performed by the chosen instance administrator user. The instance administrator owns all files in the installation directory tree. Only the instance administrator is allowed to install and administer the TimesTen instance. See "TimesTen instance administrator" on page 10. All TimesTen daemon processes are owned by the instance administrator.

Prior to installing TimesTen as non-root, certain tasks must be performed by the user `root`. Those tasks are outlined in "Prerequisites for non-root installations and Access Control on UNIX systems" on page 38.

Cache Connect

For Cache Connect, the TimesTen internal user must match the Oracle user. External Client/Server users must match the Oracle user. If you are using the Cache Connect Administrator interface, the user must be an internal TimesTen user.

Replication

If Access Control is enabled, replication daemon administration and replication schema changes are restricted to users having the ADMIN privilege. See "Privileges" on page 14.

Changes applied to a replication subscriber are made regardless of access controls present on the subscriber.

Instance user configuration commands are not replicated.

Client/Server

Use of Access Control requires that the Client/Server **Authenticate** attribute be turned on.

To use Access Control with Client/Server applications, when the user is identified externally, the Client and the Server must be on the same machine. Remote Client/Server access is only supported by TimesTen when Access Control is enabled, if the user is a TimesTen internal user.

TimesTen ignores the values of UID, PWD and PWDCrypt if specified in the Server DSN. The user name and password must be explicitly declared on the Client side.

If PWD or PWDCrypt is specified in Client/Server applications, TimesTen assumes that the user is internally identified, otherwise TimesTen assumes that the user is externally identified and authenticated by the operating system.

Migrating from previous releases

If you do not enable Access Control and you install as user `root`, you do not need to make any changes to your application or the installation and administration of TimesTen to use this release.

If you plan to install as a non-root user, but do not plan to enable Access Control, see "Prerequisites for non-root installations and Access Control on UNIX systems" on page 38.

If you plan to install with Access Control enabled, you will need to change your application to support authentication of users for specific operations. See "Authenticating users and privileges" on page 13. For details on each operation, see the *Oracle TimesTen In-Memory Database API and SQL Reference Guide* and the *Oracle TimesTen In-Memory Database Operations Guide*.

Root access

Instance startup/shutdown

Permission to start and stop the main TimesTen daemon is restricted to the TimesTen instance administrator. To provide a uniform interface for instance operation, the following new functionality has been added to the **ttDaemonAdmin** utility:

To start the TimesTen main daemon:

```
% ttDaemonAdmin -start
```

To stop an instance:

```
% ttDaemonAdmin -stop
```

Instance data store

A DSN for a minimal instance-wide data store is defined by TimesTen at install time to guarantee that TimesTen always has something with which to connect. The instance DSN looks like:

```
DSN=TT_instance
DataStore=$TIMESTEN_INFODIR/TT_instance
Driver=$TIMESTEN_DIR/lib/libtten.so
```

This data store gets special treatment from the daemon, and has special access restrictions placed on it. Regular users may connect to the instance data store, but are restricted to SELECT privileges, except that any user can change their own password.

TimesTen users

TimesTen instance administrator

The owner of a TimesTen installation is the "TimesTen instance administrator." Only a member of the TimesTen administrators group can install and administer TimesTen. The user installing the instance automatically becomes the administrator for that instance. Only that user may start or stop the instance, and only that user may administer the users in the instance. Use of the GroupRestrict attribute is required for all data stores in an instance using Access Control. Therefore the instance administrator must have corresponding group membership.

Though a specific user name is not required, we recommend the name "timesten." All examples in the TimesTen documentation set use the name `timesten` to represent the instance administrator.

For details on establishing the TimesTen instance administrators group, see "Create the TimesTen instance administrators group" on page 38.

On Windows systems, the user `System` automatically becomes the TimesTen instance administrator when Access Control is selected at install time.

On UNIX systems, the user `root` becomes the TimesTen instance administrator when Access Control is selected at install time and the installation is performed by user `root`.

TimesTen instance users

TimesTen instance users are user names that have been identified to the instance. They are defined at the instance level and apply to all data stores in the instance. Initially, only one user name is known to the instance: the instance administrator.

Only the instance administrator has permission to create or delete users. Individual users have permission to change their own passwords.

Instance users may have internal user names or external user names.

Internal user

A user name that has been defined within the TimesTen instance is referred to as an “internal user.” It has no significance outside of the particular instance of TimesTen in which it was defined. Internal users are authenticated by the TimesTen instance. See "CREATE USER"in the *Oracle TimesTen In-Memory Database API and SQL Reference Guide*. TimesTen user names (ODBC UID attributes) are automatically converted to upper case (case insensitive).

External user

A user name that is identified by the operating system or some other external mechanism is referred to as an “external user.” In this release only the operating system user name is recognized as an external user. External users are assumed to have been authenticated by their native mechanism. See "CREATE USER"in the *Oracle TimesTen In-Memory Database API and SQL Reference Guide*. A password is not required by TimesTen since the user was authenticated by the operating system at login time.

UNIX external user names are case sensitive. Windows external user names are not. When connecting from UNIX platforms, TimesTen automatically converts the external user name to upper case, rendering it case insensitive.

The **PWDCrypt** attribute provides a way to deal with the special characters and case sensitivity used in passwords that might create difficulties if specified in cleartext PWD attributes.

Before installation

Several steps must be taken to prepare a machine for TimesTen installation. These steps are needed once per machine and require root permission. As in previous releases, kernel parameters must be adjusted.See “Installation prerequisites” on page 22. This release of TimesTen requires that additional steps be performed prior to installation if either Access Control is to be enabled or you plan to install as non-root.

TimesTen administrators group

An operating system group needs to be defined for those users who will be allowed to install and administer TimesTen instances. This can be an

existing group, but we suggest that a group named “timesten” be created specifically for this purpose. “Create the TimesTen instance administrators group” on page 38. The member of the TimesTen administrators group who installs TimesTen becomes the TimesTen instance administrator.

Instance registry directory

TimesTen maintains a “registry” of all TimesTen instances installed on a given machine. The instance registry itself is not required for operation. It is only accessed during installation and uninstallation. TimesTen patches the information it contains into the libraries and executables of the instances, where it becomes an integral part of operation. The “registry” is only accessible by the TimesTen installation scripts. It is not accessible by TimesTen users or the individual who is installing TimesTen.

On Unix platforms, the instance registry is located in `/etc/TimesTen`. Initial creation of the `/etc/TimesTen` directory requires root access. Creation of this directory is a once per machine, pre-installation step. See “Create the TimesTen registry” on page 39. The disk space required for the files in this directory is less than 2k bytes.

On Windows the instance registry is contained in the operating system registry. No action is required by users or the individual who is installing TimesTen.

Installation directories, files and the daemon port

Installation itself must be performed by the chosen instance administrator user. The instance administrator owns all files in the installation directory tree. Only the instance administrator will be allowed to operate the instance.

Installation directories

The installer suggests default destination directories, based on the user performing the installation.

Instance home directory

The instance may be installed in any directory to which the instance administrator has sufficient permission.

On Unix, the installer suggests `/opt/TimesTen/tt51` as in previous releases.

On Windows, the installer suggests the directory pattern as used in previous releases of TimesTen, `C:\TimesTen\tt51`

The TimesTen documentation refers to the directory as *`install_dir`*.

Daemon home directory

The "home" or current working directory of the running daemon is known as the daemon home directory. This directory must be owned by the instance administrator, with `rwxr-xr-x` permissions. The daemon verifies this when it starts up. This directory can be anywhere on a local drive, but cannot be on an NFS mounted file system.

On UNIX, the installer suggests the use of *`install_dir`*`/info` if installed as non-root or `/var/TimesTen/tt51` as in previous releases if running as root.

On Windows, the *`install_dir`*`\srv\info` directory is used for this purpose, just as in previous releases.

Password file

If access control is selected at installation time, user and password data is stored in the file *`install_dir`*`/info/ttpasswd`.

Initially, this file contains a single entry for the instance administrator. The presence of this file indicates to the daemon that Access Control has been selected. If this file is missing, once Access Control has been defined, an error occurs.

This file is readable and writable only by the instance administrator. Passwords are stored in encrypted form and are not known to the instance administrator.

Daemon port

Though the instance registry enforces port uniqueness for TimesTen instances, note that the possibility of the TimesTen main daemon port conflicting with ports used by non-TimesTen applications always exists. See "Changing the daemon port number on UNIX" on page 40.

Authenticating users and privileges

Certain TimesTen utility APIs, XLA operations, utilities, procedures and SQL operations require user authentication to continue when Access Control is enabled. For details on each operation, see the specific chapters of the *Oracle TimesTen In-Memory Database API and*

SQL Reference Guide and the *Oracle TimesTen In-Memory Database Operations Guide*.

All TimesTen utilities prompt for a password if needed. See Chapter 2, "Utilities" in the *Oracle TimesTen In-Memory Database API and SQL Reference Guide*.

Client/Server utilities always prompt for a password if no PWD attribute is specified, since they must always use **Authenticate**.

Scripts built on utilities requiring passwords may want to use the **PWDCrypt** attribute.

Privileges

For a description of the TimesTen Access Control privileges, see "Access Control Privileges" in the *Oracle TimesTen In-Memory Database API and SQL Reference Guide*,

GroupRestrict

If access control is in use, the **GroupRestrict** attribute must be enabled for all data stores in the instance. The instance administrator must be included in the GroupRestrict groups being used.

If access control is not enabled, GroupRestrict functions as before.

Maintaining users and privileges

TimesTen allows the instance administrator to create, drop and alter users, when Access Control is enabled. It also allows the instance administrator to grant and revoke privileges for users. For details see Chapter 13, "SQL Statements in the *Oracle TimesTen In-Memory Database API and SQL Reference Guide*.

Administration of users is done at the instance level by establishing a connection to the system data store, and using the SQL commands to create and modify users. These commands are not transactional and cannot be rolled back.

Listing of defined users and privileges

The **ttUsers** built-in procedure lists current instance users and their privileges.

The **ttSchema** utility allows user definitions and privilege information to be output in the form of SQL statements that can be used to recreate the user environment within a different instance.

2

TimesTen Installation

This chapter contains configuration information that you will need to review before installing your TimesTen product on your system, in the sections:

- Platforms and configurations
- Installation instances
- Choosing the appropriate TimesTen components
- Installation prerequisites
- Operating system security considerations
- License file
- Prerequisites for non-root installations and Access Control on UNIX systems
- Changing the daemon port number on UNIX

You will find a description of the procedures to install TimesTen on your platform:

- Installing TimesTen on Windows systems
- Installing TimesTen on Solaris systems
- Installing TimesTen on HP-UX systems
- Installing TimesTen on HP-UX Memory Windows
- Installing TimesTen on AIX systems
- Installing TimesTen on Linux systems
- Installing TimesTen on Tru64 UNIX systems

This chapter also contains information to help you configure TimesTen after installation, work with the demo applications, migrate data stores to this release and view the TimesTen documentation:

- Using the Cache Administrator
- Informational messages on Windows systems
- Informational messages on UNIX systems
- ODBC installation

- Environment modifications
- Web server configuration
- Migrating data stores to TimesTen 6.0
- Building and running the demo applications
- Viewing the online documentation

Finally, this chapter contains information that helps you troubleshoot any problems that may arise during the installation process:

- Installation problems

Platforms and configurations

Platform support

Times Ten Data Manager and TimesTen Client/Server are supported in the following environments:.

Environment	32-bit	64-bit
Microsoft Windows 2000, Windows XP and Windows Server 2003 for Intel CPUs.	Yes	
Solaris 10 for AMD64 CPUs.	Yes	Yes
Solaris 8, 9 and 10 for UltraSparc architecture CPUs.	Yes	Yes
SUSE LINUX Enterprise Server 9 for Intel IA-32 and EM64T and AMD64 CPUs.	Yes	Yes
Red Hat Enterprise Linux ES, AS and WS 3 and 4 for Intel Itanium2 CPUs.	Yes	Yes
Red Hat Enterprise Linux AS, ES and WS 3 and 4 for Intel IA-32 and EM64T and AMD64 CPUs.	Yes	Yes
MontaVista Linux Carrier Grade Edition Release 3.1 for Intel IA-32 and EM64T CPUs.	Yes	Yes

Environment	32-bit	64-bit
HP-UX 11i and HP-UX 11i v2 for PA-RISC	Yes	Yes
HP-UX 11i v2 for Itanium2.	Yes	Yes
AIX 5L 5.2 and 5.3 for POWER systems.	Yes	Yes
Tru64 UNIX 5.1B-2 for AlphaChip EV68		Yes

JDK support

Note: TimesTen supports the Sun JVM and the BEA WebLogic JRockit JVM for Linux and Windows x86 systems. For details on JRockit, see `www.bea.com`.

TimesTen supports the following JDKs on the specified platforms:

Environment	JDK 1.4	JDK 5.0
Microsoft Windows 2000, Windows XP and Windows Server 2003 for Intel CPUs.	Yes	Yes
Solaris 10 for AMD64 CPUs.	Yes (32-bit and 64-bit)	Yes (32-bit and 64-bit)
Solaris 8, 9 and 10 for UltraSparc architecture CPUs.	Yes (32-bit and 64-bit)	Yes (32-bit and 64-bit)
SUSE LINUX EnterpriseServer 9 for Intel IA-32, EM64T and AMD64 CPUs.	Yes (32-bit and 64-bit)	Yes (32-bit and 64-bit)
Red Hat Enterprise Linux AS, ES and WS 3 and 4 for Intel Itanium2 processors.	Yes	

Environment	JDK 1.4	JDK 5.0
Red Hat Enterprise Linux AS, ES and WS 3 and 4 for Intel IA-32 and EM64T and AMD64 CPUs. Supported with BEA WebLogic JRockit 5.0 JVM.	Yes (32-bit and 64-bit)	AMD64 uses JDK 5.0 for 32-bit and 64-bit. No JDK support for EM64T
MontaVista Linux Carrier Grade Edition Release 3.1 for Intel IA-32 and EM64T CPUs. Supported with BEA WebLogic JRockit JVM.	Yes (32-bit and 64-bit)	Yes
HP-UX 11i and HP-UX 11i v2 for PA-RISC 32- and 64-bit.	Yes (32-bit and 64-bit)	Yes
HP-UX 11i v2 for Itanium2.	Yes (32-bit and 64-bit)	Yes
AIX 5L 5.2 and 5.3 for POWER systems.	Yes (32-bit and 64-bit) (Uses IBM JDK)	
Tru64 UNIX 5.1B-2 for AlphaChip EV68	Yes	Yes

Client/Server configurations

Any TimesTen client can connect to any TimesTen server on any platform where TimesTen is supported.

A TimesTen 5.0 or later client can connect to a Server DSN of a newer major release of TimesTen. For example, a 51 client can connect to a 6.0 Server DSN. Applications linked to newer TimesTen clients cannot connect to a Server DSN of any older releases. For example, a 6.0 client cannot connect to a 5.1 Server DSN.

Cache Connect to Oracle

TimesTen Cache Connect to Oracle allows you to cache Oracle Database data in TimesTen. The following Oracle releases are supported with this option:

- Oracle 10g Release 2 (Oracle 10.2.0.1)
- Oracle 10g Release 1 (Oracle 10.1.0.5 or above)
- Oracle 9i Release 2 (Oracle 9.2.0.4 or above)

Cache Connect is supported on the 32-bit and 64-bit platforms specified in this table:

Environment	32-bit	64-bit
Microsoft Windows 2000, Windows XP and Windows Server 2003 for Intel CPUs.	Yes	
Solaris 10 for x86 systems.		
Solaris 8, 9 and 10 for UltraSparc architecture CPUs.	Yes	Yes
SUSE LINUX EnterpriseServer 9 for Intel IA-32, EM64T and AMD64 CPUs.	Yes	Yes
Red Hat Linux ES, AS and WS 3 and 4 running on Intel Itanium2 processors.		Yes
Red Hat Enterprise Linux AS, ES and WS 3 and 4 for Intel IA-32 and EM64T and AMD64 CPUs.	Yes	Yes
HP-UX 11i and HP-UX 11i v2 for PA-RISC 32-bit and 64-bit.	Yes	Yes
HP-UX 11i v2 for Itanium2.	Yes	Yes
AIX 5L 5.2 and 5.3 for POWER systems.	Yes	Yes
Tru64 UNIX 5.1B-2 for AlphaChip EV68		Yes

Replication configurations

TimesTen-to-TimesTen Replication is supported only between identical platforms.

Installation instances

On UNIX, you can install more than one *instance* of any TimesTen release later than 5.0. By default, the instance name for this release is tt51.

If an instance of a particular release of TimesTen already exists on the machine, and you would like to install a second instance of the same TimesTen release, you must supply a unique instance name and port number. The TimesTen installation script can detect if an instance of the particular release of TimesTen already exists on the machine and will prompt you for a new instance name and port number for the main TimesTen daemon.

The instance name appears in the installation path and is the key used to access all necessary information about that particular installation of TimesTen. The instance name also appears in some TimesTen file names.

Note: On Windows, you can only install one instance of any major and minor release of TimesTen. The TimesTen installation script does not prompt you to supply an instance name.

Instance names

The instance name is case-insensitive and can have up to 255 characters. The name must be NON-NULL and can include underscores (_) or period (.), but no other special characters.

As each TimesTen installation is identified by the instance name, you can retrieve information about the TimesTen release number and port settings using the **ttVersion** utility.

Instance port numbers

Any time that you install more than one instance of TimesTen with the same major and minor release numbers on the same machine, the TimesTen installation script also requires that you specify a non-default port number for the main TimesTen daemon.

All TimesTen data stores that replicate to each other must use the same daemon port number. This port number is set at install time and can be verified using the **ttVersion** utility.

Choosing the appropriate TimesTen components

TimesTen allows you to select the components of TimesTen that you wish to install.

Components available on Windows

Type	Description
Compact	Installs the TimesTen client, ODBC drivers and examples.
Typical	Installs the TimesTen Data Manager, TimesTen Client, TimesTen Server, documentation and examples.
Custom	You may customize installation by selecting any of the following components: TimesTen Data Manager, the TimesTen Data Manager debug libraries, TimesTen Client and/or TimesTen Server.

Components available on UNIX:

Components	Description
TimesTen Client	Installs the TimesTen Client only. No other TimesTen components are installed on the machine. Use this installation to allow the TimesTen Client to access the Server on a remote machine.
TimesTen Data Manager	Installs the TimesTen Data Manager only. Use this installation to run the TimesTen Data Manager locally.
TimesTen Client, Server and Data Manager	Installs the TimesTen Data Manager, Client and Server on a single machine. Use this installation to: • Allow a Client on another machine to access the TimesTen Server on this machine. • Allow the TimesTen Client on this machine to access the TimesTen Server either locally or on a remote machine. • Allow applications to access the TimesTen Data Manager locally.

If you have already installed some components and you would like to add a component, you must install a new instance of TimesTen.

Installation prerequisites

Before installing TimesTen, make sure the appropriate requirements are met for your operating system.

On platforms where JDBC is supported you must have the appropriate version of the JDK installed on your machine to use JDBC. See "Platforms and configurations" on page 16 to learn which JDK is required for your platform.

Windows requirements

On Windows, TimesTen uses TCP/IP. If TCP/IP is not currently installed on your system, TimesTen will not install. To install TCP/IP, use the **Protocols** tab of the Network Control Panel.

UNIX requirements

In general, on UNIX systems, you must configure:

- The number of semaphores, and
- Allowable shared memory.

In addition, you may need to:

- Ensure you have the latest operating system patches
- Configure your file system to allow large files
- Configure your Java environment
- Configure your Client/Server environment
- Configure network settings for Replication

This section outlines some of the changes that need to be made on any UNIX system. It is followed by sections that describe changes required for each specific UNIX platform on which TimesTen is supported

Semaphores TimesTen consumes 1 SEMMNI per active data store, plus 1 additional SEMMNI per TimesTen instance where Client/Serve communication is done through shared memory. Set this semaphore to minimally allow for the number of data stores you intend to run.

Java On UNIX systems, if you are running JDBC, install the latest JDK patches. Refer to the website of the OS provider for the patches you may need.

It is possible for the JVM to run out of memory when selecting large VARCHAR/VARBINARY columns within a JDBC application. The maximum heap size used by the JVM can be increased by specifying the Java command line option `-Xmx`.

Other Client/ Server Settings

The maximum number of concurrent IPC connections to a TimesTen Server allowed by TimesTen is 9,999. However, system limits can take precedence on the number of connections to a single DSN. Client/ Server users can increase the file descriptor limit to support a large number of connections and processes.

For example, on a Solaris server, you may change the file descriptor limit to have a maximum of 1024 simultaneous client/server connections by adding the line:

```
set rlim_fd_max = 1080
```

in `/etc/system`.

In this case, 1080 is greater than the number of anticipated client/server connections and allows for a few extra connections.

AIX

Replication

For replication, TCP send and receive buffers should be increased to *a minimum* of 512KB. You may need to embed the following commands into a script that can be run at system boot time:

```
# /usr/sbin/no -p -o tcp_recvspace=524288
# /usr/sbin/no -p -o tcp_sendspace=524288
```

HP-UX

Semaphores

On HP-UX systems, to connect to more than 2 data stores simultaneously, you must increase the value of the kernel parameter `semmns`.

1. To view existing kernel parameter settings, log in as user `root`.

   For HP-UX 11i, use the command:

   ```
   # /usr/sbin/kmtune
   ```

   For HP-UX 11iv2, use the command:

   ```
   # /usr/sbin/kctune
   ```

Shared memory

On HP-UX systems, you also must increase the value of the parameter `shmmax`. To make these changes:

1. Use the `kmtune` or `kctune` commands above, or run the HP **System Administration Manager** to see existing kernel parameter settings:
   ```
   # /usr/sbin/sam
   ```
2. Double-click **Kernel Configuration**, then double-click **Configurable Parameters**.
3. Scroll down the list of parameters to `semmns` and change its value to a minimum of 4096 or greater.
4. For HP-UX 11i systems, also scroll down the list of parameters to `shmmax` and change its value to a maximum of 0x40000000.

   Note: The value `0x20000000` (a 2 followed by seven zeroes) indicates that the largest shared memory segment that can be created is 512 MB. The size of the shared memory segment required for a shared data store is larger than the requested data store size. Set this value high enough to support the largest shared memory segment needed.

   ---

5. Recompile the kernel. Choose **Create a New Kernel** from the Actions menu.
6. Reboot the system.

Large data stores

On 64-bit HP-UX systems, if you expect to have data stores that are larger than 2GB, you must enable large files. By default, HP-UX supports files that are no greater than 2GB in size.

To enable large files, create the filesystems using `newfs` with the `-o largefiles` option. Use the command:

```
% /usr/sbin/fsadm -F hfs -o largefiles device_name
```

For example:

```
% /usr/sbin/fsadm -F hfs -o largefiles \
    /dev/vg02/rlvol1
```

Replication

For replication, TCP send and receive buffers should be increased to *a minimum* of 512KB. You may need to embed the following commands into a script that can be run at system boot time:

For HP-UX 11i, 11.23 (11iv2)

```
# /usr/bin/ndd -set /dev/tcp tcp_xmit_hiwater_lfp 524288
# /usr/bin/ndd -set /dev/tcp tcp_recv_hiwater_lfp 524288
# /usr/bin/ndd -set /dev/tcp tcp_xmit_hiwater_lnp 524288
# /usr/bin/ndd -set /dev/tcp tcp_recv_hiwater_lnp 524288
# /usr/bin/ndd -set /dev/tcp tcp_xmit_hiwater_max 524288
# /usr/bin/ndd -set /dev/tcp tcp_recv_hiwater_max 524288
```

Linux

For Linux, TimesTen has been tested with Red Hat Enterprise Linux ES, AS and WS 3 and 4, the MontaVista Carrier Grade Edition (Linux) Release 3.1 and SUSE LINUX Enterprise Server 9 minimal configurations. This minimal configuration includes selecting "Network Workstation" from the custom installation up to the complete "Everything" configuration from custom installation. The C development tools are required if native development will be done on the machine.

Semaphores

To view existing kernel parameter settings, log in as `root` and use:

```
# /sbin/sysctl -a
```

Shared memory

To increase the shared memory size to 256 MB, for example, as `root`, edit the `/etc/sysctl.conf` file by adding the line:

```
kernel.shmmax=268435456
```

To increase the shared memory size without rebooting, use:

```
% /sbin/sysctl -w kernel.shmmax=268435456
```

If you have your kernel configured with the `/proc` file system and it is mounted, then the current maximum shared memory segment size (in bytes) can be viewed by the following command:

```
% cat /proc/sys/kernel/shmmax
```

You can also change this value by the following command

```
% echo 268435456 > /proc/sys/kernel/shmmax
```

This command has the same effect as the `sysctl` command.

IPC Client/ Server

On Red Hat Linux systems, to enable more than 6 ShmIpc Client/Server connections, add the line:

```
kernel.sem = "250 32000 100 100"
```

to the `/etc/sysctl.conf` file and reboot.

This sets the parameter values as follows:

```
SEMMSL=250
SEMMNS=32000
SEMOPM=100
SEMMNI=100
```

Client/Server and Cache Administrator

If you installing TimesTen/Cache and plan to use the web-based Cache Administrator, install the following RPM packages:

For Red Hat 3.0, install:

```
compat-libstdc++-7.3-2.96.123
```

For Red Hat 4.0, install:

```
compat-libstdc++-296-2.96.132.7.2
```

These packages can be install either using the rpm command or by using the Red Hat GUI installer found in "Legacy Software Development."

Replication

For replication, TCP send and receive buffers should be increased to *a minimum* of 512KB. You may need to embed the following commands into a script that can be run at system boot time:

```
# /sbin/sysctl -w net.ipv4.tcp_rmem="4096 4194304 4194304"
# /sbin/sysctl -w net.ipv4.tcp_wmem="98304 4194304 4194304"
# /sbin/sysctl -w net.ipv4.tcp_mem="98304 4194304 4194304"
# /sbin/sysctl -w net.core.rmem_default=65535
# /sbin/sysctl -w net.core.wmem_default=65535
# /sbin/sysctl -w net.core.rmem_max=4194304
# /sbin/sysctl -w net.core.wmem_max=4194304
# /sbin/sysctl -w net.ipv4.tcp_window_scaling=1
```

Cache Connect

For Cache Connect, TCP send and receive buffers should be increased to even greater values. You may need to embed the following commands into a script that can be run at system boot time:

```
# /sbin/sysctl -w net.ipv4.tcp_rmem="4096 4194304 4194304"
# /sbin/sysctl -w net.ipv4.tcp_wmem="98304 4194304 4194304"
# /sbin/sysctl -w net.ipv4.tcp_mem="98304 4194304 4194304"
# /sbin/sysctl -w net.core.rmem_default=262144
# /sbin/sysctl -w net.core.wmem_default=262144
# /sbin/sysctl -w net.core.rmem_max=4194304
# /sbin/sysctl -w net.core.wmem_max=4194304
# /sbin/sysctl -w net.ipv4.tcp_window_scaling=1
# /sbin/sysctl -w net.ipv4.ip_local_port_range="1024 65000"
```

Solaris

Operating system patches

Solaris 8 requires patch 108827-36 or later.

To view a list of installed patches, use:

```
% showrev -p
```

IPC semaphores

On Solaris, TimesTen checks the IPC configuration at install time. If either the IPC Semaphores module or the IPC Shared Memory module is not installed, you can install them by hand. Use the commands:

```
ryps3# modload /kernel/sys/semsys
ryps3# modload /kernel/sys/shmsys
```

Increase number of semaphores

For Solaris 10 systems, the default semaphore settings should be sufficient without entries in `/etc/system`. To set shared memory on Solaris 10 systems, specify `project.max-shm-memory`.

On other Solaris systems, you may need to increase the number of semaphores. TimesTen consumes 1 SEMMNI per active data store, plus one additional SEMMNI per TimesTen instance where Client/Server communication is done through shared memory.

For each data store, TimesTen consumes 100 SEMMSL if the Connections attribute is set to the default value (64), and one additional SEMMSL for each estimated connection above the default.We recommend that you increase the number of semaphores:

1. Log in as user `root`.
2. Set or add the following lines to `/etc/system`:

```
set semsys:seminfo_semmni = 20
set semsys:seminfo_semmsl = 512
set semsys:seminfo_semmns = 10240
set semsys:seminfo_semmnu = 10240
```

Note: The values in this step are the minimum number of required semaphores. You can increase these numbers as needed.
We recommend that you use the formula: SEMMNS=SEMMNU = (SEMMNI * SEMMSL).

---

3. Reboot your system.
4. To view the current limits, use:

```
% /usr/sbin/sysdef
```

This command displays the limits for SEMMSL, SEMMNS, SEMOPM, and SEMMNI, respectively.

SEMOPM is the maximum number of operations per semop call. It does not need to be reset.

Shared memory IPC client connections

On Solaris, to have more than 6 ShmIpc-enabled Client DSN connections per process, you must make changes to the SHMSEG kernel parameter.

- To access more than 6 data stores, you must make changes to the SHMSEG kernel parameter. For example to allow a single process to

access 12 data stores, add the following line to `/etc/system` and reboot before using TimesTen:

```
set shmsys:shminfo_shmseg=12
```

Java

To run 64-bit Java applications on Solaris, using the Sun 64-bit JVM, you may need to pass the `-d64` options to the Java command line.

Other changes

Other changes that you may need to make to your Solaris system include the following:

- To allow a large number of connections to a data store, add the following lines to `/etc/system` and reboot before using TimesTen:

  ```
  set rlim_fd_cur=4096
  set rlim_fd_max=4096
  ```

- To enable large shared memory objects in Solaris, add the following line to `/etc/system` and reboot before using TimesTen:

  ```
  set shmsys:shminfo_shmmax = 0x20000000
  ```

  Note: The value `0x20000000` (a 2 followed by seven zeroes) indicates that the largest shared memory segment that can be created is 512 MB. The size of the shared memory segment required for a shared data store is larger than the requested data store size. Set this value high enough to support the largest shared memory segment needed.

Large data stores

If you keep data stores on a Solaris UFS file system, and are using transaction-consistent checkpoints, you may need to change the settings of some kernel parameters to get the best performance for your checkpoints. The Solaris UFS Throttle algorithm causes processes that write a single large file to be put to sleep when a byte count threshold exceeds the *high-water mark*. To disable the algorithm, add the line:

```
set ufs:ufs_WRITES = 0
```

to the `/etc/system` file.

Alternatively, you can increase the high-water mark by adding the line:

```
set ufs:ufs_HW = <desired value>
```

to the `/etc/system`.file

You must reboot the system for the new value to take effect.

Setting the high-water mark to the size of the checkpoint file should provide satisfactory performance, although a lower value may as well. More information on the UFS Throttle algorithm may be obtained in the

white paper, “Understanding Solaris Filesystems and Paging” (SMLI TR-98-55) available from `http://www.sun.com`.

Replication

For replication, TCP send and receive buffers should be increased to *a minimum* of 512KB. You may need to embed the following commands into a script that can be run at system boot time:

```
# /usr/sbin/ndd -set /dev/tcp tcp_xmit_hiwat=524288
# /usr/sbin/ndd -set /dev/tcp tcp_recv_hiwat=524288
```

Tru64 UNIX

Semaphores

On Tru64 UNIX systems, to view existing kernel parameter settings, log in as user `root`, and use the command:

```
# /sbin/sysconfig -q ipc
```

For each data store, TimesTen consumes 100 SEMMSL if the Connections attribute is set to the default value (64), and one additional SEMMSL for each estimated connection above the default.We recommend that you increase the number of semaphores:

1. Log in as user `root`.
2. To view the current limits, use:

   ```
   % sysconfig -q ipc
   ```

   This command displays all the parameters of the IPC subsystem.
3. If the broadcast_wakeup semaphore parameter exists on your system, it must be set to one:

   ```
   sem_broadcast_wakeup = 1
   ```
4. Run the Tru64 UNIX **dxkerneltuner**:

   ```
   % dxkerneltuner
   ```
5. Double-click **IPC**.
6. Scroll down the list of parameters and change the values of the following parameters to at least the values indicated here:

   ```
   sem_mni = 20
   sem_msl = 512
   sem_opm = 100
   sem_ume = 100
   ```

Note: The values in this step are the minimum number of required semaphores. You can increase these numbers as needed.

Shared memory

To increase the maximum address space a process may use, change the kernel values `per_proc_address_space` and `max_per_proc_address_space`

To increase the maximum data segment (malloc space) a process may use, change the kernel values `per_proc_data_size` and `max_per_proc_data_size`

For example, to change all of these values to 10GB:

1. Create a text file, `kernelparams`, that contains the following:

```
per_proc_address_space = 10737418240
max_per_proc_address_space = 10737418240
per_proc_data_size = 10737418240
max_per_proc_data_size = 10737418240
```

2. Log in as root and run the command:

```
# sysconfigdb -m -f kernelparams proc
```

3. You may need to reboot the system after you have made these changes. Alternatively, you can run the command:

```
# sysconfig -q proc
```

4. To view the value of the proc kernel subsystem, run the command:

```
% sysconfig -r subsys ttr=value
```

Shared memory IPC client connections

On Tru64 UNIX, to have more than 6 ShmIpc-enabled Client DSN connections per process, you must make changes to the SHMSEG kernel parameter. Kernel parameters can be changed with either the **dxkerneltuner** interface or the `sysconfigdb` command.

Replication

For replication, TCP send and receive buffers should be increased to *a minimum* of 512KB. You may need to embed the following commands into a script that can be run at system boot time:

```
# sysconfig -r inet tcp_recvspace=524288
# sysconfig -r inet tcp_sendspace=524288
```

Disk space requirements

The following table lists approximate disk space requirements for a typical installation. Requirements listed as “Movable” indicate that the component is installed in a user-specified file system at installation time. The variable *`install_dir`* represents the directory where the current release of TimesTen is installed. The TimesTen default installation directories for release 6.0 are:

- On Windows, `C:\TimesTen\tt51`

- On HP-UX, Solaris and Linux, `/opt/TimesTen/tt51`
- On AIX, `/usr/lpp/TimesTen/tt51`

For each operating system, TimesTen installs certain files in a system directory that cannot be moved.

Operating system	Installation type	File system	Movable?	Required space
Windows 2000, XP or 2003	TimesTen Data Manager	`%WINDIR%\SYSTEM32`	No	10 MB
		`install_dir`	Yes	20 MB
Windows 2000, XP or 2003	TimesTen Client	`%WINDIR%\SYSTEM32`	No	2 MB
		`install_dir`	Yes	15 MB
Windows 2000, XP or 2003	TimesTen Client, Server and Data Manager	`%WINDIR%\SYSTEM32`	No	10 MB
		`install_dir`	Yes	22 MB
Windows 2000, XP or 2003	TimesTen JDBC class files and DLL	*`install_dir`*	Yes	103K
Solaris 8, 9 and 10 32- or 64-bit	TimesTen Data Manager	*`install_dir`*	Yes	117 MB
		`/etc`	No	12 KB
Solaris 8, 9 and 10 32-bit	TimesTen Client	*`install_dir`*	Yes	20 MB
Solaris 8, 9 and 10 32- or 64-bit	TimesTen Client, Server and Data Manager	*`install_dir`*	Yes	265 MB
		`/etc`	No	12 KB
Red Hat Enterprise Linux ES, AS and WS 3 and 4 or SUSE LINUX Enterprise System 9	TimesTen Data Manager	*`install_dir`*	Yes	71 MB
		`/etc`	No	12 KB

Operating system	Installation type	File system	Movable?	Required space
Red Hat Enterprise Linux ES, AS and WS 3 and 4 or SUSE LINUX Enterprise System 9	TimesTen Client	*install_dir*	Yes	18 MB
Red Hat Enterprise Linux ES, AS and WS 3 and 4 or SUSE LINUX Enterprise System 9	TimesTen Client, Server and Data Manager	*install_dir*	Yes	164 MB
		/etc	No	12 KB
MontaVista Carrier Grade Edition (Linux) Release 3.1	TimesTen Data Manager	*install_dir*	Yes	84 MB
		/etc	No	12 KB
MontaVista Carrier Grade Edition (Linux) Release 3.1	TimesTen Client	*install_dir*	Yes	6 MB
MontaVista Carrier Grade Edition (Linux) Release 3.1	TimesTen Client, Server and Data Manager	*install_dir*	Yes	177 MB
		/etc	No	12 KB
HP-UX 11i and HP-UX 11i v2 32- or 64-bit for PA-RISC	TimesTen Data Manager	*install_dir*	Yes	63 MB
		/etc	No	8 KB
		/sbin	No	8 KB

Operating system	Installation type	File system	Movable?	Required space
HP-UX 11i and HP-UX 11i v2 32- or 64-bit for PA-RISC	TimesTen Client	*install_dir*	Yes	20 MB
		/etc	No	8 KB
		/sbin	No	8 KB
HP-UX 11i and HP-UX 11i v2 32- or 64-bit for PA-RISC	TimesTen Client, Server and Data Manager	*install_dir*	Yes	150 MB
		/etc	No	8 KB
		/sbin	No	8 KB
HP-UX 11i IA32	TimesTen Data Manager	*install_dir*	Yes	76 MB
		/etc	No	8 KB
		/sbin	No	8 KB
HP-UX 11i IA32	TimesTen Client	*install_dir*	Yes	23 MB
		/etc	No	8 KB
		/sbin	No	8 KB
HP-UX 11i IA32	TimesTen Client, Server and Data Manager	*install_dir*	Yes	181 MB
		/etc	No	8 KB
		/sbin	No	8 KB
HP-UX 11i IA64	TimesTen Data Manager	*install_dir*	Yes	57 MB
		/etc	No	8 KB
		/sbin	No	8 KB
HP-UX 11i IA64	TimesTen Client	*install_dir*	Yes	8 MB
		/etc	No	8 KB
		/sbin	No	8 KB
HP-UX 11i IA64	TimesTen Client, Server and Data Manager	*install_dir*	Yes	128 MB
		/etc	No	8 KB
		/sbin	No	8 KB

Operating system	Installation type	File system	Movable?	Required space
AIX 5.2 and 5.3 32- or 64-bit	TimesTen Data Manager	*install_dir*	Yes	42 MB
		/etc	No	10 KB
AIX 5.2 and 5.3 32-bit	TimesTen Client	*install_dir*	Yes	5 MB
		/etc	No	10 KB
AIX 5.2 and 5.3 32- or 64-bit	TimesTen Client, Server and Data Manager	*install_dir*	Yes	86 MB
		/etc	No	10 KB
Tru64 5.1B-2 UNIX	TimesTen Data Manager	*install_dir*	Yes	32 MB
Tru64 5.1B-2 UNIX	TimesTen Client	*install_dir*	Yes	5 MB
Tru64 5.1B-2 UNIX	TimesTen Client, Server and Data Manager	*install_dir*	Yes	71 MB

In addition to the requirements listed in this table, if you have applications with transactions that do large deletes, you may need to allocate extra space on the disk where your temporary directory resides. TimesTen creates temporary files when large amounts of space in a data store are freed by a transaction. In addition, other TimesTen operations, such as `ttRepAdmin -duplicate` operations and large deletes, use the temporary directory when copying files.

TimesTen documentation requires additional space.

By default, the temporary directory is:

- On Windows, `C:\Temp`
- On Solaris, Linux and Tru64 UNIX, `/tmp`
- On HP-UX and AIX, `/var/tmp`

You must set the location of your temporary directory by setting the `TMP` environment variable on Windows. On UNIX, you can change the location of your temporary directory by setting the `TMPDIR` environment variable. If the location of the temporary directory is changed, you must update the daemon startup script to specify the location of the temporary

directory. For the location of the script, see the platform-specific installation instructions in this chapter.

Note: On Windows, the complete temporary directory path must be less than 190 characters for the installation to complete successfully.

Stack space requirements

Most applications require a stack space of 16K on 32-bit systems and between 17K to 55K on 64-bit systems.

Cache Connect

If you are using features of TimesTen/Cache, such as CREATE CACHE GROUP or LOAD CACHE GROUP statements, you must install at least a client installation of Oracle 9i on the machine where you are installing TimesTen.

Oracle shared libraries are required in order to cache Oracle data in TimesTen. You must have Oracle 9i client, Oracle Enterprise or Oracle Database 10g installed. You also must have the `ORACLE_HOME` environment variable defined. See "ORACLE_HOME environment variable" on page 77.

Operating system security considerations

There are two mutually exclusive modes of operation for TimesTen that have OS security implications.

1. ***Non-root installation*** (available on all non-Windows platforms). In general, it is safer not to run any processes as a privileged user, such as root, unless absolutely necessary. When performing non-root installations, certain procedures must be performed as user `root`. See the "Prerequisites for non-root installations and Access Control on UNIX systems" on page 38.
2. ***GroupRestrict mode.*** When a data store is first created, it can be created in **GroupRestrict** mode so that all of its files and shared memory segments are ownership restricted to that of a particular operating system group. This mode only works if TimesTen is installed and running as `root`. See the *Oracle TimesTen In-Memory Database API and SQL Reference Guide*.

License file

This version of TimesTen contains a license file, `ttlicense.dat`. If you install TimesTen using the installation CD or ftp file no action is required on your part to use TimesTen software. TimesTen customers that create their own custom installation to redistribute TimesTen should read the following information.

The `ttlicense.dat` allows TimesTen to optionally create TimesTen products for evaluation purposes only that have expiration dates. All TimesTen customers receive the version of TimesTen that has no expiration date. Both the expiration and non-expiration versions require the `ttlicense.dat` file to exist in the proper location.

File location

On UNIX systems, the license file on the TimesTen CD or ftp package is in the `key` subdirectory. Upon install, `ttlicense.dat` is placed in the *`install_dir`*`/bin` directory

On Windows systems, the license file on the TimesTen CD or ftp package is in the `root` directory. Upon install, `ttlicense.dat` is placed in the *`install_dir`*`/srv` directory.

License-related errors

If the license file is missing or altered, upon a connection attempt you will get an error message. A description of the messages can be found in the *Oracle TimesTen In-Memory Database API and SQL Reference Guide*.

If you get one of the license error messages, you can restore the original license file from a known good copy of the same patch version of TimesTen.

In some cases, you may need to restart the TimesTen Data Manager Server (Service on Windows) in order to detect a good `ttlicense.dat` file.

Redistribution

For redistribution to your users, make sure the `ttlicense.dat` is installed in the proper location as indicated above. You may want to create a backup copy of `ttlicense.dat` in the same directory so that you can advise your users to rename the file should they encounter one of the above errors. In that case, they can then rename the backup license file instead of locating a copy of it from an installation CD or ftp

site. If you have any further questions about `ttLicense.dat`, please contact support@timesten.com.

Prerequisites for non-root installations and Access Control on UNIX systems

As discussed in Chapter 1, "Access Control and non-root installations," on UNIX systems, you can install TimesTen as a non-root user. This entire section applies to all UNIX platforms on which TimesTen is supported, unless otherwise indicated.

However, you must perform certain tasks as the user `root`, both prior to installing TimesTen and after installation. This section outlines those tasks that must be performed as the user, `root`.

Installation prerequisites for non-root installs

You must be sure that the prerequisites defined in "Installation prerequisites" on page 22 have been met, before continuing with your installation. Perform the pre-requisite steps for your particular platform.

The following steps are required for installations that are installed by a non-root user, whether they use Access Control or not. They are also required for all installations that will enable Access Control at install time.

Create the TimesTen instance administrators group

Before installing TimesTen, you must create the instance administrators group:

1. Log in as root at the shell prompt:

   ```
   % su -
   ```

2. Create an operating system group for the TimesTen administrators group. Only members of this group can install TimesTen.

   We suggest using the name `timesten` for the group, but you can choose any other name that you prefer.

3. If you prefer to install TimesTen as a non-root user, add the user(s) who are installing and administering TimesTen to the TimesTen administrators group.

When installing as a non-root user on HP-UX systems, the operating system user running the TimesTen daemon must belong to an operating system group that has been given the `MLOCK` privilege.

For example, if the user is a member of a group called `timesten`, then the following command (run as `root`) gives the `timesten` group the `MLOCK` privilege:

```
# setprivgrp timesten MLOCK
```

The `getprivgrp` command can be used to check the privileges of a group:

```
$ getprivgrp timesten
timesten: MLOCK
```

Note: On Linux and Tru64 systems, root privileges are required to use MemoryLock attribute. On Solaris systems, you must be installed as root to use **MemoryLock**=**1** or **2**.Data stores in a non-root instance of TimesTen can use settings **3** and **4** for this attribute, on Solaris systems.

Create the TimesTen registry

1. If the directory `/etc/TimesTen` does not already exist, create it.

   ```
   # mkdir /etc/TimesTen
   The disk space required for the files in this directory is less
   than 2k bytes.
   ```

2. Assign ownership and assign the permissions on this directory.

   For example, with a TimesTen administrators group named `timesten`, use:

   ```
   # chmod 775 /etc/TimesTen
   # chgrp timesten /etc/TimesTen
   ```

   If you are upgrading from a previous releases of TimesTen, you must change the existing group ownership and permissions of `/etc/TimesTen` one time per machine. In that case the `instance_info` file must be assigned write permissions for members of the TimesTen administrators group. For example:

   ```
   # chgrp -R timesten /etc/TimesTen
   # chmod 664 /etc/TimesTen/*
   ```

3. You can now install TimesTen. See the section in this chapter on installing TimesTen for your specific platform. The installer will verify the existence and permissions of `/etc/TimesTen` and will fail if not present and correct.

Post-installation requirements

For non-root installs, in order to install the TimesTen daemon start scripts in the proper locations, the user root must run the `setuproot` script located in the *`install_dir`*`/bin` directory:

```
# setuproot -install
```

Configure the syslog messages

Finally, you must configure your syslog messages. See "Informational messages on UNIX systems" on page 72 for details.

Changing the daemon port number on UNIX

The script `ttmodinstall` ships with TimesTen to allow the instance administrator to change the port number on which the main TimesTen daemon listens. If you have not stopped the TimesTen daemon before using ttmodinstall, TimesTen stops it before making the port change. After the port change, you must restart the TimesTen daemon, with the command `ttDaemonAdmin -start`.

This is useful, if you install TimesTen and later find that the port is already in use.

The script is run from the command line and takes the `-port` option with the new port number as an argument. For example:

```
% ttmodinstall -port 12345
```

Any other changes to the TimesTen instance can only be made by uninstalling TimesTen and re-installing the same or a new product.

UNIX libraries

On UNIX, TimesTen installs the production and debug versions of the Data Manager library and ODBC driver. In the `sys.odbc.ini` file, set the driver version that you want to use for each available data store. See "Defining data sources for the demo applications" on page 83 for more information on the `sys.odbc.ini` file. Also see "User and system DSNs" in the *TimesTen Developers Guide*.

The TimesTen CD-ROM contains two versions of each of the TimesTen drivers. They differ by the library and ODBC driver they use.

- Use the *production* version of TimesTen for most application development and all deployment.
- Use the *debug* version of TimesTen for debugging during application development. This version performs additional internal error-checking and is considerably slower than the production version.

On UNIX, TimesTen installs the production and debug versions of the Data Manager library and ODBC driver. In the `sys.odbc.ini` file, set the driver version that you want to use for each available data store. See

"Defining data sources for the demo applications" on page 83 for more information on the `sys.odbc.ini` file. Also see "User and system DSNs" in the *TimesTen Developers Guide*.

On UNIX, the TimesTen debug libraries are compiled with the `-g` option to display additional debug information.

See the *TimesTen Developer's Guide* for more information about using TimesTen.

Installing TimesTen on Windows systems

WINDOWS

This section discusses installation and related issues for Windows systems. For a list of Windows platforms supported by TimesTen, see "Platforms and configurations" on page 16.

Note: Before beginning installation, be sure that the prerequisites defined in "Installation prerequisites" on page 22 have been met.

Installing TimesTen

An installation program Installs your TimesTen product on Windows systems. The TimesTen CD-ROM is configured to autoplay; the installation program is automatically invoked when the CD-ROM is inserted into the CD-ROM drive.

To install TimesTen manually, insert the CD, then run the command:

```
D:\WINDOWS\SETUP.EXE
```

where `D:` is the CD-ROM drive.

Note: Each time `SETUP.EXE` is executed, the install program checks for previous installations. If a previous version of any TimesTen product exists, the setup program starts in Maintenance Mode. In order to do a complete re-install, you must first uninstall the TimesTen product in Maintenance Mode and then run `SETUP.EXE` again.

On Windows, when installing the TimesTen Data Manager, the debug library and debug ODBC driver are not installed if you choose the Typical setup. To install the debug library and debug ODBC driver, choose the Custom setup.

TimesTen does not install the debug library if you do not have Microsoft Visual C++ 6.0 installed.

Custom setup also lets you choose other custom options.

The installation program adds TimesTen directories to the system environment variables LIB and INCLUDE.

In addition, installation prompts you to add a directory to the system environment variable PATH. If you decide not to set the PATH environment variable at installation time, you can set the PATH environment variable at any time after installation on a per session basis by running the script *install_dir*\bin\ttenv.bat.

Note: On Windows, TimesTen cannot be installed in a substituted directory (a directory that is mapped to a drive letter). Attempting to install TimesTen in a substituted directory results in an error.

Installing TimesTen in silent mode

TimesTen allows you to save installation options to a batch file that you can later use to install TimesTen without having to answer each option in a dialog box. To set up silent mode:

- From a command-line, run:

  ```
  C:> setup.exe -r
  ```

  With this option, TimesTen walks you through a normal setup operation with all the dialog boxes. TimesTen saves your responses to the file `C:\WINDOWS\setup.iss`.

To run an installation in silent mode:

- From a command-line, run: `setup.exe -f1`*response_file*. For example:

  ```
  C:> setup.exe -s -f1C:\WINDOWS\setup.iss
  ```

  acquires the installation options from the response file. No dialog boxes appear. Some information pop-up dialogs may still appear, such as the one that informs you that the services are being started.

Note: Batch files from releases older than TimesTen Release 6.0 should not be used to install this release. All new prompts in the installation script for this release are assigned default answers and may produce unexpected results.

Verifying installation

To verify that TimesTen has been properly installed, check that the driver files are available and that the services are running:

1. Check that the TimesTen 6.0 Start menu shortcut has been added to the Windows Desktop **Start > Programs** menu.
2. On the Windows Desktop, choose **Start > Settings > Control Panel > Administrative Tools > Data Sources (ODBC)**. This opens the ODBC Data Source Administrator.
3. Click **Drivers**. Check to see that the correct drivers are installed. You should see the TimesTen Data Manager driver. If you installed the debug library, you should also see the TimesTen Data Manager debug driver. If you installed TimesTen Client, you should see the **TimesTen Client 6.0** driver. Click **OK**.
4. On the Windows Desktop, choose **Start > Settings > Control Panel > Administrative Tools > Services** and check that the TimesTen Data Manager 6.0 service has the word "Started" in the Status field. At this time, you can also set **Recovery** options to attempt to restart the service after a failure.

   These steps verify that the system has been installed properly.

Verifying TimesTen Client and Server installation

To verify that the Client and Server have been properly installed:

1. On the Windows Desktop, choose **Start > Settings > Control Panel > Administrative Tools > Data Sources (ODBC)**.
   This opens the ODBC Data Source Administrator.
2. Click **System DSN**.
3. Select the **RunDataCStt51** or **ShmRunDataCStt51** sample data source and click **Configure**.

Note: The **RunDataCStt51** DSN is used for client applications that use TCP/IP communications with the TimesTen Server. The **ShmRunDataCStt51** DSN is used for client applications that use a shared memory segment to communicate with a TimesTen Server on the same machine.

   This opens the TimesTen Client Data Source Setup dialog.

4. Click **Test TimesTen Server Connection** to attempt a connection to the server.

   The ODBC Administrator attempts to connect to the TimesTen Server and display a message to let you know if it was successful. When you click this button, the TimesTen Client verifies that:

- ODBC, Windows sockets, and the TimesTen Client are installed on the machine.
- The TimesTen Server you have selected is defined.
- The host machine for the TimesTen Server is running.
- The TimesTen Server is running.

5. Click **Test Data Source Connection** to attempt a connection to the data source on the TimesTen Server.

   The ODBC Data Source Administrator attempts to connect to the TimesTen data source and displays a dialog to let you know if it was successful. When you click **Test Data Source Connection**, the TimesTen Client verifies that:

   - The data source you have chosen is defined on the server.
   - The TimesTen Client can connect to the data source.

Working with the Data Manager Service and the Server

The TimesTen Data Manager Service starts automatically when you install the TimesTen Data Manager. In addition, if you installed the TimesTen Server, it is automatically started whenever the TimesTen Data Manager service is started. You can change the startup mode for the TimesTen Data Manager to require manual startup.

Note: You must have administrative privileges to set the startup mode or to start and stop the TimesTen Data Manager service.

To change the startup mode:

1. On the Windows desktop, choose **Start > Settings > Control Panel > Administrative Tools > Services.** This displays all currently available services.
2. Select **TimesTen Data Manager 6.0**.
3. Choose either **Manual** or **Automatic** from the Startup type list. Click **OK**.

If the TimesTen Data Manager startup mode is Manual, follow these instructions to start and stop the service:

1. On the Windows desktop, choose **Start > Settings > Control Panel > Administrative Tools > Services.** This displays all currently available services.
2. Select **TimesTen Data Manager 6.0**.

3. Click **Start** to start the service. If the service is already running, click **Stop** to stop the service.

Note: TimesTen writes events into the Event Log file. The Windows Application Event Log can get full. To avoid filling the Application Event Log, check the log settings in the Event Viewer. You can change the size of the Event Log or control whether it overwrites old events.

Uninstalling TimesTen

To uninstall TimesTen for Windows:

- On the Windows Desktop, choose **Start > Settings > Control Panel > Add/Remove Programs**.

To verify that removal was successful, check that:

- The TimesTen 6.0 Start menu shortcut has been removed from the **Start > Programs** menu.
- The TimesTen Data Manager 6.0 has been removed from the **Services** list.
- The TimesTen 6.0 drivers have been removed from the **ODBC Drivers** tab in the ODBC Control Panel.

Note: DSNs created by TimesTen installation are removed upon TimesTen uninstall. DSNs created by users are not removed during TimesTen uninstall.

Installing TimesTen on Solaris systems

This section discusses installation and some related topics for Solaris systems.

Note: Before beginning installation, be sure that the prerequisites defined in "Installation prerequisites" on page 22 have been met.

Installing TimesTen

To install TimesTen on your Solaris system, follow these steps:

1. Log in as the TimesTen instance administrator if installing as non-root, or
   Log in as root at the shell prompt:

```
% su -
```

2. Load the CD-ROM into the CD drive as follows:
   - If the Volume Manager is installed on the system, you don't have to mount the CD; it is automatically mounted as:

     ```
     /cdrom/tt6.0
     ```
   - Otherwise, you have to create, then mount, the `cdrom` directory as follows:

     ```
     # mkdir /cdrom
     # /etc/mount -r -F hsfs /dev/sr0 /cdrom
     ```
3. Run the setup script by typing the following:

   ```
   # cd mount_dir
   # ./setup.sh
   ```

   where *`mount_dir`* is the directory where the CD is mounted (e.g.: `/cdrom`).

   You can run the setup script with the option `-install` or `-uninstall` (default is `-install`). When you use the `-uninstall` option, the script stops the daemon if it is running and removes all files it had installed.

Note: To uninstall a TimesTen product, you must run `setup.sh -uninstall` in a directory outside of the installation instance that you wish to uninstall. For example to uninstall the default instance run `/opt/TimesTen/tt51/bin/setup.sh -uninstall`.

In addition, `setup.sh` also accepts the following options:

`-batch` *`filename`*	Installs your TimesTen product without having to respond to prompts. If *`filename`* is specified, the installation reads all installation prompts from the file. The batch file *`filename`* is optional. However, TimesTen recommends that you create the batch file and specifically indicate the instance name of the installation. If no batch file is provided or if the batch file does not contain an instance name, TimesTen installs a default instance, using "tt51" for the instance name. If an instance with the same name already exists on the installation machine, the install procedure fails. On 64-bit platforms, the batch file must also specify either the 32-bit and 64-bit version of TimesTen be installed. If no batch file is provided or no platform is specified in the batch file, the 32-bit version is installed in the default instance.
`-record` *`filename`*	Installs your TimesTen product and records responses to prompts described in *`filename`*. This option cannot be used with the TimesTen uninstaller.
`-doc`	Installs documentation.
`-help`	Displays the help message.
`-verbose`	Displays extra installation information.

The CD contains tar files of TimesTen. If the setup script cannot find the tar files to extract from, it prompts you for their location.

4. Enter your response to the setup script prompts.

Note: To install TimesTen without having to respond to prompts, use the `-batch` option with the `setup.sh` script. Batch files from releases older than TimesTen Release 6.0 should not be used to install this release. All new prompts in the installation script for this release are assigned default answers and may produce unexpected results.

The setup script performs these actions (unless your answers resulted in termination of the installation process):

- On 64-bit systems, prompts you to install one of the following releases:
  - 32-bit (default)
  - 64-bit
- Prompts you for the location of your TimesTen installation, if installing as a non-root user.
- Prompts you to determine if Access Control should be enabled, except for Client-only installs. Default answer is "No." In that case, no other changes are needed to your installation or your use of TimesTen. For more details on Access Control, see Chapter 1, "Access Control and non-root installations in this guide.
- Prompts you to choose whether to upgrade an existing instance of TimesTen, or to install a new instance of TimesTen.
- Prompts you to specify the daemon port number. If no instances of TimesTen are installed on the machine, or if no instances use the default port number 16000 for 32-bit installations and 16001 for 64-bit applications, prompts you to use the default port number.
- May prompt you to enter an instance name. See "Installation instances" on page 20.
- Prompts you to install your TimesTen product:
  - TimesTen/DataServer or
  - TimesTen/Cache
- Prompts you to install one of the following components.
  - TimesTen Client, Server and Data Manager
  - TimesTen Data Manager only
  - TimesTen Client only
- Removes any previous installation of this release of TimesTen if you are installing an upgrade.
- Untars the appropriate tar file for the component(s) being installed into the install directory, by default `/opt/TimesTen/tt51`.
- Copies the daemon scripts into the appropriate directories.
- If installed by user `root`, configures the system to start the daemon when the system boots.
- Creates the directory where data stores created by the TimesTen demo applications will reside. By default, they are stored in `/var/TimesTen/tt51/DemoDataStore`.
- If you are installing TimesTen/Cache, prompts you for the location of your Oracle version 9i or Oracle Database 10g installation.

- Starts the daemon.
- If there are other instances of the same patch release of TimesTen installed on the same machine, prompts you to provide a unique port to be used by the TimesTen daemon.
- If the TimesTen Server is being installed, prompts you to configure the Server: server name and logging options.
- Prompts you to install the TimesTen documentation.

The daemon writes a `timestend.pid` file into the directory the daemon was started from: `/var/TimesTen/`*`TTinstance`*`/` if installed by the user `root` or *`install_dir`*`/info` if installed by a non-root user.

This file contains the daemon's process ID. When the script to stop the daemon is run, this ID is used to determine the process to terminate. When the process terminates, the `timestend.pid` file is removed.

Note: When compiling, use an ANSI C compiler. The recommended compiler is the Sun Workshop Compiler 5.0 or 6.0

Working with the daemon and Server

The TimesTen daemon (`timestend`) starts automatically when the operating system is booted and operates continually in the background. Application developers do not interact with the daemon directly; no application code runs in the daemon and application developers do not, in general, have to be concerned with it. Application programs that use TimesTen data stores communicate with the daemon transparently by using TimesTen internal routines.

There are situations, however, when you may have to start and stop the daemon manually, using the TimesTen main daemon startup script. This section explains how to start and stop the daemon. If you have installed the TimesTen Server, it starts automatically when the TimesTen daemon is started and stops automatically when the TimesTen daemon is stopped.

Note: You must have root privileges or be the TimesTen instance administrator to interact with the TimesTen daemon.

To stop the daemon manually, enter the command:

```
# /etc/init.d/tt_TTinstance stop
```

or use the utility command `ttDaemonAdmin -stop`

To start the daemon manually, enter the command:

```
# /etc/init.d/tt_TTinstance start
```

or use the utility command `ttDaemonAdmin -start`

Uninstalling TimesTen

To uninstall all TimesTen components:

1. Log in as the TimesTen instance administrator if you installed as non-root, or
   Log in as root at the shell prompt:

   ```
   % su -
   ```

2. The TimesTen setup script is in the *install_dir*`/bin` directory. Run the script with the `-uninstall` option from a directory outside of the installation instance:

   ```
   # install_dir/bin/setup.sh -uninstall
   ```

   Uninstalling the system removes all TimesTen libraries and executables and also stops and uninstalls the daemon. You can execute `ps` to verify that all TimesTen processes have terminated. To verify that TimesTen has been successfully uninstalled, verified that the *install_dir* no longer exists.

Installing TimesTen on HP-UX systems

HP-UX

This section discusses installation and some related topics for HP-UX systems.

Note: Before beginning installation, be sure that the prerequisites defined in "Installation prerequisites" on page 22 have been met.

Installing TimesTen

To install the TimesTen Data Manager on your system, follow these steps:

1. Log in as the TimesTen instance administrator if installing as non-root, or
   Log in as root at the shell prompt:

   ```
   % su -
   ```

2. Load the CD-ROM into the CD drive as follows: If the `cdrom` directory doesn't exist, create it:

   ```
   # mkdir /cdrom
   ```

   - Mount the CD-ROM, as follows:

– If your system is configured to mount the CD-ROM at /cdrom, type:

```
# /etc/mount /cdrom
```

– Otherwise, mount the CD-ROM device name to the /cdrom directory, as follows:

```
# /etc/mount -r cdfs CD-ROM_device_name /cdrom
```

where *CD-ROM_device_name* is the name of the CD-ROM device.

3. Run the setup script by typing the following:

```
# cd mount_dir
# ./SETUP.SH;
```

where *mount_dir* is the directory where the CD is mounted (e.g.: /cdrom).

You can run the setup script with the option -install or -uninstall (default is -install). When you use the -uninstall option, the script stops the daemon and Server, if they are running, and removes all files it had installed.

Note: To uninstall TimesTen, you must run setup.sh -uninstall in a directory outside of the installation instance that you wish to uninstall. For example to uninstall the default instance run /opt/TimesTen/tt51/bin/setup.sh -uninstall.

Specify the option at the end, outside the quotation marks. In addition, `setup.sh` also accepts the following options:

`-batch` *`filename`*	Installs your TimesTen product without having to respond to prompts. If *`filename`* is specified, the installation reads all installation prompts from the file. The batch file *`filename`* is optional. However, TimesTen recommends that you create the batch file and specifically indicate the instance name of the installation. If no batch file is provided or if the batch file does not contain an instance name, TimesTen installs a default instance, using "tt51" for the instance name. If an instance with the same name already exists on the installation machine, the install procedure fails. On 64-bit platforms, the batch file must also specify either the 32-bit and 64-bit version of TimesTen be installed. If no batch file is provided or not platform is specified in the batch file, the 32-bit version is installed in the default instance.
`-record` *`filename`*	Installs your TimesTen product and records responses to prompts described in *filename*. This option cannot be used with the TimesTen uninstaller.
`-doc`	Installs documentation.
`-help`	Displays the help message.
`-verbose`	Displays extra installation information.

The CD contains tar files of TimesTen. If the setup script cannot find the tar files to extract from, it prompts you for their location.

4. Enter your response to the setup script prompts.

Note: To install TimesTen without having to respond to prompts, use the `-batch` flag with the `setup.sh` script. Batch files from releases older than TimesTen Release 6.0 should not be used to install this release. All new prompts in the installation script for this release are assigned default answers and may produce unexpected results.

The script performs these actions (unless your answers resulted in termination of the installation process):

- On 64-bit systems, prompts you to install one of the following releases:
  - 32-bit (default)
  - 64-bit
- Prompts you for the location of your TimesTen installation, if installing as a non-root user.
- Prompts you to determine if Access Control should be enabled, except for Client-only installs. Default answer is "No." In that case, no other changes are needed to your installation or your use of TimesTen. For more details on Access Control, see Chapter 1, "Access Control and non-root installations in this guide.
- Prompts you to choose whether to upgrade an existing instance of TimesTen, or to install a new instance of TimesTen.
- Prompts you to specify the daemon port number. If no instances of TimesTen are installed on the machine, or if no instances use the default port number 16000 for 32-bit installations and 16001 for 64-bit applications, prompts you to use the default port number.
- May prompt you to enter an instance name. See "Installation instances" on page 20.
- Prompts you to install your TimesTen product:
  - TimesTen/DataServer or
  - TimesTen/Cache
- Prompts you to install one of the following components.
  - TimesTen Client, Server and Data Manager
  - TimesTen Data Manager only
  - TimesTen Client only
- Removes any previous installation of this release of TimesTen if you are installing an upgrade.
- Untars the appropriate tar file for the component(s) being installed into the install directory, by default `/opt/TimesTen/tt51`.
- Copies the daemon scripts into the appropriate directories.
- If installed by user `root`, configures the system to start the daemon when the system boots.
- Creates the directory where data stores created by the TimesTen demo applications will reside, `/var/TimesTen/`*`TTinstance`*`/DemoDataStore`.
- If you are installing TimesTen/Cache, prompts you for the location of your Oracle version 9i or Oracle Database 10g installation.

- Starts the daemon.
- If there are other instances of the same patch release of TimesTen installed on the same machine, prompts you to provide a unique port to be used by the TimesTen daemon.
- If the TimesTen Server is being installed, prompts you to configure the Server: server name and logging options.
- Prompts you to install the TimesTen documentation.

The daemon writes a `timestend.pid` file into the directory the daemon was started from: `/var/TimesTen/`*`TTinstance`*`/` if installed by the user `root` or *`install_dir`*`/info` if installed by a non-root user.

This file contains the daemon's process ID. When the script to stop the daemon is run, this ID is used to determine which process to terminate. Once the process is terminated, the `timestend.pid` file is removed.

Note: When doing any compiling, use an ANSI C compiler.

Working with the TimesTen daemon and Server

The TimesTen daemon starts automatically when the operating system is booted and operates continually in the background. Application developers do not interact with the daemon (`timestend`) directly; no application code runs in the daemon and application developers do not, in general, have to be concerned with it. Application programs that use TimesTen data stores communicate with the daemon transparently by using TimesTen internal routines.

There are situations, however, when you may have to start and stop the daemon manually, using the TimesTen main daemon startup script. This section explains how to start and stop the daemon. If you have installed the TimesTen Server, it starts automatically when the TimesTen daemon is started and stops automatically when the TimesTen daemon is stopped.

Note: You must have root privileges or be the TimesTen instance administrator to interact with the TimesTen daemon.

If you installed TimesTen as root, the daemon startup file on HP-UX is:

`/etc/rc.config.d/tt_`*`TTinstance`*

If you installed TimesTen as a non-root user, It is:

install_dir/startup

To stop the daemon manually, enter the command:

```
# /sbin/init.d/tt_TTinstance stop
```

or use the utility command `ttDaemonAdmin -stop`

To start the daemon manually, enter the command:

```
# /sbin/init.d/tt_TTinstance start
```

or use the utility command `ttDaemonAdmin -start`

Uninstalling TimesTen

To uninstall TimesTen, follow these steps:

1. Log in as the TimesTen instance administrator if installing as non-root, or
   Log in as root at the shell prompt:

   ```
   % su -
   ```

2. The TimesTen setup script is in the `install_dir/bin` directory. Run the script with the `-uninstall` in a directory outside of the installation instance flag by typing:

   ```
   # install_dir/bin/setup.sh -uninstall
   ```

   Uninstalling the system removes all TimesTen libraries and executables and also stops and uninstalls the daemon and Server. You can execute `ps` to verify that all TimesTen processes have terminated. To verify that TimesTen has been successfully uninstalled, verify that the *install_dir* no longer exists.

Installing TimesTen on HP-UX Memory Windows

Use a separate instance for each memory window

An instance of TimesTen can run in a memory window. A separate instance of TimesTen is required for each memory window. During installation, the TimesTen installer prompts you to indicate whether this instance is to be run in a memory window.

For a memory windows installation, the installer appends the instance name and port number of the daemon to `/etc/services.window` allowing the instance name to be used as a key to the `getmemwindow(1M)` command. Use the `getmemwindow <instance>` command to determine which port is being used by the instance.

Using TimesTen in a memory window

In order to use a TimesTen instance running in a memory window, you must launch your application using the HP-UX `setmemwindow(1M)` command.

For example, given instance `tt_ins1`, use:

```
% setmemwindow -j -i `getmemwindow tt_ins1` <prog>
```

TimesTen utilities are used without the `setmemwindow` command, for example:

```
% ttBackup ...
```

Address Space Considerations

The maximum size for any one data store remains 1GB with 32-bit TimesTen.

TimesTen allocates a single shared memory segment per data store. TimesTen may also allocate shared memory segments when configured to use the shared memory IPC mechanism for client/server.

The daemon and utility programs (programs) provided by TimesTen are linked with `EXEC_MAGIC`, using the `-N` option to `ld(1)`. You may change the TimesTen programs to be marked `SHMEM_MAGIC`, enabling 2GB of shared memory within the window. Any single data store is still limited to 1GB.

For example, to use `SHMEM_MAGIC`, log in as root and use:

```
# chatr -M tt_instance/bin/timesten* tt_instance/bin/
*Cmd tt_instance/bin/ttcserver
```

To return to EXEC_MAGIC, use:

```
# chatr -N tt_instance/bin/timesten* tt_instance/  bin/
*Cmd tt_instance/bin/ttcserver
```

To determine if a program is SHMEM_MAGIC or EXEC_MAGIC, use

```
# chatr binary
```

The `chatr(1M)` command prints “normal executable” for `EXEC MAGIC` programs. It prints “SHMEM_MAGIC” for programs so marked.

Note: If the TimesTen programs are marked `SHMEM_MAGIC`, the user application must be marked `SHMEM_MAGIC` also. Failure to mark the application `SHMEM_MAGIC` may result with an Invalid Argument error (`EINVAL, errno=22`) when attempting to connect to TimesTen.

- If a connection is made to a data store with **ExclAccess=1**, then memory windows will not be used. In this case, TimesTen does not allocate shared memory but rather space for the data store is allocated from the process' private data space.

Troubleshooting

TimesTen support may ask for all of the following in order to diagnose a problem using memory windows.

- How many memory windows do you have configured?

  ```
  % /usr/sbin/kmtune -q max_mem_windows
  ```

- What is the maximum shared memory segment size?

  ```
  % /usr/sbin/kmtune -q shmmax
  ```

- How many windows are you using?

  ```
  % cat /etc/services.window
  ```

- Do you have the correct instance in your path?

  ```
  % ttVersion
  % ttStatus
  % getmemwindow tt_instance
  ```

- Can you connect with a utility provided by TimesTen?

  ```
  % ttIsql -connStr dsn=my_dsn
  ```

- Can you successfully run a demo program? The TimesTen demos are located under`/opt/TimesTen/`*`tt_instance`*`/demo/`

- What other segments are in use?

  ```
  % ipcs -m -a
  ```

- Does "`setmemwindow(1M)`" or a TimesTen utility such as ttStatus return silently when you expected output?

- Check the error status from the "`setmemwindow`" command.

- What does the "`memwin_stats`" tool show?

  ```
  % memwin_stats -w
  ```

  The `memwin_stats` tool may be downloaded from HP at `ftp://contrib:9unsupp8@hprc.external.hp.com/`

- What error are you getting when you try to connect?

The following list is not exhaustive but may help sort out the problem.

- Not enough core (`ENOMEM, errno=12`) indicates a problem allocating the requested amount of shared memory. Can you attach with small **PermSize** and **TempSize** attributes?

- Shared memory can be fragmented. Sometimes, you can attach with increasingly larger segments until you allocate what you want. Are you attempting to allocate more than 1GB within your window (2GB if using `SHMEM_MAGIC`)?
- Permission Denied (`EACCES, errno=13`) indicates that you are attempting to attach to the wrong instance or are pointing to the wrong memory window. Which `-i` argument is passed to `setmemwindow(1M)`?
- Invalid Argument (`EINVAL, errno=22`) indicates that the shared segment may have been allocated in another quadrant. Did you mark the TimesTen programs `SHMEM_MAGIC`? Did you also mark your application `SHMEM_MAGIC`?
- No space left on device (`ENOSPC, errno=28`) may indicate that the system is not configured for enough shared memory segments or identifiers or that the system may have insufficient swap space to allocate the shared segment. Check the values of `shmseg`, `shmmni`, `maxswapchunks` and run the `swapinfo(1M)` command.

Installing TimesTen on AIX systems

This section discusses installation and some related topics for AIX systems.

Installing TimesTen

Before you can install the TimesTen software, you have to add and mount the CD-ROM file system. To add the CD-ROM setup and install TimesTen:

1. Log in as the TimesTen instance administrator if installing as non-root, or
   Log in as root at the shell prompt.
1. Insert the CD-ROM into the CD-ROM drive.
2. Enter:

   ```
   # crfs -v cdrfs -p ro -d cd0 -m /usr/cdrom/
       TimesTen6.0
   ```

   This creates the directory into which you will mount the CD-ROM.
3. To mount the CD-ROM, enter:

```
# mount /usr/cdrom/TimesTen6.0
```

After the CD-ROM setup is complete, you can install TimesTen as follows:

4. Still logged in as user root or the TimesTen instance administrator, run the setup script by typing:

```
# cd mount_dir
# ./setup.sh
```

where *mount_dir* is the directory where the CD is mounted (e.g.: `/usr/cdrom/TimesTen6.0`).

You can run the `setup` script with the option `-install` or `-uninstall` (default is `install`). When you use the `-uninstall` option, the script stops the daemon if it is running and removes all files it had installed. In addition, `setup.sh` also accepts the following options:

`-batch` *filename*	Installs your TimesTen product without having to respond to prompts. If *filename* is specified, the installation reads all installation prompts from the file. The batch file *filename* is optional. However, TimesTen recommends that you create the batch file and specifically indicate the instance name of the installation. If no batch file is provided or if the batch file does not contain an instance name, TimesTen installs a default instance, using "tt51" for the instance name. If an instance with the same name already exists on the installation machine, the install procedure fails. On 64-bit platforms, the batch file must also specify either the 32-bit and 64-bit version of TimesTen be installed. If no batch file is provided or not platform is specified in the batch file, the 32-bit version is installed in the default instance.
`-record` *filename*	Installs TimesTen and records responses to prompts described in *filename*. This option cannot be used with the TimesTen uninstaller.
`-doc`	Installs documentation.
`-help`	Displays the help message.
`-verbose`	Displays extra installation information.

The CD contains tar files of TimesTen. If the `setup` script cannot find the tar files to extract from, it prompts you for their location.

5. Enter your response to the `setup` script prompts.

Note: To install TimesTen without having to respond to prompts, use the `-batch` flag with the `setup.sh` script. Batch files from releases older than TimesTen Release 6.0 should not be used to install this release. All new prompts in the installation script for this release are assigned default answers and may produce unexpected results.

The file script performs these actions (unless your answers resulted in termination of the installation process):

- Prompts you to choose whether to upgrade an existing instance of TimesTen, or to install a new instance of TimesTen.
- Prompts you for the location of your TimesTen installation, if installing as a non-root user.
- Prompts you to determine if Access Control should be enabled, except for Client-only installs. Default answer is "No." In that case, no other changes are needed to your installation or your use of TimesTen. For more details on Access Control, see Chapter 1, "Access Control and non-root installations in this guide.
- May prompt you to enter an instance name. See "Installation instances" on page 20.
- Prompts you to specify the daemon port number. If no instances of TimesTen are installed on the machine, or if no instances use the default port number 16000 for 32-bit installations and 16001 for 64-bit applications, prompts you to use the default port number.
- Prompts you to install your TimesTen product:
  - TimesTen/DataServer or
  - TimesTen/Cache
- Prompts you to install one of the following components.
  - TimesTen Client, Server and Data Manager
  - TimesTen Data Manager only
  - TimesTen Client only
- Removes any previous installation of this release of TimesTen.
- Untars the appropriate tar file into the install directory, by
- default `/usr/lpp/TimesTen/`tt51.
- Copies the daemon scripts into the appropriate directories.

- Creates the directory where data stores created by the TimesTen demo applications will reside, /var/TimesTen/*TTinstance*/DemoDataStore)
- If installed as root, defines the TimesTen 6.0 subsystem as part of the system configuration, so the daemon starts. (If installed by a non-root user, the root user must run the setuproot script after installation to configure your system.)
- If you are installing TimesTen/Cache, prompts you for the version of your Oracle installation (Oracle version 9i or Oracle Database 10g) and its location. A TimesTen/Cache library is installed for your particular version of Oracle.
- Starts the daemon.
- If there are other instances of the same patch release of TimesTen installed on the same machine, prompts you to provide a unique port to be used by the TimesTen daemon.
- If the TimesTen Server is being installed, prompts you to configure the Server: server name and logging options.
- Prompts you to install the TimesTen documentation.

The daemon writes a timestend.pid file into the directory the daemon was started from: /var/TimesTen/*TTinstance*/ if installed by the user root or *install_dir*/info if installed by a non-root user.

This file contains the daemon's process ID. When the script to stop the daemon is run, this ID is used to determine which process to terminate. Once the process is terminated, the timestend.pid file is removed.

Working with the TimesTen daemon and server

The TimesTen daemon starts automatically when the operating system is booted and operates continually in the background. Application developers do not interact with timestend directly; no application code runs in the daemon and application developers do not, in general, have to be concerned with it. Application programs that use TimesTen data stores communicate with the daemon transparently by using TimesTen internal routines.

There are situations, however, when you may have to start and stop the daemon manually, using the TimesTen main daemon startup script. This section explains how to start and stop the daemon. If you have installed the TimesTen Server, it starts automatically when the TimesTen daemon is started and stops automatically when the TimesTen daemon is stopped.

Note: You must have root privileges or be the TimesTen instance administrator to interact with the TimesTen daemon.

To stop the daemon manually, enter the command:

```
# stopsrc -s tt_TTinstance
```

or use the utility command `ttDaemonAdmin -stop`

To start the daemon manually, enter the command:

```
# startsrc -s tt_TTinstance
```

or use the utility command `ttDaemonAdmin -start`

To determine the status of the daemon at any time, enter the command:

```
# lssrc -s tt_TTinstance
```

Uninstalling TimesTen

To uninstall TimesTen, follow these steps:

1. Log in as the TimesTen instance administrator if installing as non-root, or
   Log in as root at the shell prompt:

   ```
   % su -
   ```

2. The TimesTen `setup` script is in the *`install_dir`*`/bin` directory. Run the script with the `-uninstall` option in a directory outside of the installation instance:
   ```
   # install_dir/bin/setup.sh -uninstall
   ```

   Uninstalling the system removes all TimesTen libraries and executables and also stops and uninstalls the daemon. You can execute `ps` to verify that all TimesTen processes have terminated. To verify that TimesTen has been successfully uninstalled, check to see that the *`install_dir`* no longer exists.

Installing TimesTen on Linux systems

This section discusses installation and some related topics for Linux systems.

Note: Before beginning installation, be sure that the prerequisites defined in "Installation prerequisites" on page 22 have been met.

Installing TimesTen

To install TimesTen on your Linux system, follow these steps:

1. Log in as the TimesTen instance administrator if installing as non-root, or
   Log in as root at the shell prompt:

   ```
   % su -
   ```

2. Load the CD-ROM into the CD drive as follows:

   ```
   # mount/mnt/cdrom
   ```

3. Run the setup script by typing the following:

   ```
   # cd /mnt/cdrom
   # ./setup.sh
   ```

   You can run the setup script with the option `-install` or `-uninstall` (default is `-install`). When you use the `-uninstall` option, the script stops the daemon and Server if they are running and removes all files it had installed. In addition, `setup.sh` also accepts the following options:

`-batch` *`filename`*	Installs your TimesTen product without having to respond to prompts. If *`filename`* is specified, the installation reads all installation prompts from the file. The batch file *`filename`* is optional. However, TimesTen recommends that you create the batch file and specifically indicate the instance name of the installation. If no batch file is provided or if the batch file does not contain an instance name, TimesTen installs a default instance, using "tt51" for the instance name. If an instance with the same name already exists on the installation machine, the install procedure fails.
`-record` *`filename`*	Installs your TimesTen product and records responses to prompts described in *`filename`*. This option cannot be used with the TimesTen uninstaller.
`-doc`	Installs documentation.
`-help`	Displays the help message.
`-verbose`	Displays extra installation information.

The CD contains tar files of TimesTen. If the setup script cannot find the tar files to extract from, it prompts you for their location.

4. Enter your response to the setup script prompts.

Note: To install TimesTen without having to respond to prompts, use the `-batch` flag with the `setup.sh` script. Batch files from releases older than TimesTen Release 6.0 should not be used to install this release. All new prompts in the installation script for this release are assigned default answers and may produce unexpected results.

The setup script performs these actions (unless your answers resulted in termination of the installation process):

- Prompts you for the location of your TimesTen installation, if installing as a non-root user.
- Prompts you to determine if Access Control should be enabled, except for Client-only installs. Default answer is "No." In that case, no other changes are needed to your installation or your use of TimesTen. For more details on Access Control, see Chapter 1, "Access Control and non-root installations in this guide.
- Prompts you to choose whether to upgrade an existing instance of TimesTen, or to install a new instance of TimesTen.
- May prompt you to enter an instance name. See "Installation instances" on page 20.
- Prompts you to specify the daemon port number. If no instances of TimesTen are installed on the machine, or if no instances use the default port number 16000 for 32-bit installations and 16001 for 64-bit applications, prompts you to use the default port number.
- Prompts you to install your TimesTen product:
  - TimesTen/DataServer or
  - TimesTen/Cache
- Prompts you to install one of the following components.
  - TimesTen Client, Server and Data Manager
  - TimesTen Data Manager only
  - TimesTen Client only
- Removes any previous installation of this release of TimesTen if you are installing an upgrade.
- Untars the appropriate tar file for the component(s) being installed into the install directory, by default `/opt/TimesTen/tt51`.
- Copies the daemon scripts into the appropriate directories.

- If installed by user `root`, configures the system to start the daemon when the system boots.
- Creates the directory where data stores created by the TimesTen demo applications will reside. By default they reside in `/var/TimesTen/`*`TTinstance`*`/DemoDataStore`.
- Starts the daemon.
- If there are other instances of the same patch release of TimesTen installed on the same machine, prompts you to provide a unique port to be used by the TimesTen daemon.
- If the TimesTen Server is being installed, prompts you to configure the Server: server name, port number and logging options.
- Prompts you to install the TimesTen documentation.

The daemon writes a `timestend.pid` file into the directory the daemon was started from: `/var/TimesTen/`*`TTinstance`*`/` if installed by the user `root` or *`install_dir`*`/info` if installed by a non-root user.

This file contains the daemon's process ID. When the script to stop the daemon is run, this ID is used to determine the process to terminate. When the process terminates, the `timestend.pid` file is removed.

Working with the TimesTen daemon and Server

The TimesTen daemon (`timestend`) starts automatically when the operating system is booted and operates continually in the background. Application developers do not interact with `timestend` directly; no application code runs in the daemon and application developers do not, in general, have to be concerned with it. Application programs that use TimesTen data stores communicate with the daemon transparently by using TimesTen internal routines.

There are situations, however, when you may have to start and stop the daemon manually, using the TimesTen main daemon startup script. This section explains how to start and stop the daemon. If you have installed the TimesTen Server, it starts automatically when the TimesTen daemon is started and stops automatically when the TimesTen daemon is stopped.

Note: You must have root privileges or be the TimesTen instance administrator to interact with the TimesTen daemon.

To stop the daemon manually, enter the command:

```
# /etc/rc.d/init.d/tt_TTinstance stop
```

or use the utility command `ttDaemonAdmin -stop`

To start the daemon manually, enter the command:

```
# /etc/rc.d/init.d/tt_TTinstance start
```

or use the utility command `ttDaemonAdmin -start`

Uninstalling TimesTen

To uninstall all TimesTen components, follow these steps:

1. Log in as the TimesTen instance administrator if you installed as non-root, or
   Log in as root at the shell prompt:

   ```
   % su -
   ```

2. The TimesTen setup script is in the *install_dir*`/bin` directory. Run the script with the `-uninstall` flag in a directory outside of the installation instance, by typing:

   ```
   # install_dir/bin/setup.sh -uninstall
   ```

   Uninstalling the system removes all TimesTen libraries and executables and also stops and uninstalls the daemon and Server. You can execute `ps` to verify that all TimesTen processes have terminated. To verify that TimesTen has been successfully uninstalled, verify that the *install_dir* no longer exists.

Installing TimesTen on Tru64 UNIX systems

TRU 64

This section discusses installation and some related topics for Tru64 UNIX systems.

Note: Before beginning installation, be sure that the prerequisites defined in "Installation prerequisites" on page 22 have been met.

Installing TimesTen

To install TimesTen on your Tru64 UNIX system, follow these steps:

1. Log in as the TimesTen instance administrator if installing as non-root, or
   Log in as root at the shell prompt:

   ```
   % su -
   ```

2. Load the CD-ROM into the CD drive.
3. Create the mount directory, if it does not already exist:

   ```
   # mkdir /mnt
   ```

4. Mount the CD-ROM:

   ```
   # /sbin/mount -r -t cdfs /dev/disk/cdrom0c /mnt
   ```

5. Run the setup script by typing the following:

   ```
   # cd /mnt
   # ./setup.sh
   ```

   You can run the setup script with the option `-install` or `-uninstall` (default is `-install`). When you use the `-uninstall` option, the script stops the daemon if it is running and removes all files it had installed.

Note: To uninstall TimesTen, you must run `setup.sh -uninstall` in a directory outside of the installation instance that you wish to uninstall. For example to uninstall the default instance run `/opt/TimesTen/tt51/bin/setup.sh -uninstall`.

In addition, `setup.sh` also accepts the following options:

`-batch` *`filename`*	Installs your TimesTen product without having to respond to prompts. If *`filename`* is specified, the installation reads all installation prompts from the file. The batch file *`filename`* is optional. However, TimesTen recommends that you create the batch file and specifically indicate the instance name of the installation. If no batch file is provided or if the batch file does not contain an instance name, TimesTen installs a default instance, using "tt51" for the instance name. If an instance with the same name already exists on the installation machine, the install procedure fails.
`-record` *`filename`*	Installs your TimesTen product and records responses to prompts described in *`filename`*. This option cannot be used with the TimesTen uninstaller.
`-doc`	Installs documentation.

`-help`	Displays the help message.
`-verbose`	Displays extra installation information.

The CD contains tar files of TimesTen. If the setup script cannot find the tar files to extract from, it prompts you for their location.

6. Enter your response to the setup script prompts.

Note: To install TimesTen without having to respond to prompts, use the `-batch` option with the `setup.sh` script. Batch files from releases older than TimesTen Release 6.0 should not be used to install this release. All new prompts in the installation script for this release are assigned default answers and may produce unexpected results.

The setup script performs these actions (unless your answers resulted in termination of the installation process):

- Prompts you to choose whether to upgrade an existing instance of TimesTen, or to install a new instance of TimesTen.
- Prompts you for the location of your TimesTen installation, if installing as a non-root user.
- Prompts you to determine if Access Control should be enabled, except for Client-only installs. Default answer is "No." In that case, no other changes are needed to your installation or your use of TimesTen. For more details on Access Control, see Chapter 1, "Access Control and non-root installations in this guide.
- Prompts you to specify the daemon port number. If no instances of TimesTen are installed on the machine, or if no instances use the default port number 16001.
- May prompt you to enter an instance name. See "Installation instances" on page 20.
- Prompts you to install your TimesTen product:
  - TimesTen/DataServer or
  - TimesTen/Cache
- Prompts you to install one of the following components.
  - TimesTen Client, Server and Data Manager
  - TimesTen Data Manager only
  - TimesTen Client only
- Removes any previous installation of this release of TimesTen if you are installing an upgrade.

- Untars the appropriate tar file for the component(s) being installed into the install directory, by default `/opt/TimesTen/tt51`.
- Copies the daemon scripts into the appropriate directories.
- If installed by user `root`, configures the system to start the daemon when the system boots.
- Creates the directory where data stores created by the TimesTen demo applications will reside, `/var/TimesTen/`*`TTinstance`*`/DemoDataStore`.
- If you are installing TimesTen/Cache, prompts you for the location of your Oracle version 9i or Oracle Database 10g installation.
- Starts the daemon.
- If there are other instances of the same patch release of TimesTen installed on the same machine, prompts you to provide a unique port to be used by the TimesTen daemon.
- If the TimesTen Server is being installed, prompts you to configure the Server: server name and logging options.
- Prompts you to install the TimesTen documentation.

The daemon writes a `timestend.pid` file into the directory the daemon was started from: `/var/TimesTen/`*`TTinstance`*`/` if installed by the user `root` or *`install_dir`*`/info` if installed by a non-root user.

This file contains the daemon's process ID. When the script to stop the daemon is run, this ID is used to determine the process to terminate. When the process terminates, the `timestend.pid` file is removed.

Working with the TimesTen daemon and Server

The TimesTen daemon (`timestend`) starts automatically when the operating system is booted and operates continually in the background. Application developers do not interact with the daemon directly; no application code runs in the daemon and application developers do not, in general, have to be concerned with it. Application programs that use TimesTen data stores communicate with the daemon transparently by using TimesTen internal routines.

There are situations, however, when you may have to start and stop the daemon manually, using the TimesTen main daemon startup script. This section explains how to start and stop the daemon. If you have installed the TimesTen Server, it starts automatically when the TimesTen daemon is started and stops automatically when the TimesTen daemon is stopped.

Note: You must have root privileges or be the TimesTen instance administrator to interact with the TimesTen daemon.

To stop the daemon manually, enter the command:

```
# /sbin/init.d/tt_TTinstance stop
```

or use the utility command `ttDaemonAdmin -stop`

To start the daemon manually, enter the command:

```
# /sbin/init.d/tt_TTinstance start
```

or use the utility command `ttDaemonAdmin -start`

Uninstalling TimesTen

To uninstall all TimesTen components:

1. Log in as the TimesTen instance administrator if you installed as non-root, or
   Log in as root at the shell prompt:

   ```
   % su -
   ```

2. The TimesTen setup script is in the *install_dir*`/bin` directory. Run the script with the `-uninstall` flag in a directory outside of the installation instance, by typing:

   ```
   # install_dir/bin/setup.sh -uninstall
   ```

   Uninstalling the system removes all TimesTen libraries and executables and also stops and uninstalls the daemon. You can execute `ps` to verify that all TimesTen processes have terminated. To verify that TimesTen has been successfully uninstalled, verified that the *install_dir* no longer exists.

Using the Cache Administrator

The Cache Administrator is a web-based tool used to set cache definitions. This feature is available on systems where TimesTen/Cache has been installed. See "Cache Connect to Oracle" on page 19.

To start the Cache Administrator use the URL:

```
http://machine_name:port/cache
```

`machine_name` is the host name of the machine where the TimesTen daemon or Data Manager service is running, or localhost if using a web browser on the same machine where TimesTen is installed.

`port` is the TimesTen daemon webserver port number that was configured during the installation of TimesTen/Cache. Its value is stored in the PORT variable in the `/var/TimesTen/TTinstance/webserver.config` file on UNIX systems or the `install_dir\srv\info\webserver.config` on Windows systems.

For details on setting the environment variables required to use Cache Connect to Oracle, see "Environment modifications" on page 73.

For details on setting up the webserver, see "Web server configuration" on page 78.

The following Web browsers are supported for the Cache Administrator:

- Internet Explorer 6.0
- Firefox 1.0 and later

Note: If you use Firefox for the Cache Administrator, turn the password manager off. Otherwise you cannot see Cache Administrator menu items after connecting to a data store.

Informational messages on Windows systems

As the TimesTen Data Manager service operates, it generates error, warning, informational and debug messages. These messages may be useful for TimesTen system administration and for debugging applications.

To view the messages, follow these steps:

1. On Windows XP, choose **Start > Programs > Administrative Tools > Event Viewer**.
   On Windows2000, choose **Start > Settings > Control Panel > Administrative Tools > Event Viewer**.
   The Event Viewer window appears.
2. From the Log menu, choose **Application**. The window changes to display only log messages generated by applications.
   - Messages with the phrase "**TimesTen Data Manager 6.0**" in the "Source" column were generated by the TimesTen Data Manager service.
   - Messages with the phrase "**TimesTen Server 6.0**" in the "Source" column were generated by the TimesTen Server service.

- Messages with the phrase "**TimesTen Replication 6.0**" in the "Source" column were generated by the TimesTen Replication Agent.

3. To view a TimesTen message, double-click it. This displays the message window.
4. Click **Next** or **Previous** to view additional messages.

Note: You can also use the **ttDaemonLog** utility to view messages logged by the TimesTen Data Manager. For a description of the system administration utilities, see "TimesTen Utilities" of the *TimesTen Reference Guide*.

Informational messages on UNIX systems

As the TimesTen daemon operates, it generates error, warning, informational and debug messages for TimesTen system administration and for debugging applications.

By default, TimesTen logs daemon messages using the `LOG_USER` facility defined by `syslog`.

To specify the syslog facility used to log TimesTen Daemon and subdaemon messages, on a separate line of the `ttendaemon.options` file add:

`-facility` *`name`*

Possible *`name`* values are: `auth`, `cron`, `daemon`, `local0-local7`, `lpr`, `mail`, `news`, `user`, or `uucp`.

The `syslog` facility allows messages to be routed in a variety of ways, including recording them to a file. The disposition of messages is under the control of the configuration file, `/etc/syslog.conf`

Entries in the `syslog.conf` file contain two columns. The first column contains a list of the types of messages to log to a particular file. The second column contains the name of the log file. A tab appears between the message type and file name. Each entry in the `syslog.conf` file has the format:
message_type file_name.
Message types are specified in two parts:

subsystem-facility.severity-level

Depending on the configuration specified in that file, messages can be logged into various files. For the TimesTen daemon, specify the message types: `user.debug`, `user.info`, `user.warn` and `user.err`.

You can also use the wildcard character * to represent the subsystem-facility. Since debug messages are ranked highest, specifying `*.debug` or `user.debug` is sufficient in preparing a file for the daemon log. In a message type list, delimit items by semi-colons. For example:

```
*.debug /var/adm/syslog/syslog.log
user.err; user.warn; user.info /var/adm/messages
```

To make changes to `/etc/syslog.conf`, you must have root privileges or be the TimesTen instance administrator. Changes only take effect after the `syslog` daemon (`syslogd`) process is terminated (with the command `kill -1`) and restarted.

For further details, see your operating system's documentation for `syslog.conf` or `syslogd` for information on configuring this file.

Note: If the `/etc/syslog.conf file` does not exist on your system, create one according to the `syslog.conf` manual page so the daemon can log its data to the `syslog` facility.

To check whether or not your configuration file is set up correctly, run the TimesTen **ttSyslogCheck** utility. Finally, once `syslogd` has been set up correctly, you may use the TimesTen **ttDaemonLog** utility to view only those messages in the system log file that TimesTen logged.

ODBC installation

On Windows systems, TimesTen makes use of the Microsoft ODBC 3.5 SDK. The ODBC SDK's redistributable components are installed in `C:\WINDOWS\SYSTEM32` on Windows systems.
Microsoft only permits TimesTen to redistribute portions of the ODBC SDK; those portions are installed automatically (if they are not already present). Other components—Microsoft sample programs, online help files, and C language header files—are provided as part of the Microsoft ODBC SDK, which can be installed separately as required.
Additionally, the ODBC C language header files and ODBC online help are bundled as part of Microsoft Visual C++ 6.0 and Microsoft Visual Studio .NET. Most TimesTen developers do not need to install the SDK separately.

On UNIX systems, no separate SDK installation is required.

Environment modifications

This section describes various environment variables that you may need to set, depending on the features of TimesTen that your application uses.

The following table summarizes, in alphabetical order, the environment variables detailed in this section and other parts of this guide. Some of these environment variables are platform specific.

Environment Variable	What to include	For settings and other information, see:
CLASSPATH	Set to the location of the JDK to be used by your Java applications	"CLASSPATH environment variable" on page 76 and "Using the Cache Administrator" on page 70.
LIB, LIBPATH, LD_LIBRARY_PATH or SHLIB_PATH	On UNIX systems, include the library directory under the TimesTen installation directory	"Shared library path environment variable" on page 77.
ODBCINI	The location where the odbc.ini file used by TimesTen data stores is to be found.	"ODBCINI environment variable" on page 75
ORACLE_HOME	If using TimesTen/Cache, set to the location of the Oracle installation. Required if you are using TimesTen/Cache.	"ORACLE_HOME environment variable" on page 77 and "Using the Cache Administrator" on page 70
PATH	Include the bin directory under the TimesTen installation directory. On Windows, also include the path to the Oracle installation if you are using TimesTen/Cache.	"PATH environment variable" on page 75, "Shared library path environment variable" on page 77 and "Installing TimesTen on Windows systems" on page 41.
SYSODBCINI	Set to the location where the sys.odbc.ini file used by TimesTen system data stores is to be found. This environment variable should be set in the start-up script.	"SYSODBCINI environment variable" on page 76

Environment Variable	What to include	For settings and other information, see:
SYSTTCONNECTINI	Set to the location where the sys.ttconnect.ini file used by TimesTen Client applications to define logical server names.	"SYSTTCONNECTINI environment variable" on page 76
TMP or TMPDIR	Set to the location of the temporary directory. TimesTen uses this directory during recovery and other operations.	"Disk space requirements" on page 30

PATH environment variable

TimesTen provides utilities for managing and debugging TimesTen applications. To make these utilities readily available, include the bin directory found in *install_dir* in the PATH environment variable.

Note: *install_dir* is the directory where TimesTen is installed.

WINDOWS

On Windows, the PATH environment variable must contain the bin directory of the ORACLE installation, if you are using TimesTen/Cache.

ODBCINI environment variable

UNIX

TimesTen applications use the odbc.ini file to define data sources and their data store attributes. (For a description of data store attributes, see Chapter 1, "Data Store Attributes in the *Oracle TimesTen In-Memory Database API and SQL Reference Guide*.) By default on UNIX platforms, TimesTen first looks for the .odbc.ini file in the home directory of the user running the TimesTen application. To override the name and location of this file at run-time, set the $ODBCINI environment variable to the pathname of a .odbc.ini file before launching the TimesTen application. If TimesTen cannot locate a user DSN file, the system DSN file located in /var/TimesTen/sys.odbc.ini will be used. Also, see "Defining data sources for the demo applications" on page 83 for more information on the .odbc.ini file. For non-root installations, TimesTen also looks for the sys.odbc.ini file under *install_dir*/info.

SYSODBCINI environment variable

UNIX

TimesTen applications use the sys.odbc.ini file to define system data sources and their data store attributes. (For a description of data store attributes, see Chapter 1, "Data Store Attributes" in the *Oracle TimesTen In-Memory Database API and SQL Reference Guide*.) A system data source can be used by any user on the machine. On Windows, system DSNs are defined from the **System DSN** tab of the ODBC Data Source Administrator. On UNIX, system DSNs are defined in the file `/var/TimesTen/sys.odbc.ini`. To override the name and location of this file at run-time, set the `$SYSODBCINI` environment variable to the pathname of a `sys.odbc.ini` file before launching the TimesTen application.

If TimesTen cannot locate a user DSN file, the system DSN file located in `/var/TimesTen/sys.odbc.ini` will be used. For non-root installations, TimesTen also looks for the `sys.odbc.ini` file under *`install_dir`*`/info`.

Also, see "Defining data sources for the demo applications" on page 83 for more information on the `.odbc.ini` file.

SYSTTCONNECTINI environment variable

UNIX

TimesTen client applications use the `sys.ttconnect.ini` file to define logical server names. For a description of logical server names, see Chapter 3, "Working with the TimesTen Client and Server" in the *Oracle TimesTen In-Memory Database Operations Guide*. By default on UNIX platforms, TimesTen looks in `/var/TimesTen/sys.ttconnect.ini`. To override the name and location of this file at run-time, set the `SYSTTCONNECTINI` environment variable before launching the TimesTen Client application.

For non-root installations, TimesTen also looks for the `sys.ttconnect.ini` file under *`install_dir`*`/info`.

WINDOWS

On Windows systems, logical server names can be configured using the **ODBC Data Source Administrator**.

CLASSPATH environment variable

To use the TimesTen JDBC drivers on platforms where JDBC is supported, add: *`install_dir`*`/lib/classes`*`jdk_version`*`.jar` to the CLASSPATH environment variable. For example, for JDK 5.0, set the CLASSPATH environment variable to: *`install_dir`*`/lib/classes15.jar`.

ORACLE_HOME environment variable

On platforms where TimesTen/Cache is supported, to work with Oracle data, the TimesTen Oracle agent must be running. This requires that the ORACLE_HOME environment variable be set to the path of the Oracle 9i or Oracle Database 10g installation.

Shared library path environment variable

On Solaris, and Linux systems, add: *install_dir*/lib directory to the LD_LIBRARY_PATH environment variable.

If you are using TimesTen/Cache, add $ORACLE_HOME/lib to LD_LIBRARY_PATH. See "ORACLE_HOME environment variable" on page 77.

On AIX systems, add *install_dir*/lib directory to the LIBPATH environment variable.

On HP-UX 32-bit systems, add *install_dir*/lib to the SHLIB_PATH environment variable.

On HP-UX 64-bit systems, add *install_dir*/lib to the LD_LIBRARY_PATH environment variable.

On HP-UX systems, if you are using JDBC, also add the *install_dir*/lib directory to the SHLIB_PATH environment variable.

If you are using TimesTen/Cache, SHLIB_PATH must also contain $ORACLE_HOME/lib32 and must not contain $ORACLE_HOME/lib. See "ORACLE_HOME environment variable" on page 77.

On Tru64 UNIX systems, add *install_dir*/lib directory to the LD_LIBRARY_PATH environment variable.

Web server configuration

The TimesTen daemon contains an embedded web server, that is used for the Cache Administrator. If you select to enable the web server at install time, TimesTen enables it by setting the `-webserver` option in the `ttendaemon.options` file.

This file is in the startup directory of the daemon:

On UNIX, if installed as `root`:

`/var/TimesTen/`*`TTinstance`*`/`

if installed by a non-root user:

`install_dir``/info`

On Windows:

`install_dir``\srv\info`

If you have not installed the web server and decide to enable it at a later time, you can do so by:

1. Shutting down the TimesTen daemon.
2. Adding a separate line to the `ttendaemon.options` file that contains the option `-webserver`.
3. Starting the TimesTen daemon.

For more details, see the chapter Chapter 4, "Working with the Oracle TimesTen Data Manager Daemon in the *Oracle TimesTen In-Memory Database Operations Guide*.

Various options for the web server are stored in the `webserver.config` file, also in the daemon startup directory. The TimesTen installation scripts initially set these options. Most of the options should only be changed at the request of TimesTen Customer Support.

The options that you may change without consulting TimesTen Customer Support are `HOSTSALLOW` and `HOSTSDENY`. To change a value, edit the `webserver.config` file with a text editor, change the value, save the file, then restart the TimesTen daemon.

The options in `webserver.config` file are:

PORT — The port on which the web server listens. If you change this, any scripts which start the Cache Administrator, or any links you have saved will have to be changed.

WEBROOT — The root directory of web files.

DOCROOT — A subdirectory of `WEBROOT` where the HTML files are located. The path should begin and end with a '/' on all platforms. The

default is /docs/. DOCROOT is prepended to the path, so if you supply a URL, the webserver will look for a file in the indicated path.

CGIROOT — A subdirectory of WEBROOT where the CGI scripts are located. The path should begin and end with a '/' on all platforms. The default is /cgi-bin/.

PERL — The path to the Perl interpreter. The path is set by the TimesTen installation scripts. Do not change the default path unless you are certain that the path is for a Perl version that is compatible with TimesTen and that it contains all the required libraries. The path should point to the Perl binary, not the Perl directory.

PERLLIB — The path to a directory containing perl modules. It is added to the Perl search path when a perl CGI program is run.

LOG — Specifies how verbose the logging should be. Set to verbose to log each connection.

HOSTSALLOW — Used to allow access from certain hosts (see also **HOSTSDENY** below). To allow access from anywhere, set it to all. To allow only one host access, set it to the host's IP address. To allow access from a set of hosts, set it to the prefix of IP addresses to allow, followed by a period. For example, HOSTSALLOW = 192.232. will allow access from any host whose IP address begins with 192.232. If a host matches both in HOSTSALLOW and HOSTSDENY, it is allowed access. This allows you to create the following:

```
HOSTSALLOW = 192.232.3.
HOSTSDENY = all
```

to deny access to any host unless its IP address begins with 192.232.3.

HOSTSDENY — Used to deny access to certain hosts. The format is the same as HOSTSALLOW (see above).

PASSWORD_FILE — The name of a file containing user names and passwords. If this configuration variable is set, all requests are authenticated. The password file contains lines of the form "username:password" (do not use spaces around the colon, though leading and trailing spaces and comments are allowed). Passwords are not encrypted in the password file, and are sent only base64-encoded from the browser to the server.

MIME — Some MIME types are also specified here. They are all of the form MIMETYPE:.{extension} = {mime type}. You should not remove the definitions for text/html.

Migrating data stores to TimesTen 6.0

TimesTen 6.0 cannot read data stores created with earlier releases of TimesTen. TimesTen 6.0 includes two migration utilities: **ttMigrate** and **ttBulkCp**. These utilities allow you to migrate data stores from older TimesTen releases to TimesTen Release 6.0.

For a description of these utilities, see the chapter "TimesTen Utilities" in the *TimesTen Reference Guide*.

On Windows, **ttMigrate** uses the ODBC driver manager.

On UNIX platforms, the **ttMigrate** utility is directly linked with the TimesTen Data Manager ODBC driver.

Using the ttMigrate utility

The **ttMigrate** utility saves and restores tables from a TimesTen data store in a binary data file. Using **ttMigrate**, you can save an entire data store to a single data file. The data file includes table rows as well as column and index definitions. When TimesTen restores a table in a new data store, it also restores the table's indexes.

Note: The **ttMigrate** utility cannot migrate data stores across different hardware platforms. For example, you cannot migrate a Windows data store to a Solaris data store. The release of **ttMigrate** must also match the release of the data store you are copying from or to. In the example in this section, use **ttMigrate** of the older version to save the tables of the original data store to disk files and use **ttMigrate** of the new version to migrate the files into the tables of the new data store.

For a description of the **ttMigrate** syntax and usage, see the chapter "TimesTen Utilities" in the *TimesTen Reference Guide*.

To migrate a data store from different versions:

1. Use **ttMigrate** to save the tables in the older version data store to a disk file.

   If, for example:

   - A TimesTen 4.5 data store is called `Sales`450;
   - The data file you wish to use is called `sales.dat`; and
   - TimesTen 4.5is installed in `/opt/TimesTen`45 on UNIX platforms or `C:\TimesTen\TimesTen`45 on Windows.

   On Windows, use:

```
C:\ > "C:\TimesTen\TimesTen45\bin\ttMigrate"
  -c DSN=Sales450 sales.dat
```

On UNIX, use:

```
% /opt/TimesTen45/32/bin/ttMigrate -c DSN=Sales450
  sales.dat
```

2. Use **ttMigrate** to restore the saved tables in the new data store.
3. Create a new data source name, `Sales`tt51 for the TimesTen 6.0 data store, and import the 4.5 data store:

   If, for example:

   - TimesTen 6.0 is installed in `/opt/TimesTen/`tt51 on UNIX platforms or `C:\TimesTen\`tt51 on Windows.

On Windows, use:

```
C:\ > "C:\TimesTen\tt51\bin\ttMigrate" -r DSN=Salestt51
sales.dat
```

On UNIX, use:

```
% /opt/TimesTen/tt51/bin/ttMigrate -rx DSN=Salestt51
sales.dat
```

Using the ttBulkCp utility

The **ttBulkCp** utility copies table data between TimesTen data stores and ASCII files. The data files used by **ttBulkCp** can only contain rows from a single table. They also do not store the table's column or index definitions. Therefore, when migrating from one TimesTen data store to another with **ttBulkCp**, you must first create the tables and indexes in the new data store manually. Then use **ttBulkCp** to copy the rows from the original data store to the new data store. For a description of the **ttBulkCp** syntax and usage, see the chapter "TimesTen Utilities" in the TimesTen *Reference Guide*.

Note: The release of **ttBulkCp** must match the release of the data store you are copying from or to. In this example, use **ttBulkCp** Release 4.5 to save the tables to disk files and use **ttIsql** and **ttBulkCp** Release 6.0 to copy the disk files into the tables of the new data store.

To import data from a data store created with TimesTen4.5:

1. Find all the tables you want to copy into the new release of TimesTen.
2. Use the TimesTen utility **ttBulkCp** to copy the data in each table to a disk file.

3. Define a data source name for the new data store.
4. Use the CREATE TABLE and CREATE INDEX commands with **ttIsql** to recreate each table and index you are importing.
5. Use the TimesTen utility **ttBulkCp** to copy the contents of the disk file(s) into the table(s) of the new data store. If, for example:
   - Release 4.5 is installed in: `/opt/TimesTen4.5/32` and release 6.0 is installed in `/opt/TimesTen/tt51`;
   - Your DSN for release 4.5 is called `source450` and your DSN for release 6.0 is `source_tt51`.
   - You have a **ttIsql** script named `create.sql` that creates user tables and indexes, or use the **ttSchema** utility to create the SQL statements necessary for object creation; and
   - You want to migrate the tables ABLE and BAKER from `source450` to `source_tt51`.

   To copy the tables to disk files, you would execute the commands:

```
% /opt/TimesTen45/32/bin/ttBulkCp -o
      DSN=source450 able able.save
% /opt/TimesTen45/32/bin/ttBulkCp -o
      DSN=source450 baker baker.save
```

   Next create a new data source name, `source_tt51` for the TimesTen 6.0 data store, and execute the commands:

```
% /opt/TimesTen/tt51/bin/ttIsql -connStr
      DSN=source_tt51 -f create.sql
% /opt/TimesTen/tt51/bin/ttBulkCp -i
      DSN=source_tt51 able able.save
% /opt/TimesTen/tt51/bin/ttBulkCp -i
      DSN=source_tt51 baker baker.save
```

Building and running the demo applications

Source code for several demo applications is provided in the `demo` directory as part of the TimesTen Data Manager distribution on UNIX and Windows systems. Documentation for these demos is included online in the file *`install_dir`*`/demo/README.TXT` on UNIX or *`install_dir`*`\demo\README.TXT` on Windows.

The directory *`install_dir`*`/demo/tutorial` contains demos that provide examples for the TimesTen documentation. For a description of these demos, see the README.txt file at the top of this directory.

Note: By default, the TimesTen demo applications save data store files to /var/TimesTen/*TTinstance*/demo/DemoDataStore on UNIX. On Windows, you specify the data store directory at installation time. Before running the demos, make sure your temporary directory has a minimum of 100 MB of available space.

Defining data sources for the demo applications

Before the demo applications can be executed, you must create the data source names (DSNs) that the demo applications rely on.

On Windows, the TimesTen installation program automatically creates the appropriate data source names as System DSNs. Their configuration can be viewed and modified via the ODBC program on the Control Panel.

A sample file containing definitions for the DSNs required by the TimesTen demo applications is provided in /var/TimesTen/sys.odbc.ini, if your product was installed as root. If installed by a non-root user the file is located in *install_dir*/info/sys.odbc.ini.

Building the demo applications

Source code and makefiles are provided for all the demo applications. See the README file in *install_dir*/demo for more details about the demo directory. The procedure for building the demo applications depends on the operating system, as follows:

Windows	C:\ cd *install_dir*\demo ***C:\ nmake -f Makefile***
Windows Client/Server	C:\ cd *install_dir*\demo C:\ nmake -f Makefilecs
UNIX	% cd *install_dir*/demo % make -f Makefile
UNIX Client/Server	% cd *install_dir*/demo % make -f MakefileCS

Problems running the C demo programs

To avoid problems with the demo programs, check the environment variables and installation as discussed in the demo README files.

Problems running the C demo programs on UNIX

On UNIX, when running the demo programs, check the following:

- Are one or more TimesTen drivers installed? Check the `lib/` subdirectory of the installation directory for libraries beginning with `libtten`.

  The default installation directory is:

  – `/opt/TimesTen/`*`TTinstance`*`/` on Solaris, HP-UX and Linux.

  – `/usr/lpp/TimesTen/`*`TTinstance`*`/` on AIX.
- Is the TimesTen daemon (`timestend`) running? See "Starting and stopping the daemon on UNIX" in the *Oracle TimesTen In-Memory Database API and SQL Reference Guide*.

Problems running the C demo programs on Windows

On Windows, when running the demo programs, check the following:

- Are the correct TimesTen drivers installed? Double click on **ODBC** in the Control Panel, and check the list of installed ODBC drivers.
- Are the DSNs installed correctly? Check the System DSNs in your ODBC Data Source window. There should be several DSNs set up to use TimesTen.
- Do you have write permission on the directory where the data store resides?
- Is the TimesTen service running? To start the service, double-click **Control Panel > Administrative Tools > Services**, choose the **TimesTen Data Manager** service, and click **Start**.
- If you are using the debugging version of the driver, have you installed Microsoft Visual C++ 5.0 or 6.0?

Building and running the JDBC demo applications

Source code for a demo application is provided in the *`install_dir`*`/demo` directory as part of the TimesTen Data Manager distribution. Information about these demos is included in the `README` file included in the `demo` directory.

To run the demos:

- On Solaris, add *`install_dir`*`/lib` to the `LD_LIBRARY_PATH` environment variable.
- On Windows and UNIX, add *`install_dir`*`/demo` and *`install_dir`*`/lib/classes`*`jdk_ver`*`.jar` to the `CLASSPATH` environment variable.

Note: *jdk_ver* is two digits that represent the JDK version, for example 14 for JDK 1.4. The CLASSPATH environment variable must be set to the full pathname of the classes. You can use the ttdemoenv.sh or ttdemoenv.csh in the demo directory to set these environment variables.

If using sh, ksh, bash, zsh or a similar shell, type:

```
% . install_dir/demo/ttdemoenv.sh
```

If using csh, tcsh or similar shell, type:

```
% source install_dir/demo/ttdemoenv.csh
```

- Create the data source name (DSN) that the application relies on. For details, see "Defining data sources for the demo applications" on page 83. You can use one of the demo data sources already provided by TimesTen. See the README file in the *install_dir*/demo directory to find instructions on how to run the demo.

Viewing the online documentation

Online copies of TimesTen documentation are installed along with the TimesTen product unless you choose not to install the documentation. Documentation is provided in PDF format and can be viewed with the Adobe Acrobat Reader. If you do not currently have the Adobe Acrobat Reader installed, it is available from the Adobe Systems web page, http://www.adobe.com.

Note: The online documentation represents the most current release of the documentation. Changes to documentation for maintenance releases are only reflected online.

Online documentation is installed in the *install_dir*/doc directory. All documentation files are accessible from the Start.pdf file. Links to all the resource guides are located in this file.

Installation problems

To avoid problems during installation, make sure you have met all prerequisites. Using information in the installation guide and the release notes, check that:

- You are running a supported version of the OS.
- You have sufficient disk space.
- On UNIX, you are installing as root or the TimesTen instance administrator.

- For Windows, you are installing as user `Administrator` or as a member of the local `Administrators` group.
- For Windows, TCP/IP is installed.
- You have installed all required operating system patches.
- You have made all required kernel configuration changes.

3

Data Store Upgrades

Introduction

When a TimesTen data store is loaded into shared memory, many of its attributes are fixed, including size, logging options, sharing options, TimesTen software release number, and the location of its checkpoint and log files on disk. This chapter describes the steps required to upgrade TimesTen data stores.

Release types

A major release is one in which one (or both) of the first two numbers in the release number has changed from the previous release. For example, TimesTen 4.5 and TimesTen 6.0 are considered major releases.

A patch release is one in which the first two numbers in the release number have not changed from the previous release; only the third number has changed. TimesTen data stores are binary compatible between TimesTen patch releases. TimesTen data stores are not compatible between major releases.

Upgrade modes

TimesTen allows you to perform these kinds of upgrades:

In-place upgrades

In-place upgrades are available for moving to a new patch release of TimesTen. As long as your TimesTen data stores do not reside in the TimesTen installation directory, you can uninstall an old release of TimesTen, install a new patch release of TimesTen and connect to existing data stores with the new release. No separate action is required for existing stores.

In-place upgrades require all applications to disconnect from the data store during the upgrade procedure. This kind of upgrade allows you to

preserve the existing data store without using TimesTen's backup and migration utilities.

Offline upgrades

During the time required to perform an offline upgrade, the data store is not available to applications. Offline upgrades usually require enough disk space for an extra copy of the upgraded data store.

Offline upgrades are used to:

- move to a new major or patch release of TimesTen.
- move to a different directory or machine.
- reduce data store size.
- move between 32-bit and 64-bit data stores.

You should perform offline upgrades during a time when applications do not need continuous access to the data store. For example, if there is a maintenance window during weekends, schedule the upgrade during that time.

Offline upgrades require all applications to disconnect from the data store during the upgrade procedure. The data store must also be unloaded from shared memory. Offline upgrades require you to use TimesTen's **ttMigrate** or **ttBackup** utilities. (See "ttMigrate" on page 140 and "ttBackup" on page 91 in the *TimesTen Reference Guide*.)

Online upgrades

If upgrading to a new major release of TimesTen, you can perform an online upgrade through the use of replication in order to retain continuous availability to the data store.

Online upgrades use TimesTen replication to allow applications to remain connected to the data store during the upgrade procedure.

Updates made to the data store during the upgrade are transmitted to the upgraded data store at the end of the upgrade process. Online upgrades require more resources than offline upgrades. Performing an online upgrade requires that the data store be replicated to another data store. Therefore, twice the memory and disk space are required for an online upgrade.

Online upgrades are useful for applications where continuous availability of the data store is critical. Online upgrades might still be scheduled for a time of reduced activity, but the data store and applications are available and running at all times.

The online upgrade process only supports updates to user tables during the upgrade. Data definition changes such as CREATE TABLE or CREATE INDEX are not replicated. In addition, all tables to be replicated must have a PRIMARY KEY or a unique index on non-nullable columns.

Note: Replication is not supported between 32-bit and 64-bit data stores. Also, tables migrated using the ttMigrate `-inline` command should not be replicated with tables where the option is not supported, as inline columns cannot be replicated with not inline columns.

Performing an in-place data store upgrade

To upgrade an existing data store without exporting the data store to an external format, you can perform an in-place upgrade. This requires that all applications disconnect from the data store and that the data store be unloaded from shared memory.

Unloading a data store

TimesTen data stores remain loaded into shared memory as long as any applications are connected to them. Data stores may be kept in shared memory even when no applications are connected if the RAM policy of the data store has been modified using the **ttAdmin** utility. (See "ttAdmin" on page 85 in the *TimesTen Reference Guide*.) When the replication agent has been started using the `-repStart` option of **ttAdmin**, data stores are kept in memory. In order to unload a data store:

1. Disconnect all applications from the data store.
2. In this example, the name Original DSN refers to the DSN from the previous release. Upgrade DSN is the DSN in the new release. If replication has been started, pause replication on the Original and Upgrade DSNs and then stop replication on the data store that you are unloading from memory using the commands:

```
%ttRepAdmin -dsn Original receiver -name Upgrade
      -state pause

%ttRepAdmin -dsn Upgrade receiver -name Original
      -state pause

%ttAdmin -repStop Upgrade
```

3. Verify that the RAM policy allows the data store to be unloaded. If the RAM policy is set to manual, unload the data store using the command:

```
% ttAdmin -ramUnload DSN
```

If the RAM policy is set to always or inUse, change it to manual. If the RAM policy is inUse and a grace period is set, set the grace period to zero or wait for the grace period to elapse.

4. Use the **ttStatus** utility to verify that the data store has been unloaded from memory. (See "ttStatus" on page 172 in the *TimesTen Reference Guide*.)

Moving to a new patch release of TimesTen

All applications concurrently connected to a data store must be directly linked to a TimesTen ODBC driver of the same major release. TimesTen data stores from different patch releases are structurally equivalent or identical. For example, when upgrading from release 6.0.10 to 6.0.20, you do not need to migrate your existing data stores. However, during the installation of a new major or minor release, you should disconnect your application and stop the TimesTen daemon. If you have not explicitly performed these steps, the prior release's TimesTen daemon process is stopped, effectively disconnecting all applications from the data store. While preparing for the upgrade, make sure that all data stores are unloaded from memory before upgrading TimesTen.

For a description of the procedures for unloading a data store from memory, see "Unloading a data store" on page 89.

Performing an offline upgrade

You can do an offline upgrade by exporting the data store into an external file using either the **ttMigrate** or **ttBackup** utility, then restoring the data store with the desired changes. These update procedures require that all applications disconnect from the data store and that the data store be unloaded from shared memory. For applications that require continuous availability, see "Performing an online upgrade" on page 96.

Note: If the data store to be upgraded is replicated, you must use **ttMigrate** to move the data store between versions. In addition if you rename table owners using the **ttMigrate** `-r -rename` option, you need to rename the table owners at each node in the replication scheme individually.

When exporting a TimesTen data store, you can use either the **ttMigrate** or **ttBackup** utility. The **ttMigrate** utility exports the data store in a

release-neutral format that is more flexible, while the **ttBackup** utility exports an image copy of the data store and is faster.

The general steps in an offline upgrade include:

- Disconnect all applications from the data store and unload the data store from memory. (See "Unloading a data store" on page 89.)
- Use either **ttMigrate** with the `-noRepUpgrade` option or **ttBackup** to back up the data store.
- Install the new release of TimesTen. (See Chapter 2, "TimesTen Installation.")
- Use either **ttMigrate** with the `-noRepUpgrade` option or **ttRestore** to restore the backed up data store to the new TimesTen release.
- Reconnect applications to the upgraded data store.

Note: After **ttMigrate** has been used, all autorefresh cache groups in the destination data store have AUTOREFRESH STATE set to OFF, no matter how it was set on the source data store. Reset AUTOREFRESH STATE to ON by using the ALTER CACHE GROUP statement.

Moving to a different directory

The TimesTen daemon identifies a data store by the full path name of the data store's checkpoint files. To move a TimesTen data store to a different directory, back up the data store using the **ttBackup** utility, create a new DSN definition that specifies the new data store pathname, then restore the data store into its new location using the **ttRestore** utility. When you have verified that the data store functions properly in the new location, free up the disk space for the old backup using **ttDestroy** to remove the DSN for the old data store.

For example, to move a data store from `/old/SalesData/sales` with a data store name `SalesData` (`"DSN=SalesData"`) to `/new/SalesData/sales` with data store name `NewSalesData` (`"DSN=NewSalesData"`) using the `/tmp/dump` directory for temporary storage, use the commands:

1. `mkdir /tmp/dump`
2. `ttBackup -dir /tmp/dump -fname salesdata "DSN=SalesData"`
3. Create a DSN definition for the `NewSalesData` data store and specify the new data store path: `/new/SalesData/sales`.
4. `ttRestore -dir /tmp/dump -fname salesdata "DSN=NewSalesData"`

(Verify that `NewSalesData` is operational.)

5. `rm -r /tmp/dump`
6. `ttDestroy /old/SalesData/sales`
7. Remove the DSN definition for the `SalesData` data store.

Note: You must reconfigure replication.

Moving to a different machine

Note: Before moving a replicated data store to another machine, you should be experienced with TimesTen replication. We highly advise calling TimesTen customer support for help with this procedure.

You can also use the **ttBackup** and **ttRestore** utilities to move a data store between two machines that have the same CPU architecture and are running the same operating system.

To copy a data store from one system to another with the same CPU architecture and operating system:

1. Back up the data store on the *original* system using **ttBackup**.
2. Move the backup to the new system.
3. Reconfigure the replication scheme on any replicated data stores to identify the new host machine. (See *TimesTen to TimesTen Replication Guide* for details on how to specify a host in a replication scheme.)
4. Restore the backup using **ttRestore**.

   For example, to move a data store from `/ds/SalesData` with data store name `Sales Data` on system alpha to `/data/SalesView` with data store name `Sales View` on system beta, use the following commands. This example uses the `-o` flag of **ttBackup** to use standard output for the backup. Using the `-o` flag, the backup is stored in a single file, which is easily copied over the network to the other system. Once you have copied the data store to the other system, you need to create a data source name for convenient access to the new data store copy.

	On system alpha	On system beta
1.	`ttBackup -o "DSN = Sales Data" > /tmp/salesbackup`	
2.	`ftp /tmp/salesbackup` to system beta as `/tmp/salesbackup` Note: Use the `ftp` command in binary mode.	
3.		`ttRestore -i "DSN = Sales View" < /tmp/salesbackup`
4.		`rm /tmp/salesbackup`
5.	`rm /tmp/salesbackup`	

Reducing data store size

Once a data store has been defined with a particular size, it cannot be loaded at a smaller size, even if tables or rows are deleted. A copy of the data store made with **ttBackup** also has the data store size embedded in it.

To reduce the allocated size of a data store, save a copy with the **ttMigrate** utility with the `-noRepUpgrade` option, then recreate the data store with a smaller size and restore the data.

Here are the steps that are needed:

1. Backup the old data store with **ttMigrate** `-noRepUpgrade`.
2. Create a new data source name definition for the new copy of the data store with a smaller size specification.
3. Restore the backup with **ttMigrate** `-noRepUpgrade`.

Note: Step 2 requires a new copy of the data store. To recreate the data store in place of the original, larger data store, destroy the original data store first using the **ttDestroy** utility.

Here are the steps to reduce a data store's allocated size from 400 megabytes to 100 megabytes. The data store is in `/ds/SalesData` and has a data source name called `SalesData` already created for it.

1. `ttMigrate -c DSN=SalesData -noRepUpgrade /tmp/salesbackup`
2. `ttDestroy /ds/SalesData`
3. Update the `SalesData` DSN to have a size of 100 megabytes. See "Changing data store size" in the *Oracle TimesTen In-Memory Database Operations Guide*.
4. `ttMigrate -r "DSN=SalesData;AutoCreate=1" -noRepUpgrade /tmp/salesbackup`

Moving between 32-bit and 64-bit data stores

Note: TimesTen does not support replication between 32-bit and 64-bit data stores.

The internal format of a 32-bit TimesTen data store differs from that of a 64-bit data store. To convert a 32-bit data store to a 64-bit data store:

1. Export the 32-bit data store using the TimesTen 32-bit **ttMigrate** utility with the `-noRepUpgrade` option.
2. Create a data source name for the 64-bit data store. See "Creating, connecting to and disconnecting from a data store" in the *TimesTen Developer's Guide*.
3. Import the file created in Step 1 into the 64-bit data store DSN using the 64-bit **ttMigrate** utility with the `-noRepUpgrade` option.

   For example, suppose that the 32-bit data store DSN is SalesData32 while the 64-bit data store DSN is SalesData64.
   If TimesTen is installed in `/opt/TimesTen/giraffe`, the required steps are:

1. `/opt/TimesTen/giraffe/bin/ttMigrate -c DSN=SalesData32 -noRepUpgrade /tmp/salesbackup`
2. `/opt/TimesTen/giraffe/bin/ttMigrate -r "DSN=SalesData64;AutoCreate=1" -noRepUpgrade /tmp/salesbackup`

Moving to a different major release of TimesTen

You can have multiple TimesTen major releases installed on a system at the same time. However, TimesTen data stores created by one major release cannot be accessed directly by a different major release. To migrate data between TimesTen major releases, for example from TimesTen 5.1 to 6.0, you need to export the data using the **ttMigrate**

utility from the old release and import it using the **ttMigrate** utility from the new release.

To migrate two data stores (A and B) that replicate to each other from TimesTen release `a.b` (the old release) to TimesTen release `c.d` (the new release), first upgrade data store A on release `a.b` to release `c.d`. Data store A is on node 1, and data store B is on node 2. Perform the following steps:

1. Ensure that the replication daemon that is running on both data stores is using static port specification. This is necessary because the intermediate result of these steps is that replication occurs between a version `a.b` data store and a version `c.d` data store. Replicating across releases requires static port specification for the replication daemon. See "Dynamic vs. static port assignments" in *TimesTen to TimesTen Replication Guide*.
2. On node 1, use the `a.b` release **ttAdmin** utility to stop the replication daemon on data store A:

   ```
   ttAdmin -repStop DSN
   ```

3. On node 1, use the `a.b` release **ttMigrate** utility with the `-c` option to back up data store A to a binary file:

   ```
   ttMigrate -c DSN=DSN file_name
   ```

4. On node 1, use the `a.b` release **ttDestroy** utility to destroy data store A:

   ```
   ttDestroy data_store_path
   ```

5. On node 1, use the `c.d` release **ttMigrate** utility with the `-r` option to restore data store A from the binary file. Restoring the data store upgrades it from release `a.b` to release `c.d`. The result of this step is the creation of data A on release `c.d`.

   ```
   ttMigrate -r -C 20 DSN=DSN file_name
   ```

6. On node 1, use the `c.d` release **ttAdmin** utility to start the replication daemon:

   ```
   ttAdmin -repStart DSN
   ```

   Replication is now occurring between data store A on release `c.d` and data store B on release `a.b`.

   Now upgrade data store B from release `a.b` to release `c.d`. Perform the following steps:

1. On the old release, use the **ttAdmin** utility to stop the replication daemon:

```
ttAdmin -repStop DSN
```

2. On node 2, use the *a.b* release **ttDestroy** utility to destroy data store B:

   ```
   ttDestroy data_store_path
   ```

3. On node 2, use the *c.d* release **ttRepAdmin** utility with the `-duplicate` option to back up data store A on node 1 and restore it to node 2 as data store B. In this example, *source_data_store_name* refers to data store A, and *source_system_host_name* refers to node 1.

   ```
   ttRepAdmin -duplicate -from source_data_store_name -host
   source_system_host_name
   ```

4. On node 2, use the *c.d* release **ttAdmin** utility to start the replication daemon:

   ```
   ttAdmin -repStart DSN
   ```

   The upgraded data stores are now replicating to each other.

Performing an online upgrade

In "Performing an offline upgrade" on page 90 we showed how to perform various maintenance operations on TimesTen data stores that require that all applications be stopped. This section describes how to use the TimesTen replication feature to perform online upgrades for applications that require continuous data availability. You can do an online upgrade when moving between major TimesTen releases. If moving to a patch release, you must perform an in-place or offline upgrade.

Normally, applications that require high availability to their data use TimesTen replication to keep at least one extra copy of their data stores current. The online upgrade procedures described in this section assume that you have a bi-directional replication scheme configured and running for two data stores, as described in the *TimesTen to TimesTen Replication Guide*.

Note: Replication functions across releases only if the data store of the more recent version of TimesTen was upgraded from a data store of the older version of TimesTen. A data store created in the more recent version of TimesTen is not guaranteed to replicate correctly with the older version. For example, replication between a data store created in a 5.1 version of TimesTen and a data store created in a 6.0 version of TimesTen is not supported. However, if one data store was created in a

5.1 version, and the peer data store was created in a 5.1 version and then upgraded to a 6.0 version, replication between them is supported.

Overview

An upgrade from a TimesTen release X data store to a TimesTen release Y data store is performed by making a backup of the data store with **ttMigrate** release X, then loading the backup into a release Y data store using the **ttMigrate** release Y.

Note: The **ttMigrate** `-r -rename` option, used to rename the owner of tables, cannot be used with online upgrades.

The general steps in an online upgrade include:

- Remove all connected applications from the data store to be upgraded.
- Shut down replication on the upgrade system.
- Back up the data store residing on the upgrade system using **ttMigrate** -c of release X.
- Install Release Y of TimesTen on the upgrade system.
- Migrate the replicated data store to the upgrade system using **ttMigrate** -r of release Y.
- Connect applications to the upgrade data store.
- **ttRepAdmin** -upgrade.
- **ttRepAdmin** -check.
- Restart replication.

Note: After **ttMigrate** has been used, all autorefresh cache groups in the destination data store have AUTOREFRESH STATE set to OFF, no matter how it was set on the source data store. Reset AUTOREFRESH STATE to ON by using the ALTER CACHE GROUP statement.

To maintain continuous availability, applications will continue to run while the copy and upgrade are performed. TimesTen replication captures and retains updates made during the copy and upgrade period and then applies them to the new data store. When the replication updates have been completely applied, the applications can stop accessing the original data store and start accessing the new data store.

This time line illustrates the steps for performing an online upgrade while replication is running:

Step	Original System	Upgrade System
1.	Production applications running.	Production applications running.
2.		Move applications to *Original* system.
3.	Pause replication to *Upgrade* system.	
4.		Wait for updates to propagate to *Original* system.
5.		Stop replication on *Upgrade* system.
6.		Take backup of *Upgrade* system with **ttMigrate -c.**
7.		Shut down *Upgrade* system.
8.		Install new release of TimesTen.
9.		Create data sources with the new TimesTen release, using the **ODBC Data Source Administrator** on Windows or the `.odbc.ini` file on UNIX.
10.		Restore data store from backup with **ttMigrate -r**.
11.		Apply automatic update of replication definitions. **ttRepAdmin** -upgrade
12.		Reset replication to *Original* system.
13.		Place replication to *Original* system in the `start` state.
14.		Start replication.
15.	If the *Original* data store was created with TimesTen release 4.1 or higher, reset replication to *Upgrade* system.	

16.	Start replication to *Upgrade* system.	
17.	Wait for updates to propagate to *Upgrade* system.	
18.		Start production applications.

After the above procedures have been carried out on the *Upgrade* system, the *Original* system can be upgraded using the same steps.

Limitations

Online upgrades can be performed only on data stores for which all the user tables meet the replication requirements. All the user tables must contain either a PRIMARY KEY declaration or have a unique index declared over non-nullable columns.

Requirements

To perform online upgrades, you must have the main memory and disk space to support the data store and a copy of the data store being upgraded. Both the original data store and its copy will be active for the duration of the upgrade. To protect the performance of your production applications, you may wish to perform the upgrade on a separate system.

To perform online upgrades with replication, replication must use a static port. See the *TimesTen to TimesTen Replication Guide*,

Additional disk space must be allocated to hold a backup copy of the data store made by the **ttMigrate** utility. The size of the backup copy is typically about the same as the in-use size of the data store. This size may be determined by querying the MONITOR table, using **ttIsql**:

```
C:> select perm_in_use_size from SYS.MONITOR;
```

Online upgrade example

This section describes how to do an online upgrade of two bi-directionally replicated TimesTen data stores.

We'll refer to the two TimesTen systems being upgraded as the *Original* system and the *Upgrade* system. The *Original* system remains on its current release of TimesTen for the duration of this procedure. The *Upgrade* system is converted to the new release during this procedure. After this procedure has been completed, the same steps can be applied

to the *Original* system to convert it as well. However, you may prefer to delay conversion of the *Original* system to test the upgraded release.

Follow the steps listed here in the order they are presented. The online upgrade procedures are:

Step	Original System	Upgrade System
1.	Use the **ttRepAdmin** utility to pause replication from the *Original* data store to the *Upgrade* data store: `ttRepAdmin -dsn Original -receiver -name Upgrade -state pause` This command temporarily stops the replication of updates from the *Original* system to the *Upgrade* system, but it saves any updates made on the *Original* system to the data store log files. The updates made to the *Original* system during the upgrade procedure are applied later. For details on resetting replication state, see the section "'Setting the replication state of subscribers'" in Chapter 4 of the *TimesTen to TimesTen Replication Guide*.	Stop all production applications. Any workload being run on the *Upgrade* system must start running on the *Original* system.
2.		Wait for all replication updates to be sent to the *Original* system. You can verify that all updates have been sent by applying a recognizable update to a table reserved for that purpose on the *Upgrade* data store. When the update is replicated to the *Original* data store, you know that updates have been sent.

Step	Original System	Upgrade System
3.	On the *Original* system, stop the replication agent with the **ttAdmin** utility: `ttAdmin -repStop Original` From this point, no updates will be sent to the *Upgrade* data store. For details on starting and stopping replication agents, see the section "Starting and stopping the replication agents" in Chapter 3 of the *TimesTen to TimesTen Replication Guide*.	On the *Upgrade* system, stop the replication agent with the **ttAdmin** utility: `ttAdmin -repStop Upgrade` From this point, no updates will be sent to the *Original* data store.

4.	Use **ttIsql** to alter the replication scheme so that the data stores can talk across releases. You should set the static port numbers for each data store	Use **ttRepAdmin** to stop replication from the *Upgrade* data store to the *Original* data store:
		`ttRepAdmin -dsn Upgrade`
		`  -receiver`
		`  -name Original`
		`  -state stop`
		This step stops the *Upgrade* data store from accumulating updates to send to *Original* and resets some of the replication bookmarks.

Step	Original System	Upgrade System
	If setting the port numbers, specify ALTER REPLICATION with SET PORT parameters: `ttIsql -connStr dsn=Original -e "alter replication repl.scheme alter store Upgrade on upgradehost set port 22011; alter replication repl.scheme alter store Original on originalhost set port 22012;quit;"` **Note:** If you choose to set static ports, this step can be done before the *Upgrade* data store is upgraded, but the above **ttIsql** command must be run on both data stores.	
5.		On the *Upgrade* system, use the **ttMigrate** utility to back up the data store. Depending on the size of the data store, this step may be time consuming. If sufficient disk space is free on the `/backup` file system, the following **ttMigrate** command can be used: `ttMigrate -c DSN=Upgrade /backup/upgrade.dump`

Step	Original System	Upgrade System

6.	Restart the replication agent on the *Original* data store:
	`ttAdmin -wait -repStart Original`
		If the migration is successful, destroy the *Upgrade* data store.
		To destroy a permanent data store (**Temporary**=0), use **ttDestroy**:
		`ttdestroy Upgrade`
		To destroy a temporary data store (**Temporary**=1), use **ttAdmin**:
		`ttAdmin -ramUnload Upgrade`
		Note: You can keep the old *Upgrade* data store until the migration into the new TimesTen version is successful. However, the replication agent must remain stopped and *Original*-to-*Upgrade* replication must be set to the Pause state on the *Original* data store.

7.	Resume replication from the *Original* to the *Upgrade* data store: `ttRepAdmin -dsn Original -receiver -name Upgrade -start start` At this point, the *Original* and *Upgrade* systems are on different releases of TimesTen. TimesTen supports replication between different major releases of TimesTen to help support online upgrades.	Install the upgrade release of TimesTen

Step	Original System	Upgrade System

8.		Use the **ttMigrate** command to load the backup created in Step 5 into a new version of the *Upgrade* data store.
		If *Upgrade* is a temporary data store (**Temporary**=1), first use **ttAdmin** `-ramload`:
		`ttAdmin -ramLoad Upgrade`
		`ttMigrate -r "DSN=Upgrade; AutoCreate=0" /backup/upgrade.dump`
		Note: In this step, you must use the **ttMigrate** utility supplied with the new release of TimesTen to which you are upgrading.

9.		Migrate the replication definitions using the **ttRepAdmin** utility. On some release upgrades, the replication definitions may need to change. The **ttRepAdmin** `-upgrade` option automatically upgrades the replication definitions from the previous release:
		`ttRepAdmin -dsn Upgrade -self -upgrade`
		`ttRepAdmin -dsn Upgrade -self -check`
		`ttRepAdmin -dsn Upgrade -receiver -name Original -reset`
		Check that the data store identity is correct:
		`ttRepAdmin -dsn Upgrade -self -list`

Step	Original System	Upgrade System
10.		Use **ttIsql** to alter the replication scheme so that the data stores can talk across releases. You should set the static port numbers for each data store If setting the port numbers, specify ALTER REPLICATION with SET PORT parameters: `ttIsql -connStr dsn=Original -e "alter replication repl.scheme alter store Upgrade on upgradehost set port 22011; alter replication repl.scheme alter store Original on originalhost set port 22012;quit;"` **Note:** If you choose to set static ports, this step can be done before the *Upgrade* data store is upgraded, but the above **ttIsql** command must be run on both data stores.

11.		Use **ttRepAdmin** to set replication on the *Original* data store to the `stop` state and then to the `start` state:
		
		`ttRepAdmin -dsn Upgrade -receiver -name Original -state stop`
		
		`sleep 10`
		
		`ttRepAdmin -dsn Upgrade -receiver -name Original -state start`
		
		`sleep 10`
		
		Note: The `sleep` is to ensure that each state takes effect, as the state change can take up to 10 seconds on a given machine depending on resources and operating system

Step	Original System	Upgrade System
12.		Use **ttAdmin** to start the replication agent on the new *Upgrade* data store and begin accumulating updates to send to the *Original* system: `ttAdmin -repStart Upgrade`
13.	If the applications are still running on the *Original* data store, let them continue until the *Upgrade* system has been successfully migrated and you have verified that the updates are being replicated correctly from the *Original* system to the *Upgrade* system.	Verify that the *Upgrade* system is receiving updates from the *Original* system, You can verify that updates have been sent by applying a recognizable update to a table reserved for that purpose on the Upgrade data store. When the update is replicated to the *Original* data store, you know that updates have been sent.
14.		Once you are sure that updates are being replicated correctly, you can move all the applications back to the *Upgrade* system. After verifying that the last of the updates from the *Original* system have been replicated to the *Upgrade* system, then the *Original* system is ready for upgrade.

15.	Pause replication from the *Original* data store to the *Upgrade* data store:
	`ttRepAdmin -dsn Original -receiver -name Upgrade -state pause`
	`sleep 10`
		Pause replication from the *Upgrade* data store to the *Original* data store:
		`ttRepAdmin -dsn Upgrade -receiver -name Original -state pause`
		`sleep 10`

16.	Use **ttAdmin** to stop the replication agent on the *Original* data store: `ttAdmin -repStop Original`	Use **ttAdmin** to stop the replication agent on the *Upgrade* data store: `ttAdmin -wait -repStop Upgrade`

Step	Original System	Upgrade System
17.	On the *Original* system, use the **ttMigrate** utility to back up the data store. Depending on the size of the data store, this step may be time consuming. If sufficient disk space is free on the `/backup` file system, the following **ttMigrate** command can be used: `ttMigrate -c DSN=Original /backup/original.dump`	

18.	If the migration is successful, destroy the *Original* data store.
	To destroy a permanent data store (**Temporary**=0), use **ttDestroy**:
	`ttdestroy Original`
	To destroy a temporary data store (**Temporary**=1), use **ttAdmin**:
	`ttAdmin -ramUnload Original`
	Note: You can keep the old *Original* data store until the migration into the new TimesTen version is successful. However, the replication agent must remain stopped and *Upgrade*-to-*Original* replication must be set to the `Pause` state on the *Upgrade* data store.	

Step	Original System	Upgrade System

19.	Use the **ttMigrate** command to load the backup created in Step 17 into a new version of the *Original* data store.
	If *Original* is a temporary data store (**Temporary**=1), first use **ttAdmin** `-ramload`:
	`ttAdmin -ramLoad Original`
	`ttMigrate -r "DSN=Original; AutoCreate=0" /backup/original.dump`
	Note: In this step, you must use the **ttMigrate** utility supplied with the new release of TimesTen to which you are upgrading.	

20.	Use **ttRepAdmin** to migrate the replication definitions. On some release upgrades, the replication definitions may need to change. The **ttRepAdmin** `-upgrade` option automatically upgrades the replication definitions from the previous release:
	`ttRepAdmin -dsn Original -self -upgrade`
	`ttRepAdmin -dsn Original -self -check`
	`ttRepAdmin -dsn Original -receiver -name Upgrade -reset`
	Check that the data store identity is correct:
	`ttRepAdmin -dsn Original -self -list`
		Use **ttRepAdmin** to reset the bookmark in the *Upgrade* data store log to the latest transaction:
		`ttRepAdmin -dsn Upgrade -receiver -name Original -reset`
		Use **ttAdmin** to restart the replication agent for the *Upgrade* data store:
		`ttAdmin -wait -repStart Upgrade`

Step	Original System	Upgrade System
21.	Start the replication agent for the *Original* data store: `ttAdmin -wait -repStart Original`	

22.		Restart replicating updates from *Upgrade* to *Original*:
		`ttRepAdmin -dsn Upgrade -receiver -name Original -state start`
		`sleep 10`

23.	Verify updates are replicated from *Upgrade* to *Original*.	Verify that all previous updates in the log at the time replication to *Original* was paused in Step 15 and all new updates logged since then have been replicated to the newly upgraded *Original* system.

Performing an online Client/Server upgrade on UNIX

Overview

A user application linked with a 5.0 or later release of TimesTen Client ODBC driver can connect to a Server DSN of a newer major release of TimesTen. For example, a 5.0.20 client can connect to a 5.1 Server DSN.

Note: Applications linked to newer TimesTen clients cannot connect to a Server DSN of any older releases. For example, a 5.1 client cannot connect to a 5.0.20 Server DSN.

Limitations

If a 5.0 client is connecting to a 5.1 or later Server DSN, the source of the log messages in the syslog (or in the specified file) pertaining to that connection will still be 5.0 because the 5.0 Server binary performs the logging.

Online Client/Server upgrade example

To connect an older client application to a newer TimesTen Server DSN, use the procedures described in the following example:

Let us assume that the hostname for the Server machine is `my_server`.

TimesTen 5.0.20 is installed in *`install_dir`*`_5020` at `my_ server.` The port number for the TimesTen daemon is **15000**. The port number for TimesTen Server is **15002**.

TimesTen 5.1 is installed in *`install_dir`*`_51` at `my_server`. The port number for TimesTen daemon is **15100**. The port number for TimesTen Server is **15102**.

The content of the `sys.odbc.ini` file at `my_server` is:

```
[my_server_dsn50]
Driver=<install_dir_5020>/lib/libtten.so
DataStore=/tmp/ds_50

[my_server_dsn51]
Driver=<install_dir_51>/lib/libtten.so
DataStore=/tmp/ds_51
```

The content of the TTCONNECTINI file at the TimesTen 5.0.x client machine is:

```
[my_server_50]
Network_Address=my_server
TCP_PORT=15002
```

Because the TCP_PORT is set to the TimesTen 5.0 Server port number, the 5.0 dynamically loads the correct driver for the Server DSN.

The content of `.odbc.ini` file at the TimesTen 5.0.20 client machine is:

```
[client_dsn_50]
Description=Access to my_server_dsn50 on host "server"
TTC_SERVER= my_server_50
TTC_SERVER_DSN= my_server_dsn50

[client_dsn_51]
Description= Access to my_server_dsn51 on host "server"
TTC_SERVER= my_server_50
TTC_SERVER_DSN= my_server_dsn51
DRIVER=install_dir_51/lib/libtten.so
```

Note: The full path to the 5.1 driver must be specified. Otherwise the server loads the default driver.

Step	Client Machine	Server Machine
1.		Start the TimesTen daemon for both 5.0.20 and 5.1 installations at `my_server` using the regular start script.
2.	At the client machine, using the ttIsqlCS utility, to connect to the 5.0.20 Server DSN, `my_server_dsn50`, use: `ttIsqlCS -connStr "dsn=client_dsn_50"`	
3.	Using the ttIsqlCS utility, to connect to the 5.1 Server DSN, `server_dsn51`, use: ttIsqlCS –connStr "dsn=client_dsn_51	

Record of Upgrades

A new set of TTREP_SCHEMA_VERSION tables is created each time after running the **ttMigrate** `-r` command to provide a history of the data store through all of the releases it has been migrated.

These tables do not take up much space and are helpful when debugging upgrade problems. However, you can drop these tables if you do not experience problems after running **ttMigrate**.

Example 3.1 For example, after two migrates, the tables might look like:

```
TTREP_SCHEMA_VERSION_004.REPELEMENTS
TTREP_SCHEMA_VERSION_004.REPLICATIONS
TTREP_SCHEMA_VERSION_004.REPPEERS
TTREP_SCHEMA_VERSION_004.REPSTORES
TTREP_SCHEMA_VERSION_004.REPSUBSCRIPTIONS
TTREP_SCHEMA_VERSION_004.REPTABLES
TTREP_SCHEMA_VERSION_004.TTSTORES
TTREP_SCHEMA_VERSION_005.REPELEMENTS
TTREP_SCHEMA_VERSION_005.REPLICATIONS
TTREP_SCHEMA_VERSION_005.REPPEERS
TTREP_SCHEMA_VERSION_005.REPSTORES
TTREP_SCHEMA_VERSION_005.REPSUBSCRIPTIONS
```

```
TTREP_SCHEMA_VERSION_005.REPTABLES
TTREP_SCHEMA_VERSION_005.TTSTORES
```

Index

F

G

H

I

J

K

L

M

N

O

P

R

S

T

U

V

W