

*Oracle TimesTen
In-Memory Database
Installation Guide*

Release 7.0

B31679-03



Copyright ©1996, 2007, Oracle. All rights reserved.

ALL SOFTWARE AND DOCUMENTATION (WHETHER IN HARD COPY OR ELECTRONIC FORM) ENCLOSED AND ON THE COMPACT DISC(S) ARE SUBJECT TO THE LICENSE AGREEMENT.

The documentation stored on the compact disc(s) may be printed by licensee for licensee's internal use only. Except for the foregoing, no part of this documentation (whether in hard copy or electronic form) may be reproduced or transmitted in any form by any means, electronic or mechanical, including photocopying, recording, or any information storage and retrieval system, without the prior written permission of TimesTen Inc.

Oracle, JD Edwards, PeopleSoft, Retek, TimesTen, the TimesTen icon, MicroLogging and Direct Data Access are trademarks or registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

The Programs (which include both the software and documentation) contain proprietary information; they are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright, patent, and other intellectual and industrial property laws. Reverse engineering, disassembly, or decompilation of the Programs, except to the extent required to obtain interoperability with other independently created software or as specified by law, is prohibited.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. This document is not warranted to be error-free. Except as may be expressly permitted in your license agreement for these Programs, no part of these Programs may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose.

September 2007

Printed in the United States of America

Contents

About this Guide

TimesTen documentation	1
Background reading	2
Conventions used in this guide	3
Technical Support	5

1 Access Control

Introduction.	7
Access Control	7
Limitations of Access Control and non-root installs	7
General	7
Cache Connect	8
Replication	8
Client/Server	8
Instance access	8
Instance startup/shutdown.	8
Instance data store.	9
TimesTen users	9
TimesTen instance administrator	9
TimesTen instance users	10
Before installation	11
TimesTen administrators group.	11
Instance registry directory.	11
Installation directories, files and the daemon port	12
Installation directories	12
Instance home directory	12
Daemon home directory	12
Password file.	12
Daemon port	13
Authenticating users and privileges	13
Privileges	13
GroupRestrict	13
Maintaining users and privileges	14
Listing of defined users and privileges	14
Enabling Access Control after installation on UNIX.	14

2 TimesTen Installation

Platforms and configurations	19
--	----

Platform support	19
JDK support	20
Client/Server configurations	21
Cache Connect to Oracle	22
Replication configurations	23
Installation instances	23
Instance names	24
Instance port numbers	24
Choosing the appropriate TimesTen components	24
Components available on Windows	24
Components available on UNIX	25
Installation prerequisites	25
General UNIX requirements	25
AIX	27
HP-UX	27
Linux	28
Solaris	31
Tru64 UNIX	34
Default installation directories	36
Cache Connect	36
Operating system security considerations	37
Prerequisites for non-root installations	
on UNIX systems	38
Installation prerequisites for non-root installs	38
Create the TimesTen instance administrators group	38
Create the TimesTen registry	39
Post-installation requirements	39
Configure the syslog messages	40
Changing the daemon port number on UNIX	40
UNIX libraries	40
Installing TimesTen on Windows systems	41
Installing TimesTen	41
Installing TimesTen in silent mode	42
Verifying installation	42
Verifying TimesTen Client and Server installation	43
Working with the Data Manager Service and the Server	44
Uninstalling TimesTen	45
Installing TimesTen on Solaris systems	45
Installing TimesTen	45
Working with the daemon and Server	49

Uninstalling TimesTen	50
Installing TimesTen on HP-UX systems.	50
Installing TimesTen	50
Working with the TimesTen daemon and Server	54
Uninstalling TimesTen	55
Installing TimesTen on HP-UX Memory Windows	55
Use a separate instance for each memory window	55
Using TimesTen in a memory window	56
Address Space Considerations	56
Troubleshooting.	57
Installing TimesTen on AIX systems	58
Installing TimesTen	58
Working with the TimesTen daemon and server	62
Uninstalling TimesTen	62
Installing TimesTen on Linux systems	63
Installing TimesTen	63
Working with the TimesTen daemon and Server	66
Uninstalling TimesTen	67
Installing TimesTen on Tru64 UNIX systems	67
Installing TimesTen	67
Working with the TimesTen daemon and Server	71
Uninstalling TimesTen	71
Using the Cache Administrator	72
Informational messages on Windows systems	73
Informational messages on UNIX systems.	73
Incremental install and uninstall of Cache Connect	75
ODBC installation	75
Environment modifications.	75
PATH environment variable	77
ODBCINI environment variable	77
SYSODBCINI environment variable	78
SYSTTCONNECTINI environment variable	78
CLASSPATH environment variable	78
ORACLE_HOME environment variable	79
Shared library path environment variable.	79
Web server configuration	80
Migrating data stores to TimesTen 7.0	81
Using the ttMigrate utility.	81
Using the ttBulkCp utility.	83
Building and running the demo applications	84
Defining data sources for the demo applications	84

Building the demo applications	85
Problems running the demo programs	85
Building and running the JDBC demo applications	86
Viewing the online documentation	86
Installation problems	87

3 Data Store Upgrades

Introduction	89
Data store compatibility	89
Data type compatibility	89
Data store character set	90
Data type conversion	91
Converting data types to Oracle data types	92
Upgrading data types as TimesTen data types	92
Data store character set conversion	93
Converting from the TIMESTEN8 character set.	93
Converting from a character set other than TIMESTEN8	94
Upgrade modes	95
In-place upgrades.	95
Offline upgrades	95
Online upgrades with replication.	96
Online upgrades with Client/Server.	96
Performing an in-place data store upgrade	97
Unloading a data store	97
Moving to a new patch release of TimesTen	98
Performing an offline upgrade	99
Moving to a different directory	100
Moving to a different machine.	100
Reducing data store size	102
Moving between 32-bit and 64-bit data stores.	103
Moving to a different major release of TimesTen	103
Performing an online upgrade with replication	106
Overview	106
Limitations	109
Requirements	109
Online upgrade example	110
Performing a Client/Server online upgrade from a TimesTen version prior to 6.0 115	
Overview	115
Limitations	116
Client/Server online upgrade example.	116

Performing a Client/Server online upgrade from TimesTen version 6.0 and above	118
Client/Server online upgrade.	118
Client/Server online upgrade with continuous access to the data store	119
Record of Upgrades	119

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

TimesTen documentation is available on the product distribution media and on the Oracle Technology Network:

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

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

Book Titles	Description
<i>Oracle TimesTen In-Memory Database Installation Guide</i>	Contains information needed to install and configure TimesTen on all supported platforms.
<i>Oracle TimesTen In-Memory Database Introduction</i>	Describes all the available features in the Oracle TimesTen In-Memory Database.
<i>Oracle TimesTen In-Memory Database Operations Guide</i>	Provides information on configuring TimesTen and using the <code>ttIsql</code> utility to manage a data store. This guide also provides a basic tutorial for TimesTen.
<i>Oracle TimesTen In-Memory Database C Developer's and Reference Guide</i> and the <i>Oracle TimesTen In-Memory Database Java Developer's and Reference Guide</i>	Provide information on how to use the full set of available features in TimesTen to develop and implement applications that use TimesTen.
<i>Oracle TimesTen In-Memory Database API Reference Guide</i>	Describes all TimesTen utilities, procedures, APIs and provides a reference to other features of TimesTen.

<i>Oracle TimesTen In-Memory Database SQL Reference Guide</i>	Contains a complete reference to all TimesTen SQL statements, expressions and functions, including TimesTen SQL extensions.
<i>Oracle TimesTen In-Memory Database Error Messages and SNMP Traps</i>	Contains a complete reference to the TimesTen error messages and information on using SNMP Traps with TimesTen.
<i>Oracle TimesTen In-Memory Database TTClasses Guide</i>	Describes how to use the TTClasses C++ API to use the features available in TimesTen to develop and implement applications.
<i>TimesTen to TimesTen Replication Guide</i>	Provides information to help you understand how TimesTen Replication works and step-by-step instructions and examples that show how to perform the most commonly needed tasks. This guide is for application developers who use and administer TimesTen and for system administrators who configure and manage TimesTen Replication.
<i>TimesTen Cache Connect to Oracle Guide</i>	Describes how to use Cache Connect to cache Oracle data in TimesTen data stores. This guide is for developers who use and administer TimesTen for caching Oracle data.
<i>Oracle TimesTen In-Memory Database Troubleshooting Procedures Guide</i>	Provides information and solutions for handling problems that may arise while developing applications that work with TimesTen, or while configuring or managing TimesTen.

Background reading

For a Java reference, see:

- Horstmann, Cay and Gary Cornell. *Core Java(TM) 2, Volume I-- Fundamentals (7th Edition) (Core Java 2)*. Prentice Hall PTR; 7 edition (August 17, 2004).

A list of books about ODBC and SQL is in the Microsoft ODBC manual included in your developer's kit. 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:

- Kyle Geiger. *Inside ODBC*. Redmond, WA: Microsoft Press. 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, Second Edition*. McGraw-Hill Osborne Media. 2002.

For information about Unicode, see:

- The Unicode Consortium, *The Unicode Standard, Version 5.0*, Addison-Wesley Professional, 2006.
- 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 <code>.odbc.ini</code> or <code>ttconnect.ini</code> file.
<i>italic code font</i>	A variable in a code example that you must replace. For example: <code>Driver=install_dir/lib/libtten.sl</code> Replace <code>install_dir</code> with the path of your TimesTen installation directory.

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

If you see...	It means...
<i>fixed width italics</i>	Variable; must be replaced with an appropriate value. In some cases, such as for parameter values in built-in procedures, you may need to single quote (' ') the value.
[]	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...
<i>install_dir</i>	The path that represents the directory where the current release of TimesTen is installed.
<i>TTinstance</i>	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.
<i>bits</i> or <i>bb</i>	Two digits, either 32 or 64, that represent either the 32-bit or 64-bit operating system.
<i>release</i> or <i>rr</i>	Two digits that represent the first two digits of the current TimesTen release number, with or without a dot. For example, 70 or 7.0 represents TimesTen Release 7.0.

<i>jdk_version</i>	Two digits that represent the version number of the major JDK release. Specifically, 14 represent JDK 1.4; 5 represents JDK 5.
<i>timesten</i>	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.
<i>DSN</i>	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>

Access Control

Introduction

Access Control

With TimesTen you can optionally install a layer of internal 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 internally defined privileges. In TimesTen, user privileges are granted on a instance wide-basis. A user’s privileges apply to all data stores in a given TimesTen instance or installation.

Limitations of Access Control and non-root installs

General

You can enable Access Control when you install TimesTen. You can also choose to enable it after installation by using the **ttmodinstall** utility. See [“Enabling Access Control after installation on UNIX” on page 14](#). Access Control cannot be disabled after installation of TimesTen. You must uninstall and re-install TimesTen if you want to disable Access Control.

The instance administrator owns all files in the installation directory tree. Only the instance administrator can administer the TimesTen instance. See [“TimesTen instance administrator” on page 9](#). 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 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 13](#).

Changes are applied to a replicated subscriber data store regardless of the settings or presence of Access Control on the subscriber.

Instance user configuration commands are not replicated.

Client/Server

If a TimesTen client connects to a TimesTen server, and the server side data store has Access Control enabled, the server's **Authenticate** attribute must be enabled.

To use Access Control with Client/Server applications, when the user is identified externally, the Client and the Server processes must be on the same machine. When Access Control is enabled, remote Client/Server access is only supported with TimesTen internal users.

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

When Access Control is enable, 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.

Instance access

Instance startup/shutdown

Permission to start and stop the main TimesTen daemon is restricted to the TimesTen instance administrator.

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 following is the definition of the instance DSN for a root installation:

```
[TT_instance]
Driver=install_dir/lib/libtten.suffix
DataStore=/var/TimesTen/instance/TT_instance
```

The following is the definition of the instance DSN for a non-root installation:

```
[TT_instance]
Driver=install_dir/lib/libtten.suffix
DataStore=install_dir/info/TT_instance
```

This data store gets special treatment from the daemon, and has special access restrictions placed on it. Any user can connect to the instance data store to change their own password. However, users other than the instance administrator have only SELECT privileges on the instance data store.

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 TimesTen because only the instance administrator user can 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 other users in that instance. If the [GroupRestrict](#) attribute is set, the instance administrator user must have corresponding group membership.

Note: 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, a TimesTen instance administrator user is the OS user who installs that instance of Timesten.

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 an 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 be 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 SQL Reference Guide*. TimesTen user names (as specified in the UID DSN attribute) 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 some external mechanism. See ["CREATE USER"](#) in the *Oracle TimesTen In-Memory Database 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 allows you to encrypt a password rather than use cleartext passwords, and it also provides a way to deal with the special characters and case sensitivity used in passwords that might create difficulties if specified in clear text within the PWD DSN attribute.

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. See [“Installation prerequisites” on page 25](#). Additional steps must be performed before 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 the TimesTen instance becomes the [TimesTen instance administrator](#) for that instance.

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, but it is essential for correct installation and uninstallation of TimesTen. It is not accessible by TimesTen users including the instance administrator user.



On Unix platforms, for root user installs, the instance registry is located in the directory `/etc/TimesTen/`. Initial creation of the `/etc/TimesTen/` directory may require 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 including the instance administrator user.

Installation directories, files and the daemon port

Installation of TimesTen must be performed by the chosen instance administrator user. The instance administrator owns all files in the installation directory tree. Only the instance administrator can 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/tt70`. For non-root users, the installer suggests the home directory of the user, usually defined by the environment variable `$HOME`.

On Windows, the installer suggests the directory pattern of TimesTen, `C:\TimesTen\tt70`.

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

Daemon home directory

The “home” or current working directory of the running the main TimesTen daemon is known as the daemon home directory. This directory must be owned by the instance administrator, with `rxwxr-xr-x` permissions on UNIX systems. The daemon verifies both the permissions and ownership of this directory when it starts. This directory can be anywhere on a local drive, but cannot be on an NFS mounted file system, except on Linux x86 32-bit or 64-bit systems.

On UNIX, the installer suggests the use of *install_dir/info* if installed as non-root or `/var/TimesTen/tt70` if running as root.

On Windows, the *install_dir\srv\info* directory is used for this purpose.

Password file

If access control is selected at installation time, user and password data is stored in the file *install_dir/srv/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, an error occurs after Access Control is enabled.

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 as one way hashes, so they cannot be recovered.

Daemon port

Though the instance registry enforces port/TCP/IP uniqueness for TimesTen instances, 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](#) for ways to change the demon port number after installation.

Authenticating users and privileges

When Access Control is enabled, certain TimesTen utility APIs, XLA operations, utilities, procedures and SQL operations require user authentication. For details on each operation, see the specific chapters of the *Oracle TimesTen In-Memory Database API Reference Guide* and the *Oracle TimesTen In-Memory Database Operations Guide*.

All TimesTen utilities prompt for a password if needed. See [Chapter 3, “Utilities”](#) in the *Oracle TimesTen In-Memory Database API 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, rather than embedding a cleartext password in the script.

Privileges

For a description of the TimesTen Access Control privileges, see [“Access Control Privileges”](#) in the *Oracle TimesTen In-Memory Database SQL Reference Guide*,

GroupRestrict

The instance administrator must be included in the **GroupRestrict** groups being used.

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 5, “SQL Statements in the *Oracle TimesTen In-Memory Database SQL Reference Guide*.”](#)

Administration of users is done at the instance level by establishing a connection to any 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 **ttUserPrivileges** built-in procedure lists the privileges granted to users defined in the instance.

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.

Enabling Access Control after installation on UNIX



On UNIX, the **ttmodinstall** utility allows the instance administrator to enable Access Control if it was not enabled at install time. If you have not stopped the TimesTen daemon before using **ttmodinstall**, the utility stops the daemon before changing the port number. After the port change, the daemon is automatically restarted. If you have not stopped the entire TimesTen instance, then **ttmodinstall** will stop the instance, make the necessary changes, then restart the instance.

This is useful, if you install TimesTen and later determine that you want to enable Access Control.

The utility is run from the command line and takes the `-enableAccessControl` option. For example:

```
% ttmodinstall -enableAccessControl
```

Note: Disabling Access Control can only be done by uninstalling and re-installing the same or a different release of TimesTen.

The **ttmodinstall** utility can also modify the path supplied to the `ORACLE_HOME` environment variable that provides Cache Connect to Oracle with the knowledge of where Oracle is installed.

All other changes to the TimesTen instance can only be made by uninstalling and re-installing the same or a differently release of TimesTen. (See [“Changing the daemon port number on UNIX” on page 40](#) and [“ORACLE_HOME environment variable” on page 79](#).)

TimesTen Installation

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

- [Platforms and configurations](#)
- [Installation instances](#)
- [Choosing the appropriate TimesTen components](#)
- [Installation prerequisites](#)
- [Operating system security considerations](#)
- [Prerequisites for non-root installations 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 7.0](#)

- [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 IA-32 and EM64T and AMD64 CPUs.	Yes	Yes
Asianux 2.0 for Intel IA-32 and EM64T and AMD64 CPUs.	Yes	Yes
Solaris 10 for AMD64 CPUs.	Yes	Yes
Solaris 8, 9 and 10 for UltraSparc CPUs.	Yes	Yes
SuSE LINUX Enterprise Server 9 and 10 for Intel IA-32 and EM64T and AMD64 CPUs.	Yes	Yes
SuSE LINUX Enterprise Server 10 for Itanium2 CPUs		Yes
Red Hat Enterprise Linux 3, 4 and 5 for Intel Itanium2 CPUs.		Yes
Red Hat Enterprise Linux 3, 4 and 5 for Intel IA-32 and EM64T and AMD64 CPUs.	Yes	Yes
Oracle Enterprise Linux 4 and 5 for Intel IA-32 and EM64T and AMD64 CPUs.	Yes	Yes
MontaVista Linux Carrier Grade Edition Release 4.0 for Intel IA-32, EM64T and AMD64 CPUs.	Yes	Yes
HP-UX 11i, 11i v2 and 11iv3 for PA-RISC	Yes	Yes
HP-UX 11i v2 and 11iv3 for Itanium2.	Yes	Yes
AIX 5L 5.2 and 5.3 for POWER CPUs	Yes	Yes
Tru64 UNIX 5.1B for Alpha EV68 CPUs		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	BEA WebLogic JRockit 5.0
Microsoft Windows 2000, Windows XP and Windows Server 2003 for Intel IA-32 and EM64T and AMD64 CPUs	Yes	Yes	Yes
Asianux 2.0 for Intel IA-32 and EM64T and AMD64 CPUs.	Yes	Yes	Yes
Solaris 10 for AMD64 CPUs	Yes (32-bit and 64-bit)	Yes (32-bit and 64-bit)	Yes (32-bit and 64-bit)
Solaris 8, 9 and 10 for UltraSparc CPUs	Yes (32-bit and 64-bit)	Yes (32-bit and 64-bit)	Yes (64-bit only)
SuSE LINUX Enterprise Server 9 and 10 for Intel IA-32 and EM64T and AMD64 CPUs	Yes (32-bit and 64-bit)	Yes (32-bit and 64-bit)	Yes (32-bit and 64-bit)
SuSE LINUX Enterprise Server 10 for Itanium2 CPUs	Yes		Yes
Red Hat Enterprise Linux 3, 4 and 5 for Intel Itanium2 processors	Yes		Yes
Red Hat Enterprise Linux 3, 4 and 5 for Intel IA-32	Yes	Yes	Yes
Red Hat Enterprise Linux 3, 4 and 5 for EM64T and AMD64 CPUs		Yes (32-bit and 64-bit)	Yes (32-bit and 64-bit)

Environment	JDK 1.4	JDK 5.0	BEA WebLogic JRockit 5.0
Oracle Enterprise Linux 4 and 5 for Intel IA-32 CPUs.	Yes	Yes	Yes
Oracle Enterprise Linux 4 and 5 for EM64T and AMD64 CPUs.		Yes (32-bit and 64-bit)	Yes (32-bit and 64-bit)
MontaVista Linux Carrier Grade Edition Release 4.0 for Intel IA-32 and EM64T CPUs	Yes (32-bit and 64-bit)	Yes (32-bit and 64-bit)	Yes (32-bit and 64-bit)
HP-UX 11i, 11i v2 and 11iv3 for PA-RISC 32- and 64-bit	Yes (32-bit and 64-bit)	Yes (32-bit and 64-bit)	
HP-UX 11i v2 and 11iv3 for Itanium2	Yes (32-bit and 64-bit)	Yes (32-bit and 64-bit)	
AIX 5L 5.2 and 5.3 for POWER CPUs	Yes (32-bit and 64-bit) (Uses IBM JDK 1.4.2 or greater)	Yes (32-bit and 64-bit) (Uses IBM JDK)	
Tru64 UNIX 5.1B for Alpha EV68 CPUs	Yes		

Client/Server configurations

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

A TimesTen 6.0 client can connect to a 6.0 TimesTen Server of any patch level. If the `-insecure-backwards-compat` option is set in the `ttendaemon.options` file, a TimesTen 6.0 client can connect to a TimesTen 7.0 or newer server, under certain configurations.

A TimesTen 7.0 or later client can connect to a TimesTen 6.0 or newer server, under certain configurations.

For configuration details see "[Configuring TimesTen Client and Server](#)" in the *Oracle TimesTen In-Memory Database Operations Guide*.

Cache Connect to Oracle

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

- Oracle Database 10g Release 2 (Oracle 10.2.0.1.0 or above)
- Oracle Database 10g Release 1 (Oracle 10.1.0.5.0 or above)
- Oracle Database 11i Release 1 (On Linux x86 32-bit systems only)

See "[Cache Connect support for Oracle client/server releases](#)" in *TimesTen Cache Connect to Oracle Guide*.

Restrictions

The Oracle client version does not need to match the Oracle Server version when using TimesTen Cache Connect to Oracle.

To enable the Cache Connect to Oracle option on Monta Vista Carrier Grade Edition, the 10.2.0.3 version of the Oracle Instant Client must be installed.

Platform support

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 IA-32 and EM64T and AMD64 CPUs	Yes	Yes
Asianux 2.0 for Intel IA-32 and EM64T and AMD64 CPUs.	Yes	Yes
Solaris 10 for AMD64 CPUs systems	Yes	Yes
Solaris 8, 9 and 10 for UltraSparc CPUs	Yes	Yes
SuSE LINUX Enterprise Server 9 and 10 for Intel IA-32, EM64T and AMD64 CPUs	Yes	Yes
SuSE LINUX Enterprise Server 10 for Itanium2 CPUs		Yes

Environment	32-bit	64-bit
Oracle Enterprise Linux 4 and 5 for Intel IA-32 CPUs.	Yes	Yes
Oracle Enterprise Linux 4 and 5 for EM64T and AMD64 CPUs.		Yes
Red Hat Enterprise Linux 3, 4 and 5 running on Intel Itanium2 processors		Yes
Red Hat Enterprise Linux 3, 4 and 5 for Intel IA-32 and EM64T and AMD64 CPUs	Yes	Yes
HP-UX 11i, 11i v2 and 11iv3 for PA-RISC 32-bit and 64-bit	Yes	Yes
HP-UX 11i v2 and 11iv3 for Itanium2	Yes	Yes
AIX 5L 5.2 and 5.3 for POWER CPUs	Yes	Yes
Tru64 UNIX 5.1B for Alpha EV68 CPUs		Yes

Replication configurations

TimesTen-to-TimesTen Replication is supported only between identical platforms and bit-levels.

Installation instances



On UNIX, you can install more than one *instance* of any TimesTen release. By default, the instance name for this release is tt70.

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.

You can retrieve information about the TimesTen instance name, 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 TCP/IP port number for the main TimesTen daemon.

All TimesTen data stores that replicate to each other must use the same daemon port number, except when the `-remoteDaemonPort` option is specified in duplicate operations. 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, 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 TimesTen 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: <ul style="list-style-type: none">• 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 TimesTen and you would like to add or remove components, you must run the installer and select the option “Upgrade an existing instance,” and then select the instance which you would like to change.

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 19](#) to learn which JDK is required for your platform.



General 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 may 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

Filesystem options

On the Veritas file system, if you plan to have TimesTen applications that use `DurableCommits=1`, use the `mincache=direct` and `convosync=direct` options to ensure durability.

Options that convert `dsync` into `sync` or `fdatsync` into `sync` or those that treat all writes such that the file is opened with `O_SYNC` should be avoided.

Semaphores

TimesTen consumes 1 SEMMNI per active data store, plus 1 additional SEMMNI per TimesTen instance where Client/Server communication is done through shared memory. For each active data store, TimesTen consumes 100 SEMMSL if the `Connections` attribute is set to the default value, and one additional SEMMSL for each connection above the default.

Java

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

To run 64-bit Java applications on all systems except AIX systems, if you are using the Sun 64-bit JVM, you may need to pass the `-d64` options to the Java command line.

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 Solaris, you may change the file descriptor limit to have a maximum of 1024 simultaneous 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
```



Semaphores

HP-UX

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

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 `semms` 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 0x40000000 (a 24 followed by seven zeroes) indicates that the largest shared memory segment that can be created is 1024 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 Asianux 2.0, Red Hat Enterprise Linux 3 and 3.1 and 4, the MontaVista Linux Carrier Grade Edition Release 4.0 and SuSE LINUX Enterprise Server 9 and 10 minimal configurations. The C development tools are required if native development will be done on the machine.

Note: TimesTen does not support SELinux. When installing Linux for use with TimesTen, make sure that the SELinux option is disabled.

Large pages Large pages can be enabled only if the running Linux kernel supports large pages (also called “huge pages” in Linux community).

If large pages are supported by the kernel, there should be special files in the `/proc` directory that indicate the number and size of the large pages.

On Linux 2.4.x systems, the `/proc/sys/vm/hugetlb_pool` indicates the total size of the large pages.

On 2.6.x systems, the `/proc/sys/vm/nr_hugepages` file indicates the total number of large pages.

You can change the total number and size of the large pages by changing the contents of those files. For example, you can use:

```
echo "32" > /proc/sys/vm/nr_hugepages
```

To see the number and size of the allocated large pages use:

```
cat /proc/meminfo
```

The following output from this command would indicate that you have 16 large pages, each of the size 256MB for a total of 4GB:

```
HugePages_Total: 16  
HugePages_Free: 16  
Hugepagesize: 262144 kB
```

Note: Since large pages must be allocated on a contiguous memory space, the actual large page size allocated may be smaller than requested. Also, the large page size itself is not configurable. The value of `Hugepagesize` in `/proc/meminfo` indicates the system's fixed large page size.

If PAM (Pluggable Authentication Modules) is enabled, You may need to change the `/etc/security/limits.conf` file.

You must also set `/proc/sys/vm/hugetlb_shm_group` to the group ID of the user that is running the main TimesTen daemon. If TimesTen is installed as `root`, specify 0. This is the default.

The OS now is ready for the large page support. To enable this feature on TimesTen, simply set `-linuxLargePageAlignment Size_in_MB` in the daemon options file (`ttendaemon.options`).

You should specify the large page alignment size in MB, which is the `Hugepagesize` value in `/proc/meminfo`.

Once you set up large pages, TimesTen uses as many large pages as possible. If there are not enough pages, TimesTen uses the normal pages after consuming all available large pages.

When TimesTen uses large pages, the HugePages_Free file in /proc/meminfo changes.

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 2048 MB, for example, as root, edit the /etc/sysctl.conf file by adding the line:

```
kernel.shmmax=2147483648
```

If your configuration is greater than 8GB, you should also increase the value of the shmall parameter. The value is in KB and should be equal to ceil(SHM_MAX/PAGE_SIZE). Page size is generally 4K on x86 systems and 16K on Itanium. For example, for a 64GB data store on Itanium, you should specify the following parameters values:

```
kernel.shmmax=68719476736
kernel.shmall=4194304
```

To increase the shared memory size without rebooting, use:

```
% /sbin/sysctl -w kernel.shmmax=2147483648
```

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 2147483648 > /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 128 100"
```

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

To increase the shared memory size without rebooting, use:

```
% /sbin/sysctl -w kernel.sem = "250 32000 128 100"
```

This sets the parameter values as follows:

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

Client/Server and Cache Administrator

If you are installing the Cache Connect to Oracle option and plan to use the web-based Cache Administrator or if you plan to use TimesTen client/server configurations, 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. To make these changes, add the lines:

```
net.ipv4.tcp_rmem="4096 4194304 4194304"  
net.ipv4.tcp_wmem="98304 4194304 4194304"  
net.ipv4.tcp_mem="98304 4194304 4194304"  
net.core.rmem_default=65535  
net.core.wmem_default=65535  
net.core.rmem_max=4194304  
net.core.wmem_max=4194304  
net.ipv4.tcp_window_scaling=1
```

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

Cache Connect

For Cache Connect, TCP send and receive buffers should be increased to even greater values. To make these changes, add the lines:

```
net.ipv4.tcp_rmem="4096 4194304 4194304"  
net.ipv4.tcp_wmem="98304 4194304 4194304"  
net.ipv4.tcp_mem="98304 4194304 4194304"  
net.core.rmem_default=262144  
net.core.wmem_default=262144  
net.core.rmem_max=4194304  
net.core.wmem_max=4194304  
net.ipv4.tcp_window_scaling=1  
net.ipv4.ip_local_port_range="1024 65000"
```

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



Operating
system
patches

Solaris

Solaris 8 requires patch 108827-36 or later.

To view a list of installed patches, use:

```
% showrev -p
```

Filesystem options

In addition to the filesystem options listed in the section “[General UNIX requirements](#)” on page 25, on Solaris UFS file systems, if you plan to have TimesTen applications that use `DurableCommits=1`, intensive mount the file system with the `-forcedirectio` option.

IPC semaphores

On Solaris 8 and 9, 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`.

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_semmnl = 512
set semsys:seminfo_semmns = 2000
set semsys:seminfo_semmnu = 2000
```

Note: The values in this step are the minimum number of required semaphores. You can increase these numbers as needed. You can use the following formula as a guide, although in practice, SEMMNS and SEMMNU can be much less than SEMMNI * SEMMSL because not every program in the system needs semaphores.

$$\text{SEMMNS}=\text{SEMMNU} = (\text{SEMMNI} * \text{SEMMSL}).$$

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
```

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 = 0x240000000
```

Note: The value `0x240000000` (a 24 followed by seven zeroes) indicates that the largest shared memory segment that can be created is 1024 MB. The size of the shared memory segment required for a data store is larger than the data store size permanent 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
```

Default installation directories

The TimesTen default installation directories for release 7.0 are:

- On Windows, `C:\TimesTen\tt70`
- On HP-UX, Solaris and Linux, `/opt/TimesTen/tt70`
- On AIX, `/usr/lpp/TimesTen/tt70`

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 large deletes, use the temporary directory when copying files.

The temporary directory is operating system-dependent. Usually it is located in these directories:

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

You can change 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.

Note: On Windows, the complete temporary directory path must be less than 190 characters for the installation to complete successfully. In addition, TimesTen does not support file path names that contain multi-byte characters. Please make sure that the installation path, data store path, transaction log path, and temporary file path do not contain any multibyte characters.

Cache Connect

If you are using the Cache Connect to Oracle option of TimesTen, you must have at least a client installation of Oracle Database 9i or 10g on the machine where you are installing TimesTen.

Oracle client shared libraries are required in order to cache Oracle data in TimesTen by Cache Connect to Oracle. You must have Oracle Database 9i client or Oracle Database 10g installed. You also must have the `ORACLE_HOME` environment variable defined before running the installer. See [“ORACLE_HOME environment variable” on page 79](#).

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 once as user root. See the [“Prerequisites for non-root installations 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 and permissions 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 Reference Guide](#).

Prerequisites for non-root installations on UNIX systems



As discussed in [Chapter 1, “Access Control,”](#) 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 may need to 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 25](#) 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. These procedures 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`.
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 name that you prefer.
3. 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, if you want to use the MemoryLock feature of TimesTen.

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 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 TimesTen 5.1, you must change the existing group ownership and permissions of the existing `/etc/TimesTen` directory on each machine that you wish to install TimesTen 7.0. 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, to install the TimesTen daemon start scripts in the proper system locations, the user root must run the `setuproot` script located in the `install_dir/bin` directory:

```
# setuproot -install
```

This step is only necessary if you want the TimesTen instance to start each time the machine is rebooted.

Note: If you install these scripts into your system directory, you must manually remove them in the case that you want to uninstall your TimesTen instance, using

```
# setuproot -uninstall
```

Configure the syslog messages

For non-root installs, the default location for daemon system message logs is to a file within the installation directory. For root installs, the default location is the `syslog` mechanism. See [“Informational messages on UNIX systems” on page 73](#) for details.

Changing the daemon port number on UNIX



The **ttmodinstall** utility allows 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**, the utility stops the daemon before changing the port number. After the port change, the daemon is automatically restarted.

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

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

```
% ttmodinstall -port 12345
```

The **ttmodinstall** utility can also enable Access Control and modify the path supplied to the `ORACLE_HOME` environment variable. All 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 Data Manager library and ODBC driver. In the `sys.odbcc.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 84](#) for more information on the `sys.odbcc.ini` file. Also see ["User and system DSNs"](#) in *Oracle TimesTen In-Memory Database Operations Guide*.

Installing TimesTen on Windows systems



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 19](#).

Note: Before beginning installation, be sure that the prerequisites defined in [“Installation prerequisites” on page 25](#) have been met.

Installing TimesTen

An InstallShield program installs your TimesTen instance 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 TimesTen exists, the setup program starts in Maintenance Mode, which allows you to uninstall or repair the existing TimesTen product. In order to do a install a new version of TimesTen where the first and second version number (e.g. 6.1.2 and 6.1.5) match, you must first uninstall TimesTen in Maintenance Mode and then run `SETUP.EXE` again.

The TimesTen installation prompts you to make these choices at installation time:

- Which component would you like to install?

See [“Components available on Windows” on page 24](#).

- Do you want to install the Cache Connect to Oracle option?

If you intend to cache Oracle data in a TimesTen cache group, select this option. You can incrementally install this option at a later time, as well, using the Modify option to the installation script when the major and minor release numbers of the TimesTen installation match exactly.

- Do you want to enable Access Control?

By default, Access Control is not enabled. See [Chapter 1, “Access Control”](#) for more details.

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 subdirectory 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`.

You can now use this file to run an installation in silent mode:

- From a command-line, run: `setup.exe -s -flresponse_file`. For example:

```
C:> setup.exe -s -flC:\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 7.0 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 when batch files from different versions are used.

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 7.0 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 TimesTen Client, you should see the **TimesTen Client 7.0** driver. Click **OK**.
4. On the Windows Desktop, choose **Start > Settings > Control Panel > Administrative Tools > Services** and check that the TimesTen Data Manager 7.0 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 **RunDataCS_tt70**, **RunDataCS_tt70_64** or **ShmRunDataCStt70_64** sample data source and click **Configure**.

Note: The **RunDataCS_tt70** DSN is used for client applications that use TCP/IP communications with the TimesTen Server, on 32-bit platforms. On 64-bit platforms, use the **RunDataCS_tt70_64** DSN.

The **ShmRunDataCStt70** DSN is used for client applications that use shared memory to communicate with a TimesTen Server on the same machine. On 64-bit platforms, use the **ShmRunDataCStt70_64** DSN.

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 7.0**.
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 7.0**.

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**.
- Alternatively, you can use the **Modify** option to the TimesTen Installation script to uninstall just the Cache Connect to Oracle option from TimesTen.

To verify that removal was successful, check that:

- The TimesTen 7.0 Start menu shortcut has been removed from the **Start > Programs** menu.
- The TimesTen Data Manager 7.0 has been removed from the **Services** list.
- The TimesTen 7.0 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 25](#) 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 user root.
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 explicitly mount the CD; it may be automatically mounted at:

```
/cdrom/tt7.0
```

- 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.
- To add the Cache Connect to Oracle option to an existing TimesTen installation, use the `-installCache` option with the setup script.
- To uninstall just the Cache Connect to Oracle option from an existing TimesTen installation, use the `-removeCache` option with the setup script.

Note: To install or make modifications to an instance, run `setup.sh` from the installation media. To uninstall, run `setup.sh -uninstall`.

The `setup.sh` script takes these options:

Option	Description
<code>-install</code>	Installs TimesTen.
<code>-uninstall</code>	Uninstalls TimesTen

<code>-batch</code> <i>filename</i>	Installs or uninstalls TimesTen without having to respond to prompts. If <i>filename</i> is specified, the installation reads all installation prompts from the file. The batch file <i>filename</i> 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 “tt70” for the instance name. If an instance with the same name already exists on the installation machine, the install procedure fails.
<code>-installDoc</code>	Installs the TimesTen documentation.
<code>-installCache</code>	Installs the Cache Connect to Oracle option
<code>-removeCache</code>	Incrementally uninstalls the Cache Connect to Oracle option.
<code>-help</code>	Displays the help message.
<code>-record</code> <i>filename</i>	Installs or uninstalls TimesTen and records responses to prompts described in <i>filename</i> . The file can then be used as the parameter to the <code>-batch</code> option.
<code>-verbose</code>	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 or make modifications to an instance, run `setup.sh` from the installation media. To uninstall, run `setup.sh -uninstall`. Batch files from older releases of TimesTen cannot 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 when batch files from different versions are used.

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 to:
 - Install a new instance
 - Upgrade an existing instance (This option allows you to incrementally install the Cache Connect option. The major and minor version numbers of the TimesTen release must match exactly.)
 - Display information about an existing instance or
 - Quit the installation.
- Prompts you to choose the default instance name or choose a name for your TimesTen instance. See [“Installation instances” on page 23](#).
- Prompts you to install as root or a non-root user.
- Prompts you to install TimesTen:
 - Oracle TimesTen In-Memory Database
 - Oracle TimesTen In-Memory Database with Cache Connect to Oracle
- Prompts you to install one of the following components.
 - Client/ Server and Data Manager
 - Data Manager only
 - Client only
- Prompts you for the location of your TimesTen installation and specific files and directories.
- 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 17000 for 32-bit installations and 17001 for 64-bit applications, prompts you to use the default port number. If the default port number is already in use, prompts you to enter a unique port number.
- 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”](#) in this guide.
- Prompts you for the TimesTen Server port number.

- 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/tt70`.
- 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`, if installed as `root`, or `install_dir/info/DemoDataStores` if installed as a non-root user.
- 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 daemon and Server

The TimesTen main daemon (`timestend`) starts automatically when the operating system is booted, if the instance startup scripts have been installed in `/etc/init.d/`, 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, use the utility command:

```
ttDaemonAdmin -stop
```

To start the daemon manually, 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 user root.
2. The TimesTen setup script is in the `install_dir/bin` directory. To uninstall or modify the installation, run the script with the `-uninstall` option from the installation media.

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.

- To uninstall just the Cache Connect to Oracle option from an existing TimesTen installation, from the installation media, use:

```
# ./setup.sh -removeCache
```

Installing TimesTen on HP-UX systems



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 25](#) 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 user root.
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.

- To add the Cache Connect to Oracle option to an existing TimesTen installation, use the `-installCache` option with the setup script.
- To uninstall just the Cache Connect to Oracle option from an existing TimesTen installation, use the `-removeCache` option with the setup script.

Note: To install or make modifications to an instance, run `setup.sh` from the installation media. To uninstall, run `setup.sh -uninstall`.

Specify the option at the end, outside the quotation marks. The `setup.sh` script takes these options:

Option	Description
<code>-install</code>	Installs TimesTen.
<code>-uninstall</code>	Uninstalls TimesTen

<code>-batch</code> <i>filename</i>	Installs or uninstalls TimesTen without having to respond to prompts. If <i>filename</i> is specified, the installation reads all installation prompts from the file. The batch file <i>filename</i> 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 “tt70” for the instance name. If an instance with the same name already exists on the installation machine, the install procedure fails.
<code>-installDoc</code>	Installs the TimesTen documentation.
<code>-installCache</code>	Installs the Cache Connect to Oracle option
<code>-removeCache</code>	Incrementally uninstalls the Cache Connect to Oracle option.
<code>-help</code>	Displays the help message.
<code>-record</code> <i>filename</i>	Installs or uninstalls TimesTen and records responses to prompts described in <i>filename</i> . The file can then be used as the parameter to the <code>-batch</code> option.
<code>-verbose</code>	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 or uninstall TimesTen without having to respond to prompts, use the `-batch` flag with the `setup.sh` script. Batch files from older releases of TimesTen cannot 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 when batch files from different versions are used.

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 to:
 - Install a new instance
 - Upgrade an existing instance (This option allows you to incrementally install the Cache Connect option. The major and minor version numbers of the TimesTen release must match exactly.)
 - Display information about an existing instance or
 - Quit the installation.
- Prompts you to choose the default instance name or choose a name for your TimesTen instance. See [“Installation instances” on page 23](#).
- Prompts you to install TimesTen:
 - Oracle TimesTen In-Memory Database
 - Oracle TimesTen In-Memory Database with Cache Connect to Oracle
- Prompts you to install one of the following components.
 - Client/ Server and Data Manager
 - Data Manager only
 - Client only
- Prompts you for the location of your TimesTen installation and specific files, if installing as a non-root user.
- 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 17000 for 32-bit installations and 17001 for 64-bit applications, prompts you to use the default port number.
- 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”](#) in this guide.
- Prompts you for the TimesTen Server port number.
- 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/tt70`.

- 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 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 main 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, use the utility command:

```
ttDaemonAdmin -stop
```

To start the daemon manually, 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 user root.
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 directory 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.

- To uninstall just the Cache Connect to Oracle option from an existing TimesTen installation, from the installation media, use:

```
# ./setup.sh -removeCache
```

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:

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

To return to `EXEC_MAGIC`, use:

```
# chattr -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

```
# chattr binary
```

The `chattr(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 shmmmax`
- 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?
`% ttIsq1 -connStr dsn=my_dsn`
- Can you successfully run a demo program? The TimesTen demos are located under `install_dir/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 user root.
2. Insert the CD-ROM into the CD-ROM drive.
3. Enter:

```
# crfs -v cdrfs -p ro -d cd0 -m /usr/cdrom/
TimesTen7.0
```

This creates the directory into which you will mount the CD-ROM.

4. To mount the CD-ROM, enter:

```
# mount /usr/cdrom/TimesTen7.0
```

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

5. 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/TimesTen7.0).

- To add the Cache Connect to Oracle option to an existing TimesTen installation, use the `-installCache` option with the startup script.
- To uninstall just the Cache Connect to Oracle option from an existing TimesTen installation, use the `-removeCache` option with the setup script.

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. The `setup.sh` script takes these options:

Option	Description
<code>-install</code>	Installs TimesTen.
<code>-uninstall</code>	Uninstalls TimesTen
<code>-batch <i>filename</i></code>	Installs or uninstalls TimesTen without having to respond to prompts. If <i>filename</i> is specified, the installation reads all installation prompts from the file. The batch file <i>filename</i> 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 “tt70” for the instance name. If an instance with the same name already exists on the installation machine, the install procedure fails.
<code>-installDoc</code>	Installs the TimesTen documentation.

<code>-installCache</code>	Installs the Cache Connect to Oracle option
<code>-removeCache</code>	Incrementally uninstalls the Cache Connect to Oracle option.
<code>-help</code>	Displays the help message.
<code>-record <i>filename</i></code>	Installs or uninstalls TimesTen and records responses to prompts described in <i>filename</i> . The file can then be used as the parameter to the <code>-batch</code> option.
<code>-verbose</code>	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 or uninstall TimesTen without having to respond to prompts, use the `-batch` flag with the `setup.sh` script. Batch files from older releases of TimesTen cannot 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 when batch files from different versions are used.

The file 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 to:
 - Install a new instance
 - Upgrade an existing instance (This option allows you to incrementally install the Cache Connect option. The major and minor version numbers of the TimesTen release must match exactly.)
 - Display information about an existing instance or
 - Quit the installation.

- Prompts you to choose the default instance name or choose a name for your TimesTen instance. See [“Installation instances” on page 23](#).
- Prompts you to install TimesTen:
 - Oracle TimesTen In-Memory Database
 - Oracle TimesTen In-Memory Database with Cache Connect to Oracle
- Prompts you to install one of the following components.
 - Client/ Server and Data Manager
 - Data Manager only
 - Client only
- Prompts you for the location of your TimesTen installation and specific files, if installing as a non-root user.
- 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 17000 for 32-bit installations and 17001 for 64-bit applications, prompts you to use the default port number.
- 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”](#) in this guide.
- Prompts you for the TimesTen Server port number.
- 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, default `/usr/lpp/TimesTen/tt70`.
- 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 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, use the utility command:

```
ttDaemonAdmin -stop
```

To start the daemon manually, use the utility command:

```
ttDaemonAdmin -start
```

To determine the status of the daemon at any time, use the `ttStatus` utility.

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 user `root`.

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 directory:

```
# 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 25 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 user root.
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
```

- To add the Cache Connect to Oracle option to an existing TimesTen installation, use the `-installCache` option with the startup script.
- To uninstall just the Cache Connect to Oracle option from an existing TimesTen installation, use the `-removeCache` option with the setup script.

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. The `setup.sh` script takes these options:

Option	Description
<code>-install</code>	Installs TimesTen.
<code>-uninstall</code>	Uninstalls TimesTen
<code>-batch filename</code>	Installs or uninstalls TimesTen without having to respond to prompts. If <i>filename</i> is specified, the installation reads all installation prompts from the file. The batch file <i>filename</i> 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 “tt70” for the instance name. If an instance with the same name already exists on the installation machine, the install procedure fails.
<code>-installDoc</code>	Installs the TimesTen documentation.
<code>-installCache</code>	Installs the Cache Connect to Oracle option
<code>-removeCache</code>	Incrementally uninstalls the Cache Connect to Oracle option.
<code>-help</code>	Displays the help message.
<code>-record filename</code>	Installs or uninstalls TimesTen and records responses to prompts described in <i>filename</i> . The file can then be used as the parameter to the <code>-batch</code> option.
<code>-verbose</code>	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 or uninstall TimesTen without having to respond to prompts, use the `-batch` flag with the `setup.sh` script. Batch files from older releases of TimesTen cannot 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 when batch files from different versions are used.

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

- Prompts you to:
 - Install a new instance
 - Upgrade an existing instance (This option allows you to incrementally install the Cache Connect option. The major and minor version numbers of the TimesTen release must match exactly.)
 - Display information about an existing instance or
 - Quit the installation.
- Prompts you to choose the default instance name or choose a name for your TimesTen instance. See [“Installation instances” on page 23](#).
- Prompts you to install TimesTen:
 - Oracle TimesTen In-Memory Database
 - Oracle TimesTen In-Memory Database with Cache Connect to Oracle
- Prompts you to install one of the following components.
 - Client/ Server and Data Manager
 - Data Manager only
 - Client only
- Prompts you for the location of your TimesTen installation and specific files, if installing as a non-root user.
- 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 17000 for 32-bit installations and 17001 for 64-bit applications, prompts you to use the default port number.
- 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](#) in this guide.

- Prompts you for the TimesTen Server port number.
- 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/tt70`.
- 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 main 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, use the utility command:

```
ttDaemonAdmin -stop
```

To start the daemon manually, 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 user root.
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 directory, 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 25 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 user root.
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
```

- To add the Cache Connect to Oracle option to an existing TimesTen installation, use the `-installCache` option with the startup script.
- To uninstall just the Cache Connect to Oracle option from an existing TimesTen installation, use the `-removeCache` option with the setup script.
- 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 install or make modifications to an instance, run `setup.sh` from the installation media. To uninstall, run `setup.sh -uninstall`.

The `setup.sh` script takes these options:

Option	Description
<code>-install</code>	Installs TimesTen.
<code>-uninstall</code>	Uninstalls TimesTen
<code>-batch filename</code>	Installs or uninstalls TimesTen without having to respond to prompts. If <i>filename</i> is specified, the installation reads all installation prompts from the file. The batch file <i>filename</i> 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 “tt70” for the instance name. If an instance with the same name already exists on the installation machine, the install procedure fails.
<code>-installDoc</code>	Installs the TimesTen documentation.

<code>-installCache</code>	Installs the Cache Connect to Oracle option
<code>-removeCache</code>	Incrementally uninstalls the Cache Connect to Oracle option.
<code>-help</code>	Displays the help message.
<code>-record filename</code>	Installs or uninstalls TimesTen and records responses to prompts described in <i>filename</i> . The file can then be used as the parameter to the <code>-batch</code> option.
<code>-verbose</code>	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 or uninstall TimesTen without having to respond to prompts, use the `-batch` option with the `setup.sh` script. Batch files from releases older than TimesTen Release 7.0 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 when batch files from different versions are used.

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

- Prompts you to:
 - Install a new instance
 - Upgrade an existing instance (This option allows you to incrementally install the Cache Connect option. The major and minor version numbers of the TimesTen release must match exactly.)
 - Display information about an existing instance or
 - Quit the installation.
- Prompts you to choose the default instance name or choose a name for your TimesTen instance. See [“Installation instances” on page 23](#).
- Prompts you to install TimesTen:
 - Oracle TimesTen In-Memory Database

- Oracle TimesTen In-Memory Database with Cache Connect to Oracle
- Prompts you to install one of the following components.
 - Client/ Server and Data Manager
 - Data Manager only
 - Client only
- Prompts you for the location of your TimesTen installation and specific files, if installing as a non-root user.
- 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 17000 for 32-bit installations and 17001 for 64-bit applications, prompts you to use the default port number.
- 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](#) in this guide.
- Prompts you for the TimesTen Server port number.
- 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/tt70`.
- 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 main 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, use the utility command:

```
ttDaemonAdmin -stop
```

To start the daemon manually, 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`.
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 directory, 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.

- To uninstall just the Cache Connect to Oracle option from an existing TimesTen installation, from the installation media, use:

```
# ./setup.sh -removeCache
```

Using the Cache Administrator

The Cache Administrator is a web-based tool used to set cache definitions. This feature is available on systems where the Cache Connect to Oracle option has been installed. See [“Cache Connect to Oracle” on page 22](#).

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 web server port number that was configured during the installation of the Cache Connect to Oracle option. Its value is stored in the `PORT` variable in the `/var/TimesTen/TTinstance/webserver.config` file on UNIX systems for root installs, `install_dir/info/webserver.config` for non-root UNIX installs, or the `install_dir\srv\info\webserver.config` on Windows systems.

Note: The Cache Administrator will not work if it is running on a Windows machine with the Windows Firewall ON (this is the usual setting). In this case, the Windows Administrator must add an exception to allow the Cache Administrator to connect through the Windows Firewall.

For details on setting the environment variables required to use Cache Connect to Oracle, see [“Environment modifications” on page 75](#).

For details on setting up the web server, see [“Web server configuration” on page 80](#).

The following web browsers are supported for the Cache Administrator:

- Internet Explorer 6.0
- Firefox 1.5 and greater

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 7.0**” in the “Source” column were generated by the TimesTen Data Manager service.
 - Messages with the phrase “**TimesTen Server 7.0**” in the “Source” column were generated by the TimesTen Server service.
 - Messages with the phrase “**TimesTen Replication 7.0**” 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 “[Utilities](#)” in the *Oracle TimesTen In-Memory Database API 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. At installation time, you determine whether these messages go into a file or to the `syslog` facility.

For root installs, TimesTen logs daemon messages using the `LOG_USER` facility defined by `syslog`, by default.

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 support or error 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 determine if your `syslog` 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.

Incremental install and uninstall of Cache Connect

TimesTen allows you to incrementally install the Cache Connect to Oracle option after having completed an installation. Likewise, you can uninstall just the Cache Connect option of TimesTen.

On UNIX, to incrementally install the Cache Connect option, use the `-installCache` option when installing TimesTen

On Windows, use the following steps:

1. Open the **Add or Remove Programs** control panel.
2. Choose **Oracle TimesTen In-Memory Database**.
3. Click the **Change/Remove** button.
4. In the next dialog, select the **Modify** radio button and click Next.
5. Check **Cache Connect for Oracle** checkbox in the Select Features list and click Next.

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 available separately from Microsoft 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, Microsoft Visual Studio .NET or Microsoft Visual Studio 2005. 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 78 and “Using the Cache Administrator” on page 72.
LIB, LIBPATH, LD_LIBRARY_PATH or SHLIB_PATH	On UNIX systems, include the lib directory under the TimesTen installation directory	“Shared library path environment variable” on page 79.
ODBCINI	The location where the odbc.ini file used by TimesTen data stores is to be found.	“ODBCINI environment variable” on page 77
ORACLE_HOME	If using the Cache Connect to Oracle option, set to the location of the Oracle installation. Required if you are using the Cache Connect to Oracle option.	“ORACLE_HOME environment variable” on page 79 and “Using the Cache Administrator” on page 72
PATH	Include the bin directory under the TimesTen installation directory. On Windows, also include the path to the Oracle installation if you are using the Cache Connect to Oracle option.	“PATH environment variable” on page 77 , “Shared library path environment variable” on page 79 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 78

Environment Variable	What to include	For settings and other information, see:
SYSTTCONNECTINI	Set to the location where the <code>sys.ttconnect.ini</code> file used by TimesTen Client applications to define logical server names.	“SYSTTCONNECTINI environment variable” on page 78
TMP or TMPDIR	Set to the location of the temporary directory. TimesTen uses this directory during recovery and other operations.	“Default installation directories” on page 36

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.



On Windows, the PATH environment variable must also contain the `bin` directory of the ORACLE installation, if you are using the Cache Connect to Oracle option.



ODBCINI environment variable

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 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 84](#) 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

TimesTen applications use the `sys.odbcc.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 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.odbcc.ini`. To override the name and location of this file at run-time, set the `$SYSODBCINI` environment variable to the pathname of a `sys.odbcc.ini` file before launching the TimesTen application.

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

Also, see [“Defining data sources for the demo applications”](#) on page 84 for more information on the `.odbcc.ini` file.



SYSTTCONNECTINI environment variable

TimesTen client applications use the `sys.ttconnect.ini` file to define logical server names. For a description of logical server names, see [Chapter 2, “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`.



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

CLASSPATH environment variable

On Windows and UNIX platforms, add `install_dir/demo` and `install_dir/lib/ttjdbcjdk_version.jar` to the CLASSPATH environment variable. For example, for JDK 5.0, set the CLASSPATH environment variable to: `install_dir/lib/ttjdbc5.jar`.

ORACLE_HOME environment variable

On platforms where the Cache Connect to Oracle option is supported, to work with Oracle data, the TimesTen cache agent must be running. This requires that the `ORACLE_HOME` environment variable be set to the path of the Oracle Database 9i or 10g installation at the time that you install TimesTen.



The **`ttmodinstall`** utility allows the instance administrator to change the path supplied to the `ORACLE_HOME` environment variable after installation. This feature is useful if you install TimesTen and later find that the Oracle installation has been moved.

The utility is run from the command line and takes the `-changeOracleSettings` option, which will prompt you to supply the new path name.



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 the Cache Connect to Oracle option, add `$ORACLE_HOME/lib` to `LD_LIBRARY_PATH`. See [“ORACLE_HOME environment variable” on page 79](#).



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. If you are using the Cache Connect to Oracle option, `SHLIB_PATH` must also contain `$ORACLE_HOME/lib32` and must not contain `$ORACLE_HOME/lib`. See [“ORACLE_HOME environment variable” on page 79](#).

On HP-UX 64-bit systems, add `install_dir/lib` to the `LD_LIBRARY_PATH` environment variable. If you are using the Cache Connect to Oracle option, `SHLIB_PATH` must also contain `$ORACLE_HOME/lib` and must not contain `$ORACLE_HOME/lib32`. See [“ORACLE_HOME environment variable” on page 79](#).

TRU 64

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 the Cache Connect to Oracle option is installed. 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 3, “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. The options should only be changed at the request of TimesTen Customer Support.

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.

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 7.0

TimesTen 7.0 cannot read data stores created with earlier releases of TimesTen. TimesTen 7.0 includes two migration utilities: **ttMigrate** and **ttBulkCp**. These utilities allow you to migrate data stores from older TimesTen releases to TimesTen Release 7.0.

For a description of these utilities, see "[Utilities](#)" in *Oracle TimesTen In-Memory Database API 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 "Utilities" in the *Oracle TimesTen In-Memory Database API 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 6.0 data store is called `Sales600`;
- The data file you wish to use is called `sales.dat`; and
- TimesTen 6.0 is installed in `/opt/TimesTen60` on UNIX platforms or `C:\TimesTen\TimesTen60` on Windows.

On Windows, use:



```
C:\ > "C:\TimesTen\tt60\bin\ttMigrate"  
-c DSN=Sales600 sales.dat
```



On UNIX, use:

```
% /opt/tt60/32/bin/ttMigrate -c DSN=Sales600  
sales.dat
```

2. Use **ttMigrate** to restore the saved tables in the new data store.
3. Create a new data source name, `Salestt70` for the TimesTen 7.0 data store, and import the 6.0 data store:

If, for example:

- TimesTen 7.0 is installed in `/opt/TimesTen/tt70` on UNIX platforms or `C:\TimesTen\tt70` on Windows.



On Windows, use:

```
C:\ > "C:\TimesTen\tt70\bin\ttMigrate" -r DSN=Salestt70  
sales.dat
```



On UNIX, use:

```
% /opt/TimesTen/tt70/bin/ttMigrate -r DSN=Salestt70
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 "Utilities" in *Oracle TimesTen In-Memory Database API 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 6.0 to save the tables to disk files and use **ttIsql** and **ttBulkCp** Release 7.0 to copy the disk files into the tables of the new data store.

To import data from a data store created with TimesTen6.0:

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 6.0 is installed in: /opt/TimesTen6.0/32 and release 7.0 is installed in /opt/TimesTen/tt70;
 - Your DSN for release 6.0 is called `source600` and your DSN for release 7.0 is `source_tt70`.
 - 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 `source600` to `source_tt70`.

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

```
% /opt/tt60/32/bin/ttBulkCp -o
    DSN=source600 able able.save
% /opt/tt60/32/bin/ttBulkCp -o
    DSN=source600 baker baker.save
```

Next create a new data source name, `source_tt70` for the TimesTen 7.0 data store, and execute the commands:

```
% /opt/TimesTen/tt70/bin/ttIsql -connStr
    DSN=source_tt70 -f create.sql
% /opt/TimesTen/tt70/bin/ttBulkCp -i
    DSN=source_tt70 able able.save
% /opt/TimesTen/tt70/bin/ttBulkCp -i
    DSN=source_tt70 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 files used in demos that provide examples for the TimesTen documentation. For a description of these demos, see the `README.txt` file at the top of these directories.

Note: By default, the TimesTen demo applications save data store files to `/var/TimesTen/TTinstance/demo/DemoDataStore` on UNIX for root installs, and `install_dir/info/DemoDataStores` on non-root installs. 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.odbcc.ini`, if your product was installed as root. If installed by a non-root user the file is located in `install_dir/info/sys.odbcc.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.

Problems running the demo programs

Make sure you run the `install_dir/demo/ttdemoenv.sh`, `.csh` or `.bat` file to set up your demo environment correctly. 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 for a root installation is:

- `/opt/TimesTen/TTinstance/` on Solaris, HP-UX and Linux.
- `/usr/lpp/TimesTen/TTinstance/` on AIX.

- Is the TimesTen main daemon (`timestend`) running? See "[Starting and stopping the daemon on UNIX](#)" in the *Oracle TimesTen In-Memory Database Operations 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**.

Building and running the JDBC demo applications

Source code for a demo application is provided in the `install_dir/demo/jdbc` 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, source the `install_dir/demo/ttdemoenv.sh` or `ttdemoenv.csh` on Unix or run `ttdemoenv.bat` on Windows, first, to set up your demo environment correctly.

You can use the `ttdemoenv.sh` or `ttdemoenv.csh` in the `demo` directory on UNIX or `ttdemoenv.bat` on Windows 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 84](#). 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>.

Online documentation is installed in the `install_dir/doc` directory.

Note: The online documentation represents the most current release of the documentation.

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 who is a member of the local Administrators group.
- You have installed all required operating system patches.
- You have made all required kernel configuration changes.

Data Store Upgrades

Introduction

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

Data store compatibility

Starting with TimesTen version 7.0.0.0.0, TimesTen version numbers consist of five components. Prior versions of TimesTen used only three numbers to indicate the version, such as 5.1.35. The first two numbers in the version are used to indicate a major release of TimesTen, such as 5.1.x or 7.0.x.y.z. The third number indicated the patch release of a major release of TimesTen. For example, TimesTen version number 5.1.35 indicates the 35th patch release of TimesTen version 5.1.

TimesTen data stores are not compatible between major releases, but they are always compatible between patch releases. For example, a data store created with TimesTen version 5.1.35 is not compatible with a TimesTen version 7.0.0.0.0 application, but a data store created with TimesTen 7.0.0.0.0 will be compatible with a TimesTen version 7.0.1.0.0 application.

When referring to a TimesTen version, the version number will often be abbreviated to the major version number. For example, version 7.0.0.0.0 may be abbreviated to 7.0.

Data type compatibility

Beginning with TimesTen version 7.0, TimesTen supports a selection of Oracle data types in addition to the original TimesTen data types that are maintained for backward compatibility. For details on both the new and backward-compatible data types, see [“Type specifications” on page 8](#) in

the *Oracle TimesTen In-Memory Database SQL Reference Guide*. Because some of the new Oracle data types have the same names as the backward-compatible TimesTen data types, a set of aliases has been added for addressing the data types. Which data types the aliases refer to depends on the **TypeMode** that has been set for the data store. See “**TypeMode**” on page 21 of the *Oracle TimesTen In-Memory Database API Reference Guide* for more information.

TimesTen backward-compatible data types in version 7.0 are replication-compatible with the data types in versions of TimesTen prior to 7.0. However, TimesTen backward-compatible data types are not compatible with TimesTen Cache Connect to Oracle, only the new Oracle data types can be used with Cache Connect to Oracle. If you wish to use Cache Connect to Oracle, you must convert any original TimesTen data types to the new Oracle data types when performing the data store upgrade with **ttMigrate**. See “**Converting data types to Oracle data types**” on page 92 for details.

Oracle data types are not replication-compatible with versions of TimesTen prior to 7.0. If you wish to perform an upgrade that requires replication with a version of TimesTen from before 7.0, you must upgrade the original data types as TimesTen data types. See “**Upgrading data types as TimesTen data types**” on page 92 for more information.

Data store character set

Beginning with TimesTen 7.0, TimesTen requires a data store to be configured to support a specific character set when it is created. The character set for the data store is specified using the data store attribute **DatabaseCharacterSet**. The value of this attribute is used to determine which characters may be input to and output from character fields, and how character data is stored and sorted. See “**Choosing a database character set**” on page 74 of the *Oracle TimesTen In-Memory Database Operations Guide* for more information.

Before upgrading your data store to TimesTen 7.0, you must specify a data store character set by adding the **DatabaseCharacterSet** attribute to your data store’s DSN. This attribute will be ignored by versions of TimesTen prior to 7.0. In most cases, you will want to choose a data store character set that makes sense for your region and that matches the character data that is already present in your data store. However, there are three important restrictions you must consider:

- If you plan to use the data store with TimesTen Cache Connect to Oracle, you must specify a value for **DatabaseCharacterSet** that is

the same as the character set specified for the Oracle database that the TimesTen data store connects to.

- Replication is not possible between data stores with different character sets. Because data stores created with versions of TimesTen prior to 7.0 do not have a data store character set specified, a special data store character set, TIMESTEN8, has been created, which allows replication compatibility between data stores created by TimesTen 7.0 and those created by earlier releases. If you plan to perform the data store upgrade as an online upgrade with replication (see [“Performing an online upgrade with replication” on page 106](#)), then you must specify a **DatabaseCharacterSet** of TIMESTEN8 in your TimesTen 7.0 DSN.
- If you use TimesTen Client/Server and intend to connect to the upgraded data store with an application linked to a Client ODBC library from a version prior to TimesTen 7.0, you must specify a **DatabaseCharacterSet** of TIMESTEN8 in your TimesTen 7.0 DSN in order to ensure compatibility. See [“Performing a Client/Server online upgrade from a TimesTen version prior to 6.0” on page 115](#) or [“Performing a Client/Server online upgrade from TimesTen version 6.0 and above” on page 118](#).

Note: The TIMESTEN8 data store character set is intended for use only when transitioning from a version of TimesTen prior to 7.0. When you no longer need your data store to replicate to a pre-7.0 version of TimesTen, or to connect to a pre-7.0 client application, you should use **ttMigrate** to convert your data store to a data store character set other than TIMESTEN8. See [“Data store character set conversion” on page 93](#) for details.

Data type conversion

When performing an upgrade from a version prior to TimesTen 7.0, you must choose whether to preserve the data types in your data store as TimesTen data types, or whether to convert them to Oracle data types. Your planned use for the data store and your preferred upgrade method will have an impact on this decision.

Converting data types to Oracle data types

Note: If you intend to use your data store with TimesTen Cache Connect to Oracle, you must convert your data types to Oracle data types. However, you will not be able to perform an online upgrade using replication.

To convert the data types from a release prior to TimesTen 7.0 to Oracle data types, you must use the `-convertTypesToOra` option for **ttMigrate** when you restore your data store as part of your upgrade procedure. For example, if you restore the data store `salesdata` as part of an upgrade procedure, you may use the following to upgrade the data types to Oracle data types:

```
ttMigrate -r -convertTypesToOra salesdata salesdata.mig
```

See “[TimesTen to Oracle data type conversions](#)” on page 301 in the *Oracle TimesTen In-Memory Database API Reference Guide* for more information.

Note: Because the Oracle and TimesTen versions of some data types behave slightly differently, you should thoroughly test any applications written for versions of TimesTen prior to 7.0 with the new Oracle data types before deploying them with TimesTen 7.0.

Upgrading data types as TimesTen data types

Note: If you intend to perform an online upgrade using replication, you must upgrade your data types as TimesTen data types. See “[Online upgrades with replication](#)” on page 96 for more information.

If you choose to upgrade the data types in a data store from a version prior to TimesTen 7.0 as TimesTen data types, you do not need to use any special options when restoring the data store with **ttMigrate**. The data types from a version prior to TimesTen 7.0 will automatically be restored as TimesTen data types.

Note: The default **TypeMode** attribute for data stores in TimesTen 7.0 is 0, which indicates Oracle type mode. In Oracle type mode, a data type such as CHAR has the semantics of the Oracle CHAR data type rather than the TimesTen CHAR data type. In order to guarantee compatibility with applications written for TimesTen versions before 7.0, you should configure the DSN for your data store with a **TypeMode** of 1 before restoring the data store with **ttMigrate** as part of the upgrade procedure.

Data store character set conversion

Beginning with TimesTen 7.0, a character set must be specified for each TimesTen data store using the DSN attribute **DatabaseCharacterSet**. In some cases, you may need to change the configured data store character set as part of the upgrade process. There are two different cases in which a data store character set conversion will be required:

- You have specified the data store character set as **TIMESTEN8** in order to upgrade your data store from a version of TimesTen prior to 7.0 using online upgrade with replication and/or client/server. After the upgrade is complete for all data stores and client applications, you should convert each data store from this special transitional character set to the national character set you prefer to use for your region. See [“Converting from the TIMESTEN8 character set” on page 93](#).
- You need to change your data store’s character set from the one that you originally specified to a new one that fits your requirements more closely. See [“Converting from a character set other than TIMESTEN8” on page 94](#).

Converting from the TIMESTEN8 character set

You may use **ttMigrate** to convert a data store from **TIMESTEN8** to any other character set by completing the following steps:

1. Save the data store to a file using **ttMigrate**. For example, to save the data store `salesdata` to the file `salesdata.mig`, use the command:

```
ttMigrate -c DSN=salesdata salesdata.mig
```
2. Destroy the data store:

```
ttDestroy salesdata
```
3. Change the value of the DSN attribute **DatabaseCharacterSet** for your data store to the value specifying the new character set. For example, if you want your data store to use the **WE8ISO8859P1** character set instead of **TIMESTEN8**, use the following line in your ODBCINI file:

```
DatabaseCharacterSet=WE8ISO8859P1
```

4. Load the data store from the file using **ttMigrate** with the `-noCharsetConversion` command line option. This option ensures that no character values are changed when the data is loaded into the DSN using the new character set. For example:

```
ttMigrate -r -noCharsetConversion
        DSN=salesdata salesdata.mig
```

Note: If you find that you have accidentally converted your data store from `TIMESTEN8` to the wrong character set, you can use the same procedure to convert your data store to the correct character set without any accidental modification of the character data.

Converting from a character set other than `TIMESTEN8`

You may use **ttMigrate** to convert a data store from any character set to any other character set by completing the following steps:

1. Save the data store to a file using **ttMigrate**. For example, to save the data store `salesdata` to the file `salesdata.mig`, use the command:

```
ttMigrate -c DSN=salesdata salesdata.mig
```
2. Destroy the data store:

```
ttDestroy salesdata
```
3. Change the value of the DSN attribute **DatabaseCharacterSet** for your data store to the value specifying the new character set. For example, if you want your data store to use the `WE8ISO8859P1` character set, use the following line in your `ODBCINI` file:

```
DatabaseCharacterSet=WE8ISO8859P1
```
4. Load the data store from the file using **ttMigrate**. TimesTen will automatically convert the character data from the character set the file was saved with to the character set used by the DSN. For example:

```
ttMigrate -r DSN=salesdata salesdata.mig
```

Note: It is possible that character data will be lost in the conversion process if no mapping exists from one character set to the other for a given character.

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, such as moving from the first patch release of 7.0, version 7.0.1.0.0, to the second patch release of 7.0, version 7.0.2.0.0. 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" and "ttBackup" in *Oracle TimesTen In-Memory Database API Reference Guide*.)

Online upgrades with replication

When upgrading to a new major release of TimesTen, you may have a mission critical data store that needs to remain continuously available to your applications. You can use TimesTen replication to keep two copies of a data store synchronized, even when the data stores are from different versions of TimesTen, allowing your applications to stay connected to one copy of the data store while the other one is being upgraded. When the upgrade is finished, any updates that have been made on the active data store will be transmitted immediately to the upgraded data store, and your applications can then be switched to the upgraded data store with no data loss and no down time. For more information, see [“Performing an online upgrade with replication” on page 106](#).

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. Also, because two copies of the data store to be upgraded are required, you must have twice the memory and disk space that the data store usually requires available, if performing the upgrade on a single system.

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.

Online upgrades with Client/Server

If you are upgrading a TimesTen Client/Server installation to a new major release, you can minimize downtime by performing a client/server online upgrade. During this process, TimesTen clients from the previous version are able to continue to communicate with a data store that has been upgraded to the new version. There are two ways to do this:

- If you are upgrading from a version of TimesTen before 6.0, you can choose to leave the old version of TimesTen Server installed when you install the new version. The old version of TimesTen Server may then be configured to pass connection requests from the old version of TimesTen Client to the new version of TimesTen Server once the data store has been upgraded. For more information, see [“Performing](#)

[a Client/Server online upgrade from a TimesTen version prior to 6.0” on page 115.](#)

- If you are upgrading from a version of TimesTen that is 6.0 or above, the process is much easier. Starting with 7.0, TimesTen Server is able to talk directly to all versions of TimesTen Client from 6.0 onward. There is no need to keep the old installation of TimesTen Server when upgrading to the new version in this case. See [“Performing a Client/Server online upgrade from TimesTen version 6.0 and above” on page 118.](#)

The simple Client/Server online upgrade process minimizes, but does not eliminate, the interruption of client application access to the data store being upgraded. To maintain nearly continuous availability of a data store to all clients, you may use the approach outlined in [“Online upgrades with replication” on page 96](#) to keep an identical copy of the data store available to the old version of TimesTen Server while you upgrade the first copy to the new version. Once the upgraded copy of the data store is available to the new version of TimesTen Server, you may stop the old version and start the new version, listening on the same port. The only interruption in availability using this method will be the very brief period during which the old server is stopped and the new server is started.

On Windows, more than one version of TimesTen cannot be installed at the same time. Therefore, it is not possible to perform online upgrades with Client/Server on Windows if you are upgrading from a version of TimesTen prior to 6.0. It is also not possible to combine the Client/Server online upgrade procedure with that for performing an online upgrade with replication unless you use two different machines, one for each version of the data store that you are upgrading.

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 in shared memory as long as any applications or TimesTen agents (such as the cache or replication agents) are connected to them. Data stores may also be kept in shared memory, even when no applications or agents are connected, if the RAM policy of the data store has been modified using the **ttAdmin**

utility. (See "ttAdmin" in *Oracle TimesTen In-Memory Database API Reference Guide*.) In order to unload a data store:

1. Disconnect all applications from the data store.
2. In this example, the data store `original` is from the previous release. The data store `upgrade` is in the new release. If replication has been started, pause replication on the data stores, and then stop replication on the data store that you are unloading from memory, using the commands:

```
ttRepAdmin -receiver -name upgrade -state pause original
ttRepAdmin -receiver -name original -state pause upgrade
ttAdmin -repStop upgrade
```

3. If the cache agent has been started for the data store, stop the cache agent with the command:

```
ttAdmin -cacheStop upgrade
```
4. 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 upgrade
```

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 0 or wait for the grace period to elapse.

5. Use the `ttStatus` utility to verify that the data store has been unloaded from memory. (See "ttStatus" in *Oracle TimesTen In-Memory Database API 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 7.0.10.0.0 to 7.0.20.0.0, 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 97.

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 be disconnected 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 with replication](#)” on page 106.

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 also rename the table owners in every other data store involved in the replication scheme.

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 **ttBackup** utility may be used if you wish to move the data store to a different machine or directory. The **ttMigrate** utility *must* be used if you wish to:

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

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 97.)
- Use either **ttMigrate** with the `-c` and `-noRepUpgrade` options or **ttBackup** to back up the data store.
- Install the new release of TimesTen. (See [Chapter 2](#), “[TimesTen Installation](#).”)
- Use either **ttMigrate** with the `-r` and `-noRepUpgrade` options 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 by using **ttDestroy** to remove 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/NewSalesData`.
4. `ttRestore -dir /tmp/dump -fname salesdata "DSN=NewSalesData"`
(Verify that `NewSalesData` is operational.)
5. `rm -r /tmp/dump`
6. `ttDestroy /old/SalesData/sales/SalesData`
7. Remove the DSN definition for the `SalesData` data store.

Note: You must reconfigure replication if the data store has been configured to replicate.

Moving to a different machine

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.

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.

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/Sales/Data` with data store name `salesdata` on the source system to `/data/Sales/View` with data store name `salesview` on the destination system, 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.

Step	On the source system	On the destination system
1.	<code>ttBackup -o "DSN=SalesData" > /tmp/salesbackup</code>	
2.	<code>ftp /tmp/salesbackup</code> to the destination system as <code>/tmp/salesbackup</code> Note: Use the <code>ftp</code> command in binary mode.	
3.		<code>ttRestore -i "DSN=SalesView" < /tmp/salesbackup</code>
4.		<code>rm /tmp/salesbackup</code>
5.	<code>rm /tmp/salesbackup</code>	

Reducing data store size

Once a data store has been defined with a particular size for the permanent partition (indicated by the **PermSize** DSN attribute), 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's permanent partition size embedded in it.

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

Note: The permanent partition size of a data store cannot be reduced below the size that is actually required by the data currently stored in the data store. This value can be determined by querying the `perm_in_use_size` column of the table `sys.monitor`.

Perform these steps to reduce the permanent partition size of a data store:

1. Back up the old data store with **ttMigrate** `-c -noRepUpgrade`.
2. Create a new DSN definition for the new copy of the data store with a smaller **PermSize** value.
3. Restore the backup with **ttMigrate** `-r -noRepUpgrade`.

Note: If you wish to use the original DSN rather than create a new one in Step 2, then you must first destroy the original data store files using the **ttDestroy** utility.

Here are the steps to reduce a data store's allocated size from 400 MB to 100 MB. The data store is in `/ds/Sales/Data` and has the Data Source Name (DSN) `salesdata`.

1. `ttMigrate -c -noRepUpgrade DSN=salesdata /tmp/salesbackup`
2. `ttDestroy /ds/Sales/Data`
3. Update the DSN `salesdata` to have a size of 100 MB. See "[Changing data store size](#)" in the *Oracle TimesTen In-Memory Database Operations Guide*.
4. `ttMigrate -r -noRepUpgrade "DSN=salesdata;AutoCreate=1" /tmp/salesbackup`

Note: The temporary partition size of a data store may be changed by simply modifying the **TempSize** attribute of the DSN, unloading the data store from memory, and then reconnecting to it. See [“Unloading a data store” on page 97](#) for the steps to unload a data store from memory.

Moving 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 (DSN) for the 64-bit data store. See [“Creating a DSN on Windows” in *Oracle TimesTen In-Memory Database Operations 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 a 32-bit instance of TimesTen is installed in `/opt/TimesTen/giraffe32` and a 64-bit instance is installed in `/opt/TimesTen/giraffe64`, the required steps are:

1.

```
/opt/TimesTen/giraffe32/bin/ttMigrate
  -c -noRepUpgrade DSN=salesdata32 /tmp/salesbackup
```
2.

```
/opt/TimesTen/giraffe64/bin/ttMigrate -r
  -noRepUpgrade "DSN=salesdata64;AutoCreate=1"
  /tmp/salesbackup
```

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

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 applications of a different major release. To migrate data between TimesTen major releases, for example from TimesTen 5.1 to 7.0, 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. The procedure for this upgrade is similar to

the steps outlined in [“Moving between 32-bit and 64-bit data stores”](#) on page 103.

If you need to upgrade two or more data stores that are replicating to each other, you must perform a few extra steps in order to ensure that replication will continue to operate during and after the upgrade. For example, to migrate two replicating data stores, `master1` on host machine `masterhost` and `subscriber1` on host machine `subscriberhost`, from TimesTen release 5.1 to TimesTen release 7.0, perform the following steps:

1. Configure the replication scheme on both data stores to use static TCP/IP ports for replication. This is necessary because the intermediate result of these steps is that replication occurs between two different versions of TimesTen, and each version will not necessarily know how to find the main daemon of the other in order to have the replication ports assigned dynamically. See [“Dynamic vs. static port assignments”](#) in the *TimesTen to TimesTen Replication Guide* for more information.

2. On the machine `masterhost`, use the 5.1 release’s **ttAdmin** utility to stop the replication daemon on the data store:

```
ttAdmin -repStop master1
```

3. Next, use the 5.1 release’s **ttMigrate** utility with the `-c` option to back up data store `master1` to a binary file:

```
ttMigrate -c DSN=master1 master1.bak
```

4. Use the 5.1 release’s **ttDestroy** utility to destroy data store `master1`, where the data store’s files are located in the directory `data_store_path`:

```
ttDestroy /data_store_path/master1
```

5. Use the 7.0 release’s **ttMigrate** utility with the `-r` option to restore data store `master1` from the binary file. Restoring the data store automatically upgrades it from release 5.1 to release 7.0. If you are restoring a very large data store, you should use the `-C` option to tell **ttMigrate** to perform a checkpoint operation on the data store periodically. This saves time if the restore fails at some point before the operation has been completed. For more information, see [“Checkpoints”](#) in the *Oracle TimesTen In-Memory Database Operations Guide*.

```
ttMigrate -r -C 20 DSN=master1 master1.bak
```

6. Use the 7.0 release’s **ttAdmin** utility to start the replication daemon:

```
ttAdmin -repStart master1
```

Replication is now occurring between the data store `master1` on release 7.0 and the data store `subscriber1` on release 5.1.

Now upgrade data store `subscriber1` from release 5.1 to release 7.0.
Perform the following steps:

1. On the machine `subscriberhost`, use the 5.1 release's **ttAdmin** utility to stop the replication daemon:

```
ttAdmin -repStop subscriber1
```

2. Use the 5.1 release's **ttDestroy** utility to destroy data store `subscriber1`, where the data store's files are located in the directory `data_store_path`:

```
ttDestroy data_store_path/subscriber1
```

3. Use the 7.0 release's **ttRepAdmin** utility with the `-duplicate` option to duplicate data store `subscriber1` from data store `master1` via replication.

```
ttRepAdmin -duplicate -from master1 -host masterhost  
subscriber1
```

4. Use the 7.0 release's **ttAdmin** utility to start the replication daemon:

```
ttAdmin -repStart subscriber1
```

The data stores are now upgraded and replicating to each other.

Performing an online upgrade with replication

In “[Performing an offline upgrade](#)” on page 99 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 may instead perform an in-place or offline upgrade.

Normally, applications that require high availability of their data use TimesTen replication to keep at least one extra copy of their data stores up to date. An online upgrade works by keeping one these two copies available to the application while the other is being upgraded and is unavailable to the application. The 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 the 5.1 version of TimesTen and a data store created in the 7.0 version of TimesTen is not supported. However, if one data store was created in the 5.1 version, and the peer data store was created in the 5.1 version and then upgraded to the 7.0 version, replication between them is supported.

Note: For security reasons, replication is not normally allowed between TimesTen 7.0 and previous versions. In order to perform an online upgrade with replication, you must start the main TimesTen 7.0 daemon with the `-insecure-backwards-compat` option. See “[Communicating with older versions of TimesTen](#)” on page 71 of the [Oracle TimesTen In-Memory Database Operations Guide](#) for more information.

Overview

Upgrading a data store from an older TimesTen release to a newer release is performed by disconnecting all applications from one of two replicated copies of a data store, making a backup of the data store with **ttMigrate** from the older release, loading the backup into a newer

release data store using **ttMigrate** from the newer release, and then reconnecting all applications to the upgraded data store.

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

The general steps in an online upgrade include:

- Disconnect all applications from the data store to be upgraded.
- Shut down replication on the system being upgraded.
- Back up the data store residing on the system being upgraded using the older release's **ttMigrate** with the **-c** option.
- Install the newer release of TimesTen on the system being upgraded.
- Restore the replicated data store on the system being upgraded using the newer release's **ttMigrate** with the **-r** option.
- Reconnect all applications to the upgraded data store.
- Restart replication on the upgraded system.

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

To maintain continuous availability, applications will continue to run on one copy of the data store while the upgrade is performed on the disconnected copy of the data store. TimesTen replication retains updates made to the active copy of the data store during the upgrade period and then transfers and applies the updates to the upgraded data store when replication is restarted. When the replicated updates have been completely applied, the applications may be reconnected to the upgraded data store.

This timeline illustrates the steps for performing an online upgrade while replication is running, where the upgrade system is the system on which the data store upgrade is being performed, and the active system is the system containing the data store that the application will remain connected to:

Step	Upgrade System	Active System
1.	Configure replication to replicate to the active system using static ports.	Configure replication to replicate to the upgrade system using static ports.

2.		Connect all applications to the active data store, if they are not already connected.
3.	Disconnect all applications from the upgrade data store.	
4.		Set replication to the upgrade system to the <code>pause</code> state.
5.	Wait for updates to propagate to the active system.	
6.	Stop replication.	
7.	Back up the data store with ttMigrate <code>-c</code> .	
8.	Stop the TimesTen daemon for the old release of TimesTen.	
9.	Install the new release of TimesTen.	
10.	Create a Data Source Name for the upgraded data store on the new TimesTen release, using the ODBC Data Source Administrator on Windows or the <code>.odbc.ini</code> file on UNIX.	
11.	Restore the data store from the backup with ttMigrate <code>-r</code> .	
12.	Clear the replication bookmark and logs using ttRepAdmin <code>-receiver -reset</code> and by setting replication to the active system to the <code>stop</code> and then the <code>start</code> state.	
13.	Start replication.	
14.		Set replication to the upgrade system to the <code>start</code> state, ensuring that the accumulated updates will propagate once replication is restarted.
15.		Start replication.

16.		Wait for all of the updates to propagate to the upgrade system.
17.	Reconnect all applications to the upgrade data store.	

After the above procedures have been carried out on the upgrade system, the active 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 with replication, replication must be configured to use static ports. See [“Dynamic vs. static port assignments” on page 55](#) of the *TimesTen to TimesTen Replication Guide*.

If you are performing an online upgrade on a single system where a bidirectional replication configuration does not already exist, you must make sure that enough memory and disk space is available to support two copies of the data store being upgraded. Both the original data store and its copy will be active for the duration of the upgrade. To maintain the performance of your production applications, you may wish to create the copy of the data store on a second system.

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 `sys.monitor` table, using **ttIsql**:

```
Command> 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, using a concrete example.

We'll refer to the two TimesTen systems being upgraded as the *upgrade* system, on which TimesTen will be upgraded along with the data store, and the *active* system, which will remain operational and connected to the application for the duration of the upgrade. After this procedure has been completed, the same steps can followed to upgrade the active system, with the previously upgraded system now instead acting as the active and connected to the application. However, you may prefer to delay conversion of the active system in order to test the upgraded release.

The upgrade system in the example consists of the data store `upgrade` on the server `upgradehost`. The active system consists of the data store `active` on the server `activehost`.

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

Step	Upgrade System	Active System
1.	<p>Use ttIsql to alter the replication scheme <code>repscheme</code>, setting static replication port numbers so that the data stores can talk across releases:</p> <pre>Command> call ttRepStop; Command> ALTER REPLICATION repscheme ALTER STORE upgrade ON upgradehost SET PORT 40000 ALTER STORE active ON activehost SET PORT 40001; Command> call ttRepStart;</pre>	<p>Use ttIsql to alter the replication scheme <code>repscheme</code>, setting static replication port numbers so that the data stores can talk across releases:</p> <pre>Command> call ttRepStop; Command> ALTER REPLICATION repscheme ALTER STORE upgrade ON upgradehost SET PORT 40000 ALTER STORE active ON activehost SET PORT 40001; Command> call ttRepStart;</pre>
2.	<p>Disconnect all production applications connected to the data store. Any workload being run on the upgrade system must start running on the active system instead.</p>	<p>Use the ttRepAdmin utility to pause replication from the data store <code>active</code> to the data store <code>upgrade</code>:</p> <pre>ttRepAdmin -connStr DSN=active -receiver -name upgrade -state pause</pre> <p>This command temporarily stops the replication of updates from the data store <code>active</code> to the data store <code>upgrade</code>, but it retains any updates made to <code>active</code> in the data store log files. The updates made to <code>active</code> during the upgrade procedure are applied later, when <code>upgrade</code> is brought back up. For details on setting the replication state, see "Setting the replication state of subscribers" in Chapter 4 of the <i>TimesTen to TimesTen Replication Guide</i>.</p>

Step	Upgrade System	Active System
3.	<p>Wait for all replication updates to be sent to the data store <code>active</code>. You can verify that all updates have been sent by applying a recognizable update to a table reserved for that purpose on the data store <code>upgrade</code>. When the update appears in the data store <code>active</code>, you know that all previous updates have been sent.</p>	
4.	<p>Stop the replication agent with ttAdmin:</p> <pre>ttAdmin -repStop upgrade</pre> <p>From this point on, no updates will be sent to the data store <code>active</code>.</p>	<p>Stop the replication agent with ttAdmin:</p> <pre>ttAdmin -repStop active</pre> <p>From this point on, no updates will be sent to the data store <code>upgrade</code>.</p> <p>For details on starting and stopping replication agents, see "Starting and stopping the replication agents" in Chapter 3, "Data Store Upgrades of the <i>TimesTen to TimesTen Replication Guide</i>."</p>
5.	<p>Use ttRepAdmin to stop replication from the data store <code>upgrade</code> to the data store <code>active</code>:</p> <pre>ttRepAdmin -connStr DSN=upgrade -receiver -name active -state stop</pre> <p>This step prevents <code>active</code> from accumulating updates to send to <code>upgrade</code> and resets some of the replication bookmarks.</p>	

Step	Upgrade System	Active System
6.	<p>Use ttMigrate to back up the data store upgrade. If the data store is very large, this step could take a significant amount of time. If sufficient disk space is free on the /backup file system, the following ttMigrate command can be used:</p> <pre data-bbox="294 370 618 423">ttMigrate -c DSN=upgrade /backup/upgrade.dat</pre>	
7.	<p>If the ttMigrate command is successful, destroy the data store upgrade.</p> <p>To destroy a permanent data store (Temporary=0), use ttDestroy:</p> <pre data-bbox="294 634 524 659">ttDestroy upgrade</pre> <p>To destroy a temporary data store (Temporary=1), use ttAdmin:</p> <pre data-bbox="294 748 643 773">ttAdmin -ramUnload upgrade</pre>	<p>Restart the replication agent on the data store active:</p> <pre data-bbox="732 526 1135 550">ttAdmin -wait -repStart active</pre>
	<p>Note: You can delay destroying the old data store upgrade until the migration into the new TimesTen version is successful. However, the replication agent on both data stores must remain stopped and active-to-upgrade replication on the data store active must be remain in the pause state.</p>	
8.	<p>Install the new release of TimesTen.</p>	<p>Resume replication from active to upgrade by setting the replication state to start:</p> <pre data-bbox="732 1235 1135 1317">ttRepAdmin -connStr DSN=active -receiver -name upgrade -start start</pre>

Step	Upgrade System	Active System
9.	<p>Use ttMigrate to load the backup created in Step 6 into a new version of the data store upgrade:</p> <pre data-bbox="199 269 585 350">ttMigrate -r "DSN=upgrade;AutoCreate=0" /backup/upgrade.dat</pre> <p>If the data store is temporary (Temporary=1), first use ttAdmin <code>-ramLoad</code>:</p> <pre data-bbox="199 475 521 496">ttAdmin -ramLoad upgrade</pre>	
<hr/> <p>Note: In this step, you must use the ttMigrate utility supplied with the new release of TimesTen to which you are upgrading.</p> <hr/>		
10.	<p>Use ttRepAdmin to set replication to the data store upgrade to the stop state and then the start state:</p> <pre data-bbox="199 852 599 933">ttRepAdmin -connStr DSN=upgrade -receiver -name active -state stop</pre> <pre data-bbox="199 951 307 972">sleep 10</pre> <pre data-bbox="199 990 599 1071">ttRepAdmin -connStr DSN=upgrade -receiver -name active -state start</pre> <pre data-bbox="199 1089 307 1110">sleep 10</pre>	
<hr/> <p>Note: The <code>sleep</code> command is to ensure that each state takes effect, as the state change can take up to 10 seconds depending on the resources and operating system of the machine.</p> <hr/>		

Step	Upgrade System	Active System
11.	Use ttAdmin to start the replication agent on the new data store <code>upgrade</code> and to begin sending updates to the data store <code>active</code> : <code>ttAdmin -repStart upgrade</code>	
12.	Verify that the data store <code>upgrade</code> is receiving updates from <code>active</code> . You can verify that updates have been sent by applying a recognizable update to a table reserved for that purpose in the data store <code>active</code> . When the update appears in <code>upgrade</code> , you know that replication is operational.	If the applications are still running on the data store <code>active</code> , let them continue until the data store <code>upgrade</code> has been successfully migrated and you have verified that the updates are being replicated correctly from the <code>active</code> to <code>upgrade</code> .
13.		Once you are sure that updates are being replicated correctly, you can disconnect all of the applications from the data store <code>active</code> and reconnect them to the data store <code>upgrade</code> . After verifying that the last of the updates from <code>active</code> have been replicated to <code>upgrade</code> , the data store <code>active</code> is ready to be upgraded.
		<hr/> <p>Note: You may choose to delay upgrading <code>active</code> to the new TimesTen release until sufficient testing has been performed with the data store <code>upgrade</code> in the new TimesTen release.</p> <hr/>

Performing a Client/Server online upgrade from a TimesTen version prior to 6.0

Overview

A user application linked with a 5.0 or later release of the TimesTen Client ODBC driver can connect to a Server DSN of a newer major

release of TimesTen. For example, a 5.0 client can connect to a 7.0 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 Server DSN.

Note: For security reasons, client/server communication is not normally allowed between TimesTen 7.0 and previous versions. In order to perform an online upgrade with client/server, you must start the main TimesTen 7.0 daemon with the `-insecure-backwards-compat` option. See [“Communicating with older versions of TimesTen” on page 71](#) of the *Oracle TimesTen In-Memory Database Operations Guide* for more information.

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.

Client/Server online upgrade example

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

For this example, the hostname for the server machine is `my_server`.

TimesTen 5.1 is installed in `install_dir_51` at `my_server`. The port number for the TimesTen daemon is **15100**. The port number that TimesTen Server is listening on is **15102**.

TimesTen 7.0 is installed in `install_dir_70` at `my_server`. The port number for TimesTen daemon is **17000**. The port number that TimesTen Server is listening on is **17002**.

The `sys.odbcc.ini` file at `my_server` contains:

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

[my_server_dsn70]
Driver=install_dir_70/lib/libtten.so
DataStore=/tmp/ds_70
```

The TTCONNECTINI file at the TimesTen 5.1.x client machine contains:

```
[my_server_51]
Network_Address=my_server
TCP_PORT=15102
```

Because TCP_PORT is set to the TimesTen 5.1 Server port number, the 5.1 server dynamically loads the correct driver for the server DSN.

The .odbc.ini file at the TimesTen 5.1 Client machine contains:

```
[client_dsn_51]
Description=Access to my_server_dsn51 on host "server"
TTC_SERVER= my_server_51
TTC_SERVER_DSN= my_server_dsn51

[client_dsn_70]
Description= Access to my_server_dsn70 on host "server"
TTC_SERVER= my_server_50
TTC_SERVER_DSN= my_server_dsn70
DRIVER=install_dir_70/lib/libtten.so
```

Note: The full path to the 7.0 driver must be specified. Otherwise the server loads the default driver.

Step	Client Machine	Server Machine
1.	-	Start the TimesTen daemon for both the 5.1 and 7.0 installations at my_server using the regular start scripts.
2.	At the client machine, using the ttIsqICS utility, connect to the 5.1 server DSN my_server_dsn51: ttIsqICS -connStr "dsn=client_dsn_51"	-
3.	Using the ttIsqICS utility, connect to the 7.0 server DSN my_server_dsn70: ttIsqICS -connStr "dsn=client_dsn_70"	-

Performing a Client/Server online upgrade from TimesTen version 6.0 and above

As of TimesTen 7.0, TimesTen Server is able to talk directly to all user applications linked with TimesTen Client ODBC driver version 6.0 or higher. This makes the procedures for upgrading a TimesTen Client/Server installation much simpler. There are at least two possible ways to upgrade a TimesTen Client/Server installation, depending on your requirements for client access to the data store:

- If you do not need the data store being upgraded to be available continuously to your client applications, you may simply stop the old server, perform the migration of the data store using [ttMigrate](#), and then start the new version of the server, which should be configured to listen on the same server port.
- If it is critical that the data store be continuously available to the client applications, you may use the procedure outlined in [“Performing an online upgrade with replication” on page 106](#) to keep a second copy of the data store available during the migration of the first copy.

Note: For security reasons, client/server communication is not normally allowed between TimesTen 7.0 and previous versions. In order to perform an online upgrade with client/server, you must start the main TimesTen 7.0 daemon with the `-insecure-backwards-compat` option. See [“Communicating with older versions of TimesTen” on page 71](#) of the *Oracle TimesTen In-Memory Database Operations Guide* for more information.

Client/Server online upgrade

To perform an online upgrade of a TimesTen Client/Server system to a new major version with minimal reconfiguration, perform the following steps:

1. Stop the TimesTen Server on the old version of TimesTen. From this point until the TimesTen Server for the new version is started, your client applications will not have access to the data store. Any attempted updates to the data store by the clients will fail, and you should stop user applications if necessary.
2. Install the new version of TimesTen. At install time, configure the server to listen on the same port as the old version of TimesTen.

3. Use **ttMigrate** to migrate the data store from the old version to the new version. See “[Moving between 32-bit and 64-bit data stores](#)” on page 103 for an example of this procedure.
4. Start the new version of TimesTen Server, if you have not already. The client applications now have access to the upgraded data stores.

Note: Since both versions of TimesTen Server have been configured to listen on the same port in this procedure, if you wish to restart the old version you must first configure it to listen on a different port.

Client/Server online upgrade with continuous access to the data store

The migration of a data store using **ttMigrate** can be a long process if your data stores are very large. If it is critical that your client applications have nearly continuous access to the data store during the Client/Server online upgrade procedure, you may incorporate the procedure for performing an online upgrade with replication using the following steps:

1. Install the new version of TimesTen, making sure to configure TimesTen Server to listen on the same port as the old version. You will be asked whether you would like to start the new server, and must answer no.
2. Follow the steps in “[Performing an online upgrade with replication](#)” on page 106 to upgrade one copy of the data store. The client applications will remain connected to the other, non-upgraded copy of the data store.
3. Disconnect all clients from the old version of the data store.
4. Stop the old version of TimesTen Server.
5. Wait for all updates to finish replicating from the old version of the data store to the new version.
6. Start the new version of TimesTen Server. It will begin listening on the same port as the old version, and client applications may now connect to the new version of the data store with no change to their configurations.

Record of Upgrades

A new set of tables with the prefix `ttrep_schema_version-` is created each time that the **ttMigrate -r** command is used. These tables provide a history of the data store’s replication scheme from release to release.

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 with replication after running **ttMigrate**.

Example 3.1 For example, after two migrations, the data store might contain tables which look like this:

```
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

Numerics

64-bit platforms, supported 19

A

Access Control

authentication 13

Cache Connect 8

Client/Server 8

daemon home directory 12

daemon port 13

directories and files 12

GroupRestrict 13

instance home directory 12

introduction 7

limitations 7

password file 12

privileges 13

Replication 8

security 7

TimesTen instance

administrator 9

TimesTen instance users 10

users 13

AIX 19, 21, 23

installation 58

uninstalling 62

B

batch installation, Windows 42

building, demo applications 86

byte count threshold 33

C

CGIROOT Web Server option 81

CLASSPATH environment

variable 78

Client installation 25

compiling, demo applications 85,
86

concurrent connections

maximum for Server 26

D

daemon

on AIX 62

on HP-UX 54

on Linux 66

on Solaris 49, 71

timestend 49, 66, 71

daemon home directory 12

daemon port

Access Control 13

daemon port number

changing on UNIX 40

daemon startup script 49, 54, 62,
66, 71

Data Manager installation 25

Data Manager library

choosing 40

Data Manager Service

on Windows 44

data source connection

testing for Client/Server 44

data store

migrating 81

default installation directory 85

demo applications

building 85, 86

building and running 84

defining DSNs 84

demo programs

running on Windows 85

DOCROOT Web Server option 80

DSNs, defining 84

E

environment variables

CLASSPATH 78

NT 42

ODBCINI 77, 78

ORACLE_HOME 36, 79

SYSTCONNECTINI 78

TMP 36

TMPDIR 36

F

files

.odbc.ini 40

- documentation 86
 - driver, verifying installation on
 - Windows 42
 - enabling large files 28
 - header 75
 - makefiles 85
 - ODBCINI environment
 - variable 77, 78
 - online help 75
- G**
- GroupRestrict
 - Access Control 13
- H**
- help
 - online 75
 - HP-UX 19, 21, 23
 - installation 50
 - supported environment 19
 - uninstalling 55
- I**
- in-place upgrade 95, 97
 - informational messages 73
 - on UNIX 73
 - on Windows 73
 - installation
 - AIX 58
 - batch, Windows 42
 - components 24
 - default directory 85
 - documentation files 86
 - HP-UX 50
 - individual components 24
 - instances 23
 - Linux 63
 - ODBC 75
 - prerequisites 25
 - silent mode, Windows 42
 - Solaris 45, 67
 - Solaris 32-bit 48, 53, 60
 - Solaris 64-bit 48, 53, 60
 - instance administrators group
 - creating 38
 - instance data store 9
 - instance home directory 12
 - instance registry directory 11
 - instance startup and shutdown 8
 - instances
 - installation 23
 - naming 24
 - port numbers 24
- K**
- kernel
 - enabling large files 28
 - required changes 27, 32
 - semms parameter 27
 - shmmax parameter 27
 - shmseg parameter 27
- L**
- LD_LIBRARY_PATH 79
 - LIB environmental variable
 - on NT 42
 - LIBPATH environment variable 79
 - Linux
 - installation 63
 - uninstalling 67
 - Linux, MontaVista Carrier
 - Grade 19, 21
 - Linux, SUSE 19, 20, 22
 - LOG Web Server option 81
- M**
- makefiles 85
 - maximum
 - concurrent Server
 - connections 26
 - Microsoft Windows 19, 20, 22
 - MIME Web Server option 81
 - MontaVista Carrier Grade Linux 19, 21
 - moving data stores
 - between 32-bit and 64-bit
 - releases 103
 - example 101, 103
 - to a different directory 100
 - to a different machine 100
 - to a different major release 103
 - to a new patch release 98
- N**
- names
 - instances 24

- non-root installation
 - limitations 7
 - prerequisites 38

O

- ODBC installation 75
- ODBCINI file
 - environment variable 77, 78
- off-line upgrade 95, 99
- on-line upgrade 96
 - limitations 109
 - requirements 109
- online documentation 86
- online help 75
- Oracle Connect Administrator
 - web server configuration 80
 - working with 72
- ORACLE_HOME environment variable 36, 79

P

- PASSWORD_FILE Web Server option 81
- PATH environmental variable
 - on NT 42
- PERL Web Server option 81
- PERLLIB Web Server option 81
- platforms, supported 19
- port numbers
 - instances 24
- PORT Web Server option 80
- prerequisites. <Emphasis>See requirements
- privileges
 - Access Control 13
- problems
 - installation problems 87
- problems running demo
 - programs 85

R

- reducing data store size 102
- replication
 - across releases 106
- requirements
 - on-line upgrade 109
 - semaphores 32

S

- security
 - Access Control 7
- semaphores 32, 35
- semms 27
- Server
 - installation 24, 25
- setup script
 - on Linux 63
 - Solaris 46, 68
- setuproot 39
- Shared library path environment variable 79
- shared memory objects, enabling on HP-UX and Solaris 33
- shmmax 27
- shmseg 27
- silent mode installation
 - Windows 42
- Solaris 19, 20, 22
 - 32-bit installation 48, 53, 60
 - 64-bit installation 48, 53, 60
 - semaphore requirements 32
 - supported environment 19
 - uninstalling 50, 71
- starting the daemon
 - on AIX 62
 - on Linux 50, 55, 62, 67, 71
- startup
 - automatic 44
 - automatic (NT) 44
 - manual 44
 - manual (NT) 44
- stopping the daemon
 - on AIX 62
 - on Linux 50, 55, 62, 67, 71
- SUSE LINUX 19, 20, 22
- syslog 73
- SYSTTCONNECTINI environment variable 78

T

- TimesTen administrators group 11
- TimesTen daemon
 - on AIX 62
- TimesTen instance administrator 9
- TimesTen instance users
 - external user 10

- internal user 10
- overview 10
- TimesTen registry
 - creating 39
- TimesTen Server
 - on AIX 62
 - on HP-UX 54
 - on Solaris 49, 71
 - on Windows 44
- timestend process 49, 66, 71
- timestend.pid file 49, 54, 62, 66, 70
- TMP environment variable 36
- TMPDIR environment variable 36
- troubleshooting
 - installation problems 87
- Tru64 19, 21, 23
- ttBackup 95, 99, 100
- ttDaemonLog
 - on NT 73
- ttDestroy 102
- ttIsql 109
- ttMigrate 81, 83, 95, 99, 109
- TTREP_SCHEMA_VERSION
 - table 119
- ttRestore 100
- ttVersion utility 24

U

- uninstalling
 - on AIX 62
 - on HP-UX 55
 - on Linux 67
 - on Solaris 50, 71
 - on Windows NT 45
- UNIX 73
- unloading a data store 97
- upgrade modes
 - described 95

- in-place upgrade 95, 97
- off-line upgrade 95, 99
- on-line upgrade 96
- users
 - Access Control 13
- using the Oracle Connect Administrator 72
- utilities
 - ttVersion 24

V

- verifying installation
 - on Windows NT 42, 43

W

- Web Server
 - options 80
- Web server
 - configuration 80
- WEBROOT Web Server option 80
- Windows
 - informational messages 73
- Windows NT
 - supported environment 19
 - uninstalling 45
- working with the daemon
 - on AIX 62
 - on HP-UX 54
 - on Linux 66
 - on Solaris 49, 71
- working with the Data Manager Service
 - on Windows 44
- working with the TimesTen Server
 - on AIX 62
 - on HP-UX 54
 - on Linux 66
 - on Solaris 49, 71
 - on Windows 44