HP Software Service Manager

For supported Windows® and UNIX® operating systems

Software Version: 7.10

Upgrade Guide

Legal Notices

Warranty

The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

The information contained herein is subject to change without notice.

Restricted Rights Legend

Copyright Notices

Trademark Notices

Documentation Updates

This manual's title page contains the following identifying information:

- Software version number, which indicates the software version
- Document release date, which changes each time the document is updated
- Software release date, which indicates the release date of this version of the software

To check for recent updates, or to verify that you are using the most recent edition of a document, go to:

http://h20230.www2.hp.com/selfsolve/manuals

This site requires that you register for an HP Passport and sign in. To register for an HP Passport ID, go to:

http://h20229.www2.hp.com/passport-registration.html

Or click the **New users - please register** link on the HP Passport login page.

You will also receive updated or new editions if you subscribe to the appropriate product support service. Contact your HP sales representative for details.

Support

Visit the HP Software Support web site at:

www.hp.com/go/hpsoftwaresupport

This web site provides contact information and details about the products, services, and support that HP Software offers.

HP Software online support provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valued support customer, you can benefit by using the support web site to:

- Search for knowledge documents of interest

- Submit and track support cases and enhancement requests

- Download software patches

- Manage support contracts

- Look up HP support contacts

- Review information about available services

- Enter into discussions with other software customers

- Research and register for software training

Most of the support areas require that you register as an HP Passport user and sign in. Many also require a support contract. To register for an HP Passport ID, go to:

http://h20229.www2.hp.com/passport-registration.html

To find more information about access levels, go to:

http://h20230.www2.hp.com/new_access_levels.jsp

Contents

1 Upgrade overview

The HP Service Manager Upgrade Utility upgrades the HP Service Manager applications from versions ServiceCenter 6.2 and Service Manager 7.0 to HP Service Manager 7.10 applications. If you are running a ServiceCenter application version prior to 6.2, you should contact HP Customer Support for information about the best upgrade strategy for your version. You can also obtain an *Upgrade Guide* for previous versions of ServiceCenter at:

http://h20230.www2.hp.com/selfsolve/manuals.

You will need an HP Passport to access this site. For additional information, contact HP Customer Support or your HP sales representative.

The purpose of this *Upgrade Guide* is to describe upgrading from ServiceCenter 6.2 and Service Manager 7.0 to 7.10 only.

Topics in this section include:

- Server run-time environment (RTE) upgrade on page 14
- Application upgrade on page 15

Server run-time environment (RTE) upgrade

You can upgrade an existing ServiceCenter or Service Manager system to use an HP Service Manager 7.10 server run-time environment to take advantage of certain run-time environment features or to preserve application customizations from a previous version.

In order to upgrade your applications to version 7.10 you must first perform a server RTE upgrade. The HP Service Manager 7.10 applications depend upon features in the version 7.10 server run-time environment.

• To obtain the latest client, install the HP Service Manager 7.10 client from the installation CD-ROM by following the instructions in the HP Service Manager Installation guide.

• To obtain the latest server, install the HP Service Manager 7.10 server from the installation CD-ROM.

Application upgrade

You can upgrade your existing Service Manager applications to Service Manager 7.10 applications using the upgrade utility and resolving the differences between the two versions. All applications upgrades require you to first upgrade the server RTE and install a new Service Manager 7.10 client.

What are applications?

Applications are the HP Service Manager modules and their related configuration files. For example, Incident Management, Change Management, and Inventory Management are HP Service Manager applications.

Applications reside in the HP Service Manager file system in a series of tables described by the database dictionaries (dbdicts) named format, application, code, and enclapplication. Other database dictionaries, such as formatctrl, validity, and environment describe tables containing supporting data.

Because HP Service Manager enables you to modify existing database dictionaries or add new database dictionaries, the HP Service Manager upgrade utility identifies differences between your existing customized files and new files, and enables you to make decisions about your migration strategy.

Version numbers

When you perform an upgrade you must be aware of three version numbers, the software release version number, the application version number, and the Upgrade Utility version number.

Upgrade Utility version numbers

For example, if the version number is SC62–SM7.1 v1.0 Build 0022 English, it means that the following are true:

- Upgrade Utility upgrades Service Manager version 4 (and later releases) to Service Manager 7.10

- Upgrade Utility version number is 1.0

- Upgrade Utility build number for this version is 0022

- Language supported is English only

When the Upgrade Utility finishes, it displays the date, time, old application version, new application version, and Upgrade Utility version number on the upgrade screen:

```
10/10/08 23:12:46 Upgraded from SC62 to SM71
using SC62-SM71 v1.0 build 0022 E
```

How does customization affect the upgrade process?

Most customers modify their systems to meet their business needs. Because of changes to the functionality and behavior of a customized system, upgrading ServiceCenter or Service Manager is not an automatic process.

Protecting customized files

The Service Manager Upgrade Utility determines whether each ServiceCenter or Service Manager file is identical to the out-of-box file installed originally. If the existing file is unchanged, the Upgrade Utility replaces it with a new version. If the file on your system is different from the out-of-box version, the Upgrade Utility keeps your version and adds a prefix (NEWSM7<name>) to the name of the new version.

Resolving differences between your files and new application files, and integrating them to preserve functionality creates a custom upgrade.

Interdependency

Service Manager has an inherent application interdependency. Files often rely on other files to complete tasks. If the Upgrade Utility upgrades one file and does not upgrade a dependent file, the two may not interact properly. The application upgrade process is designed to avoid that possibility.

In some cases, the interdependent files may work well when only one is upgraded. The testing process may identify interdependent files that will not work together unless you upgrade both of them.

Comparing new files to old files

The Service Manager Upgrade Utility uses digital signatures to determine if an application matches the original Service Manager application. Digital Signature enables an entire object to be identified.

The Service Manager Upgrade Utility includes the digital signatures of every object from all Service Manager releases. It compares the signature of each object in your file system against the entire library of Service Manager signatures from the previous releases. A matching signature indicates that the object in your system is unchanged from the original object and that the utility can upgrade it safely.

If a signature does not match the object in your system, the Upgrade Utility copies the new object to your system, but does not overwrite the existing object. You can examine both versions during the resolution process later.

The Upgrade Utility does not check the data written to non-system files, such as Incident, Contact, or Request records.

Ongoing customization and upgrades

Do not modify your production system during the upgrade process. Changes made to your production system after you create mirror image development and test environments are not captured by the custom upgrade. If you change your production system, you must restart the upgrade process.

2   Phase I: Planning an upgrade

Good planning allows your upgrade to run as smoothly and quickly as possible, and helps you to avoid retracing your steps.

Topics in this section include:

Step 1: Meet the knowledge requirements

You should be an experienced Service Manager system administrator who is familiar with your installation to manage the upgrade. You should know the following:

*   How the existing Service Manager file system operates

*   How the application files function

*   How to compare records

*   How to use Rapid Application Development (RAD) environment

*   How to program in RAD for systems with customized RAD applications

▶   If you are not familiar with RAD and administering a system with customized RAD applications, contact the RAD programmer who maintains your system modifications.

If you do not have the administrative experience necessary to manage the upgrade, you may need assistance from your local application developers and database administrators. You can also contact HP Customer Support.

Service Manager tools

The utilities you will use most during the upgrade process include:

*   Database Manager

*   Database Dictionary

*   Display application

*   Forms Designer

*   System Definition

*   RDBMS: You should know how the Service Manager file system interacts with the RDBMS that stores your data.

Documentation resources

The upgrade process requires access to the following Service Manager documentation.

- Service Manager 7.10 online Help for information about database dictionaries, Database Manager, Forms Designer, Request Management and the Display application. You can access the Help when you install a client or the Help server.

- *HP Service Manager Installation Guide* for client/server installation instructions. HP bundles this guide with the installation media.

- *HP Service Manager Release Notes* for information about Software Change Requests (SCRs), enhancements, and known issues. The release notes are part of the installation materials and also available on the Customer Support Web site.

▶ For an upgrade to earlier versions, review the documentation for that version.

HP Customer Support Web site access

The HP Customer Support Web site has operating system and compatibility information, product documentation, and release notes. If you do not have an account for this Web site, contact HP Customer Support at: **www.hp.com/go/hpsoftwaresupport.**

License file

The license file is part of an AutoPass installation. If you do not have a valid license file, contact HP Customer Support.

Step 2: Meet the software requirements

Before you start your upgrade, make sure that you meet the following Service Manager system requirements:

- Your RDBMS version, operating system, and client/server environment must meet all criteria listed in the Compatibility Matrix for the target version. See the HP Customer Support Web site to review the Compatibility Matrix.

- Your existing Service Manager application release level must be HP ServiceCenter 6.2 or later or HP Service Manager 7.00 or later. If your production system is on a pre-ServiceCenter 6.2 release, contact HP Customer Support for the best upgrade path.

- Your existing ServiceCenter or Service Manager application release must be English only. To upgrade localized versions, use the localized application upgrade.

- The Service Manager server process (sm) must have read-write access to the database, or your database administrator must make all required database changes.

Shared memory

HP recommends at least 96 MB of shared memory for the upgrade to run. However, if you have a large database, you may need to allocate more shared memory to accommodate it during the upgrade process. You can increase shared memory by using the shared memory parameter in the Service Manager initialization file, sm.ini.

Disk space

You must have enough free disk space available to run the Upgrade Utility, build the custom upgrade, and store the upgrade files. If enough disk space is not available, the upgrade will fail.

Running the Upgrade Utility

You must have space for Service Manager files and enough work space to build the upgrade.

Building a custom upgrade

The amount of customization determines the amount of space required to build the custom upgrade. You need:

- 75 MB of free space for light customization.
- 150 MB of free space for moderate customization.
- 200 MB of free space for heavy customization.

Upgrade Utility files

You must have enough free space to accommodate the Upgrade Utility files. These files require 73 MB of free space.

Table 1   List of upgrade utility files

File	Size in KB	Contents
AppUpgVersion.txt	4	Contains Upgrade Utility version and build numbers to help you identify which application upgrade version you have available.
extaccess.unl	4	Used only when upgrading from SC6.2. Will overwrite the dbdict.
kmstopword.unl	84	Changes kmstopwords dbdict for use with UTF-8 characters.
preupg.bin	712	Applications and data needed to make changes to the system before the upgrade.
self_service.unl	8.3	Profile, template, and user role to enable Self Service ticketing
sqlupgrade.unl	56	Applications and data needed to determine new fields that you must add to RDBMS databases.
transfer.bin	2,930	The Service Manager Upgrade Utility and all supporting objects.

Table 1   List of upgrade utility files

File	Size in KB	Contents
unplannedChange.unl	8	Configuration records needed for new unplanned change category.
upgdbdct.dta	784	Temporary dbdicts needed for the SQL Compare process.
upgrade.dta	42,521.6	Upgrade data for all information except dbdicts.
upgrade.inf	23,000	Upgrade definition file (includes a description of what to upgrade and lists acceptable old signatures).
upgrade.mak	16	`signature` file unload (needed on the target machine).
upgrade.str	1,268	Required database dictionary upgrades.
upgrade.ver	1	Version stamp for this upgrade.

The upgrade.dta file has been separated into separate files for each table. For example, there is now a upgradeactivityactions.dta table and an upgradeactivitytype table. The approximate size for all of these tables is 47 MB.

Backup requirements

You must keep backups of your original production environment, as well as the development and test environments. HP recommends that you back up files as you resolve differences and build the custom upgrade. If your development and testing activity is on a single system, frequent backups create checkpoints that you can go back to whenever necessary.

NFS-mounted partitions

Do not install either Service Manager or the Service Manager Upgrade Utility on an NFS-mounted remote partition. This can cause serious performance degradation. The performance of an NFS-mounted partition drops significantly if it reads data in many small pieces instead of one large chunk. Service Manager generates a lot of database read/write activity. An NFS-mounted partition is significantly slower than a local drive when running the Service Manager Upgrade Utility process.

Step 3: Create development and test environments

In addition to your production system, plan to have:

* A development system that mirrors your current production environment. Use the development system to run the Upgrade Utility and build a custom upgrade. This system should not be on the same machine as the production server.

* A test system that mirrors your current production environment. Run the custom upgrade on the test system and verify it there. This system may be on the same machine as the production server.

HP recommends that you export the custom upgrade to a test environment and run acceptance tests on this version before you move to live production. However, some users choose to run acceptance tests in their development environment. When testing is complete, you can export the custom upgrade to your production environment.

Step 4: Perform a system health check

A well-maintained production system is the easiest to upgrade. Before starting the upgrade process, perform all regular maintenance on your production system. If necessary, contact HP Customer Support for recommended actions. Suspend all customization activity on the production system.

Step 5: Develop an upgrade strategy

In standard Service Manager terminology, customization refers to changes to RAD applications; tailoring refers to changes made by using Service Manager tailoring tools, such as Forms Designer and Format Control. Configuration refers to local settings in your environment records, system information record, and so on.

The upgrade process affects different parts of the Service Manager system. Besides upgrading the standard Service Manager applications, an upgrade affects the RDBMS where Service Manager is running as well as any customized files or RAD applications. For more information, see How does customization affect the upgrade process? on page 16.

Tailored systems

A list of tailored files can help you resolve differences quickly between your existing files and new files. You can also use the SQL Compare utility to determine how files differ.

RDBMS-mapped systems

Because Service Manager data files must be mapped to an RDBMS, before beginning the upgrade you must choose one of the following options:

- Allow the Upgrade Utility to modify your RDBMS tables for you. This is highly recommended, unless you are running DB2®.
- Use SQL Compare to update the RDBMS databases before beginning the upgrade process.

The upgrade can affect certain mappings and tables. Contact your database administrator for assistance and to discuss the impact on the RDBMS.

Localized systems

You can upgrade a localized system with the Upgrade Utility. Before you begin to upgrade a localized system, ensure that you have the correct language package available to which you will be upgrading. For more information and instructions on how to do install the language pack, refer to the *HP Service Manager 7.10 Language Pack Installation Guide*. The Upgrade Utility detects the presence of a localized system and runs the upgrade just as it would for an English system. You will have to make any customization and tailoring changes based on the requirements described in this document for your system configuration.

Customized RAD applications

A list of customized RAD applications and the extent of the customization is useful. If it is not available, the RAD programmer who made the changes may be able to supply information. Or, you may need to run a comparison between the existing application and the new version.

Information Technology Service Management (ITSM) Implementation Accelerator (IIA)

In Service Manager 7.10, HP has continued development of the Information Technology Service Management (ITSM) Implementation Accelerator (IIA) to implement Information Technology Infrastructure Library (ITIL) processes. This implementation has significantly changed some of the out-of-box data structures used by ServiceCenter 6.2. As part of your upgrade strategy, you will need to review and update these data structures before using Service Manager 7.10 in order to use the new workflows found in Service Desk, Incident Management, Change Management, Configuration Management, and Problem Management. This means you must tailor your existing data to match the data structures used by these new workflows.

3 Phase II: Preparing for an upgrade

Before you can develop or test a custom upgrade, you must create a mirror image of your current ServiceCenter or Service Manager production environment and prepare the data for the upgrade process.

Topics in this section include:

- Step 1: Meet RDBMS conversion requirements on page 30
- Step 2: Duplicate the production environment on page 41
- Step 3: Install the Service Manager application upgrade utility on page 42

Step 1: Meet RDBMS conversion requirements

Before upgrading a ServiceCenter 6.2 or Service Manager 7.00 system to Service Manager 7.10, you must perform the following RDBMS conversion tasks.

Task 1: Verify the list of SQL system table mappings.

See Verify the list of SQL system table mappings on page 31, and then go to the next task.

Task 2: Verify all data policy records have a SQL base name value.

See Verify that all data policy records have a SQL Base Name value on page 33, and then go to the next task.

Task 3: Remove sqlhints records for Change Management.

Go to the next task.

Task 4: Is your existing ServiceCenter data version 6.2?

Yes. See Increase the length of the topic field for the SYSATTACHMENT table on page 35 then go to the next task.

No. Go to the next task.

Task 5: Do you plan on using Service Level Agreements (SLA) with Problem Management?

Yes. If your current ServiceCenter version is 6.2, then see Increase the length of fields for Problem Management on page 37, and then go to the next task. Otherwise go to the next task.

No. Go to the next task.

Task 6: Is your existing ServiceCenter or Service Manager data on a DB2 RDBMS?

Yes. See Set DB2 application control heap size configuration parameter on page 38, and Enlarge user space for affected tables on page 39, then go to the next task.

No. Go to the next task.

Task 7: Is your existing ServiceCenter or Service Manager data on a case insensitive RDBMS?

Yes. See Upgrading a case insensitive system on page 40, and then go to the next task.

No. Go to the next task.

Convert all existing ServiceCenter data to a supported RDBMS.

See Convert all existing ServiceCenter or Service Manager data to a supported RDBMS on page 40.

After you meet these RDBMS conversion requirements, you are ready to create the development environment you will use to upgrade to Service Manager 7.10. See Step 2: Duplicate the production environment on page 41 to continue.

Verify the list of SQL system table mappings

Verify that the following tables are listed as SQL system tables.

- cascadeupd
- code
- coderevision
- displaycache
- displayoption
- displayscreen
- formatctrl
- ioaction
- ioactionrevision
- knownerrorphase
- link
- linkline
- notification
- notificationrevision
- Process
- rcenv
- rootcausephase
- schedule
- scripts

- ScriptLibrary
- ScriptLibraryrevision
- slaprofile
- triggers
- upgraderesults
- wizard

If any of these tables are missing from the list of SQL system tables, add a record for the missing table in Service Manager.

To verify the list of SQL system table mappings:

1 Log in to the existing ServiceCenter or Service Manager system with a administrator account.

2 Type **db** in the client command line, and then press **ENTER**.

   The Database Manager opens.

3 In the Table field, type **sqlsystemtables** and click **Search**.

   The sqlsystemtables form opens.

4 In the Dbdict Name field, type the name of the table you want to map as a SQL system table.

5 Select the **Map as Blob?** check box.

6 Click **Add**.

7 Repeat step 4 through step 6 for each table you want to map as a SQL system table.

   Your system is now ready to convert Service Manager data to an RDBMS.

➤ If you have already converted any of these tables to an RDBMS, import the tables back into P4 on your current ServiceCenter RTE, and then convert the tables back to an RDBMS again.

Verify that all data policy records have a SQL Base Name value

The application upgrade process requires that each data policy record have a SQL Base Name of 13 or less characters length. If any data policy records are missing a SQL Base Name entry, add a SQL Base Name value of 13 characters or less.

⛔ The application upgrade process will fail and be unable to resume if you do not provide valid SQL Base Name values for every data policy record.

To verify that all data policy records have a SQL base name value:

1   Log in to the existing Service Manager system with an administrator account.

2   Type **db** in the client command line, and then press **ENTER**.

    The Database Manager opens.

3   In the Table field, type **datadict** and click **Search**.

    A record list form opens.

4   Double-click **datadict**.

    The Data Policy form opens.

5   From the options menu, click **Advanced Search**.

6   Type the following in the Query field:

    sqlbasename=NULL

7   Click **Execute Search**.

    Service Manager displays a list of data policy records with no value for the SQL Base Name field.

8   Add a SQL Base Name value for each data policy record that does not have an existing value.

    ▶   SQL Base Name values must be 13 characters or less.

9   From the options menu, click **Advanced Search**.

10 Type the following in the Query field:

    lng(sqlbasename)>13

11   Click **Execute Search**.

Service Manager displays a list of data policy records with SQL Base Name values greater than 13 characters long.

12   Change the SQL Base Name value for each data policy record that has an existing value greater than 13 characters long to a value of 13 characters or less.

Increase the length of the topic field for the SYSATTACHMENT table

Update the SQL hints record for the topic field in the SYSATTACHMENT table to accept a larger character limit.

To increase the length of the topic field for the SYSATTACHMENT table:

1   Log in to the existing Service Manager system with an administrator account.

2   Type **db** in the client command line, and then press **ENTER**.

    The Database Manager opens.

3   In the Table field, type **sqlhints**, and click **Search**.

    The SQL Hints form opens.

4   Select your RDBMS vendor type from the Database Type field.

5   Type **SYSATTACHMENT** in the File Name field.

6   Type **topic** in the Field Name field.

7   Type the SQL type value that matches your RDBMS vendor from the list below.

Table 1   SQL Type requirements for the topic field

RDBMS type	SQL type value
db2universal	varchar(60)
oracle	varchar2(60)
sqlserver	varchar(60)

8   Click **Save**.

➤   If you have already converted any of these tables to an RDBMS, import the tables back into P4 on your current ServiceCenter RTE, and then convert the tables back to an RDBMS again.

Increase the length of the str.condition field

Update the SQL hints record for the str.condition field in the notification table to accept a larger character limit or remap the field.

To increase the length of the str.condition field:

1   Log in to the existing Service Manager system with an administrator account.

2   Type **db** in the client command line, and then press **ENTER**.

    The Database Manager opens.

3   In the Table field, type **sqlhints**, and click **Search**.

    The SQL Hints form opens.

4   Select your RDBMS vendor type from the Database Type field.

5   Type **notification** in the File Name field.

6   Type **str.condition** in the Field Name field.

7   Type the SQL type value that matches your RDBMS vendor from the list below.

Table 2   SQL Type requirements for the topic field

RDBMS type	SQL type value
db2universal	varchar(255)
oracle	varchar2(255)
sqlserver	varchar(255)

8   Click **Save**.

Increase the length of fields for Problem Management

If you purchased both Service Level Agreements (SLA) and Problem Management, you must increase the field length of several SLA fields in the sloresponse table to accommodate the long status values that Problem Management uses. You may skip this step if you do not manage Problem Management with SLA. If your current ServiceCenter version is 6.2, you must do this step.

Create or update the SQL hints for the following sloresponse fields to accept a character limit of at least 60 characters.

- initial.state
- final.state

To increase the length of fields for Problem Management:

1 Log in to the existing Service Manager system with an administrator account.

2 Type **db** in the client command line, and then press **ENTER**.

   The Database Manager opens.

3 In the Table field, type **sqlhints**, and click **Search**.

   The SQL Hints form opens.

4 Select your RDBMS vendor type from the Database Type field.

5 Type **sloresponse** in the File Name field.

6 Type **initial.state** in the Field Name field.

7 Type the SQL type value that matches your RDBMS vendor from the list below.

Table 3   SQL Type requirements for the initial.state field

RDBMS type	SQL type value
db2universal	varchar(60)
oracle	varchar2(60)
sqlserver	varchar(60)

8 Click **Add** to create a new record or **Save** to update an existing record.

9 Type **sloresponse** in the File Name field.

10 Type **final.state** in the Field Name field.

11 Type the SQL type value that matches your RDBMS vendor from the list below.

Table 4   SQL Type requirements for the final.state field

RDBMS type	SQL type value
db2universal	varchar(60)
oracle	varchar2(60)
sqlserver	varchar(60)

12 Click **Add** to create a new record or **Save** to update an existing record.

Set DB2 application control heap size configuration parameter

If you are running on a DB2 RDBMS, increase the application control heap size configuration parameter from the default of 128 pages to 256 pages.

See your DB2 documentation for information on setting the application control heap size configuration parameter.

Enlarge user space for affected tables

If you are running on a DB2 RDBMS, increase the user space from the current 4k size for the following tables:

- inbox
- kmquery
- svccatalog

To increase the user space size for these tables, customers can perform the following procedure from the command line before running the upgrade. Each table must be done separately.

Before you begin, note that:

- In this procedure, USER32K represents the name of the customer's larger sized user space. One that has buffer size of larger than 4k.
- If the kmquery table is empty, then the insert step can be skipped for this table.

To enlarge user space for affected tables:

1 From the command line: type the following commands and press Enter after each command:

   a **`create table INBOXM1_TEMP like INBOXM1 in USER32K`**

   b **`commit`**

   c **`insert into INBOXM1_TEMP (select * from INBOXM1)`**

   d **`commit`**

   e **`drop table INBOXM1`**

   f **`commit`**

   g **`rename table INBOXM1_TEMP to INBOXM1`**

   h **`commit`**

2 Repeat sequence for each table, making the applicable command substitutions. Note that the command sequence shown in Step 1 applies to inbox1.

Upgrading a case insensitive system

Previous versions of ServiceCenter were able to run on a case insensitive database systems. If you want to upgrade an existing case insensitive system, you must set up a case insensitive Microsoft SQL Server system to host your data. At this time, Service Manager cannot support case insensitive systems on DB2 or Oracle. See the Microsoft SQL Server documentation for information about setting up a case insensitive database.

After you have set up the SQL server host database, upgrade your server runtime environment and applications normally.

Convert all existing ServiceCenter or Service Manager data to a supported RDBMS

HP Service Manager requires a supported RDBMS on which to store data. Before you can upgrade the server run-time environment (RTE) or applications, you must push all of your existing Service Manager data to an RDBMS supported by Service Manager. See your Service Manager documentation for instructions on converting your existing Service Manager data to a supported RDBMS.

In some RDBMS conversion scenarios you may need to push your data to a RDBMS version supported by your current version of ServiceCenter or Service Manager, and then upgrade the RDBMS to a version supported by Service Manager 7.10.

For example, ServiceCenter 5.1 does not support Microsoft SQL Server 2005, while Service Manager 7.10 does. To upgrade to the supported version of SQL Server 2005, you can install an instance of the SQL Server 2000 client on the ServiceCenter 5.1 system and use it to connect to the SQL Server 2005 database. After you have fully upgraded your ServiceCenter 5.1 data to Service Manager 7.10, you can uninstall the SQL Server 2000 client and install a SQL Server 2005 client.

See the Service Manager compatibility matrices for a listing of supported RDBMS vendors and versions. See your RDBMS vendor documentation for instructions on upgrading RDBMS versions.

Service Manager must have an administrator account on the RDBMS and files must have room for unrestricted growth.

Step 2: Duplicate the production environment

To achieve the best results, develop and test the custom upgrade on a system that resembles your production environment as closely as possible.

To duplicate the production environment

1   Identify a server to use for the development and test environments.

    —   Unix: You can copy the files to a new location on your production machine.

    —   Windows: You must create the development system on a different machine from your production system.

2   Ensure that adequate memory and disk space is available and accessible. Frequent backups are necessary.

3   Ensure that your development and test systems meet all upgrade requirements. For more information, see Step 2: Meet the software requirements on page 22.

    —   Upgrade your RDBMS to a version compatible with HP Service Manager 7.10. See the Service Manager 7.10 compatibility matrix.

    —   Convert your RDBMS code page to Unicode. See your RDBMS vendor documentation.

4   Decide how you want to manage your RDBMS, as described in Step 5: Develop an upgrade strategy on page 26.

5   Set up the environment of your development and test machines to match your production server exactly.

    —   The operating system version and service pack level should match.

6   Copy your existing production system data onto your development system.

    HP recommends you use the native RDBMS back up utilities to back up your data. Refer to your RDBMS documentation for backup instructions.

7   Install a Service Manager 7.10 run time environment on the development system. Do not load the Service Manager 7.10 demonstration data files.

8   Install a Service Manager 7.10 client on the development system.

Step 3: Install the Service Manager application upgrade utility

Follow these steps when you set up your development and test systems as well as when you prepare your production system for an application upgrade.

⚠ Do not try to run the upgrade directly from the DVD. The upgrade needs read-write access to the files.

To install the Service Manager application upgrade utility on Unix or Windows

1 Create the following new subdirectories to the primary directory in the development or test system. They should be on the same level as the RUN directory.

   — Upgrade

   — Backups

   — CustomUpgrade

2 Ensure that the Service Manager server process (sm) has write and execute privileges for these directories.

3 Copy the Service Manager application upgrade utility files to the Upgrade directory.

4 Remove the read-only permissions from the files.

4 Phase III: Upgrading the Server Runtime Environment

You can upgrade the server run-time environment (RTE) separately from the HP Service Manager applications. If you are only upgrading your server RTE, you do not need to create and apply a custom upgrade or do conflict resolution.

Topics in this section include:

Upgrade options

The following table lists the upgrade options available with HP Service Manager 7.10.

Table 1   **Server run-time environment upgrade options**

Previous version	Upgrade options
ServiceCenter 6.2	Upgrade server RTE to 7.10 in preparation for upgrading applications to 7.10.
Service Manager 7.00	Upgrade server RTE to 7.10 in preparation for upgrading applications to 7.10.

▶ HP Service Manager does not support running previous ServiceCenter or Service Manager applications on the Service Manager 7.10 server runtime environment.

Tasks for upgrading your server runtime environment

This section contains a list of actions required to upgrade your server runtime environment (RTE) from to Service Manager 7.10. See the section for your current version of ServiceCenter for a list of required upgrade steps.

ServiceCenter 6.2 to HP Service Manager 7.10

The following tasks are for upgrading Service Manager 6.2 systems to a Service Manager 7.10 RTE. These tasks are valid for the following ServiceCenter and Service Manager starting configurations:

Table 2   Starting configurations

Application version	RTE version
SC6.2	SC6.2
Service Manager 7.00	Service Manager 7.00

Task 1:   Is your existing data fully pushed to a RDBMS?

Yes. Go to Task 2.

No. Go to Task 3.

Task 2:   Is the RDBMS a version supported by Service Manager 7.10?

Yes. Go to Task 4.

No. Go to Task 3.

Task 3:   Push your existing Service Manager data to a RDBMS supported by Service Manager 7.10.

See Step 1: Meet RDBMS conversion requirements on page 30, then go to the next task.

Task 4:   Install a HP Service Manager server run-time environment and client.

See Install a Service Manager 7.10 server and client on page 47, and then go to the next task.

Task 5: Upgrade the Knowledge Management search engine.

See Upgrade the Knowledge Management search engine on page 48, and then go to the next task.

Task 6: Review the list of required configuration file parameter changes.

See Required configuration file parameter changes on page 113.

Task 7: Make a checkpoint backup of the data files to enable you to restore from this point, if necessary.

Use your RDBMS tools to back backup your data files.

Your system is now ready for the application upgrade process. See Phase IV: Running the Upgrade Utility on page 49.

Service Manager 7.00 to HP Service Manager 7.10

Task 1: Review the list of required configuration file parameter changes.

See Required configuration file parameter changes on page 113, and then go to the next task.

Task 2: Make a checkpoint backup of the data files to enable you to restore from this point, if necessary.

Use your RDBMS tools to back backup your data files.

Your system is now ready for the application upgrade process. See Phase IV: Running the Upgrade Utility on page 49.

Install a Service Manager 7.10 server and client

HP recommends first deploying Service Manager 7.10 on a development environment to review new features and ensure system stability.

1   Perform a custom install of the Service Manager 7.10 server on the development system. Do not install the sample database.

    See the *HP Service Manager Installation Guide* for instructions.

2   Install a Service Manager 7.10 client on the development system or an available desktop system.

    See the *HP Service Manager Installation Guide* for instructions.

3   Put a copy of the original data files in the development system.

    See the your RDBMS documentation for restore instructions.

⚠   Do not start the Service Manager server yet.

Upgrade the Knowledge Management search engine

When upgrading the Knowledge Management (KM) search engine from a previous version:

- Ensure that the search engine version matches the Service Manager server version. Do not mis-match versions.

- Uninstall the previous version of the KM search engine and also remove the directories included with the search engine.

- Reindex all of your knowledgebases after you install the new version of the KM search engine.

5 Phase IV: Running the Upgrade Utility

Now that you have a functional environment, you are ready to run the upgrade utility. Follow the steps in this chapter to run the out-of-box upgrade against the data in your development system and to run your custom upgrade against your test and production systems.

> For Oracle users, you must have a granted role that includes "connect" and "resource" along with a granted system privilege of "select any dictionary" as a minimum in order to avoid errors generated by Oracle.

Topics in this section include:

Application upgrade overview
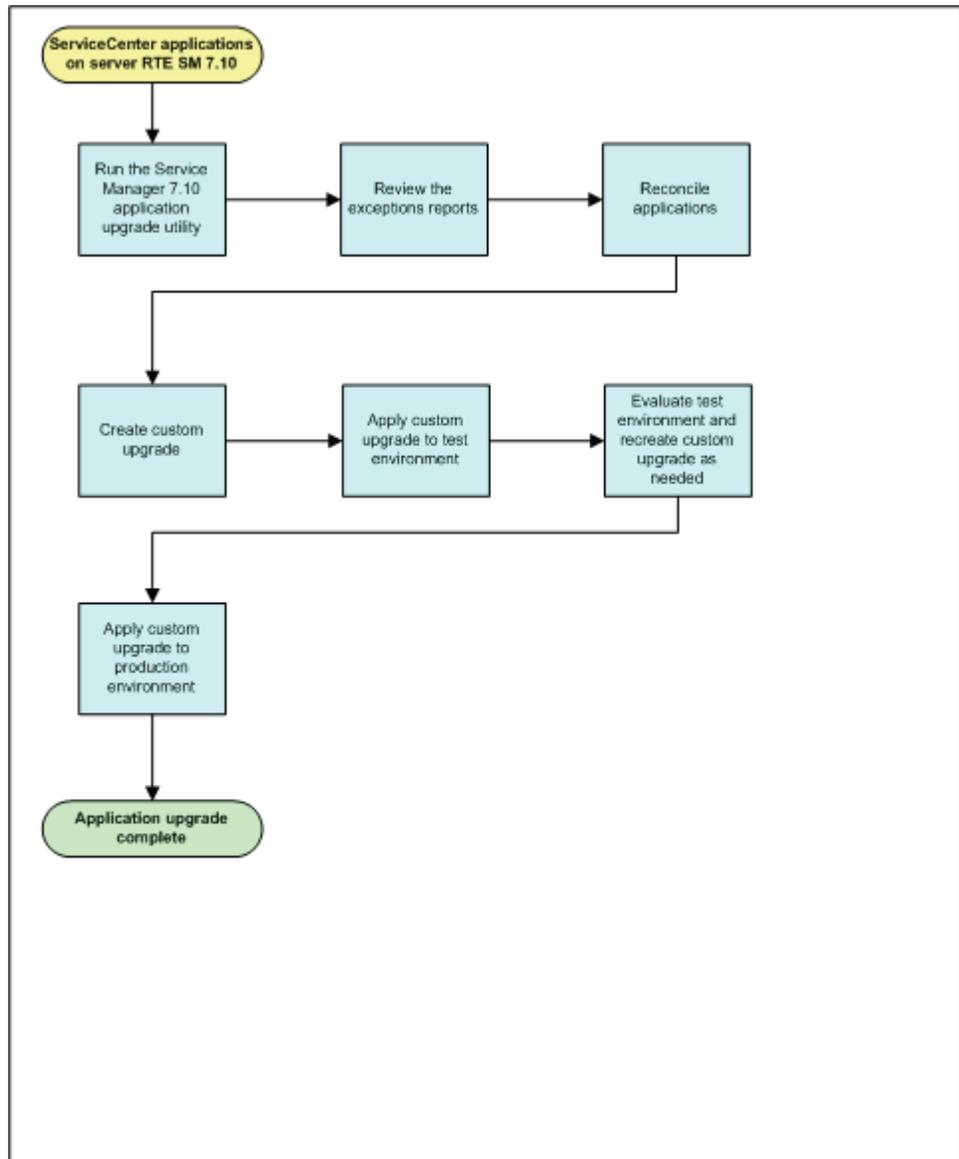
Flowchart:

- **ServiceCenter applications on server RTE SM 7.10**
  - → Run the Service Manager 7.10 application upgrade utility
  - → Review the exceptions reports
  - → Reconcile applications
  - → Create custom upgrade
  - → Apply custom upgrade to test environment
  - → Evaluate test environment and recreate custom upgrade as needed
  - → Apply custom upgrade to production environment
  - → **Application upgrade complete**

The Service Manager upgrade utility upgrades the display components, the database dictionaries, and the application data. It also purges the upgrade files at the end of the process. If the upgrade process encounters any problems while upgrading the components of any of these sections, it stops automatically at the end of the affected section to alert you that it encountered problems.

Each time the upgrade stops, you must exit the upgrade process and fix the identified problems. If the upgrade utility reports any problems with the database dictionary upgrade, resolve all issues before continuing the data phase of the upgrade. Whenever you restart Service Manager upgrade utility, it restarts where it stopped.

These are the main steps in running the upgrade utility. If you are viewing this document online, you can click each step to link to a detailed explanation of that step.

- Step 1: Prepare the Service Manager server on page 52
- Step 2: Run the SQL compare utility on page 57
- Step 3: Run the Upgrade Utility on page 62
- Step 4: Return the system to normal operation on page 68
- Step 5: Review the upgrade reports on page 69

When running an application upgrade:

- If you encounter problems that cause the upgrade process to stop unexpectedly and you cannot determine the cause, contact HP Software Customer Support. Be ready to answer these question for customer support if you call them.
  - What upgrade step was the upgrade was at when it stopped?
  - What messages showed up on the screen?
  - What messages showed up on the `sm.log` file?
  - What messages showed up on the upgrade logs?

Step 1: Prepare the Service Manager server

To run the Service Manager upgrade utility without interruption, you should stop or disable any process that might interrupt the upgrade. You can then load the necessary application upgrade files on the server and start the upgrade process. The following tasks prepare your Service Manager server for an application upgrade.

Task 1: Disable any system screen savers and power saving options.

See Disable screen savers and power saving options on page 52, and then go to the next task.

Task 2: Disable all IR keys on the system.

See Disable IR keys on page 53, and then go to the next task.

Task 3: Purge any existing upgrade files from the system.

See Purge existing upgrade files on page 53, and then go to the next task.

Task 4: Identify the application upgrade files you will need.

See Identify the correct application upgrade files on page 54, and then go to the next task.

Task 5: Load the application upgrade files.

See Load the application upgrade files on page 54.

Disable screen savers and power saving options

Disable any screen savers or power saving options on your Service Manager server before running the upgrade utility to prevent them from interrupting the upgrade process. To improve performance, you should also stop or disable any unnecessary processes on the Service Manager server.

Disable IR keys

The upgrade process runs faster if you disable all IR keys on your existing Service Manager system. You can then enable the IR keys after the application upgrade is complete and you are ready to regenerate the index.

To disable IR keys:

1   Log in to the Service Manager server with an administrator account.

2   Stop the Service Manager server.

3   Open the sm.ini file in a text editor.

4   Add the following line to the end of the file:

     ir_disable:1

5   Save the sm.ini file.

6   Start the Service Manager server.

After the application upgrade succeeds, you can enable the IR keys again by removing the ir_disable:1 entry from the sm.ini file.

Purge existing upgrade files

You must remove upgrade files from any previous versions of Service Manager before running the Service Manager application upgrade.

To purge existing upgrade files:

1   Log into the Service Manager server with an administrator account.

2   Start the Service Manager server.

3   Disable development auditing (revision tracking), if necessary.

4   Type ***aapm.upgrade.purge** on the Service Manager client command line. Press **Enter**.

5   Select **I'm done, and I want to remove the upgrade files completely**.

6   Click **OK** to purge any existing upgrade files.

7   Click **OK** again to exit the Purge Routine Completed dialog box.

Identify the correct application upgrade files

Running the upgrade the first time on your development system is different from exporting the custom upgrade files to your test or production system. The upgrade utility manages both processes, depending on the option you choose and the files that you point to.

- If you are applying the out-of-box upgrade to your development system, use the `preupg.bin` and `transfer.bin` files included with the Service Manager upgrade utility.

- If you are applying a custom upgrade to your test or production system, use the `preupg.bin` and `transfer.bin` files in your custom upgrade directory.

Load the application upgrade files

The out-of-box application upgrade files `preupg.bin` and `transfer.bin` are part of the Service Manager upgrade utility. These files contain the applications and data to run the application upgrade on your development system.

If you are performing a custom upgrade on a test or production system, use your custom upgrade files instead.

To load the application upgrade files:

1   Stop the Service Manager server.

2   Edit the server's `sm.cfg` file and insert a comment character at the beginning of each line except for one **sm -httpPort** statement.

3   Start the Service Manager server.

4   Start the Service Manager client.

5   Log on as a user with SysAdmin permission.

6   Type **db** in the Service Manager client command line, and press **ENTER**.

7   Click **Options > Import/load.**
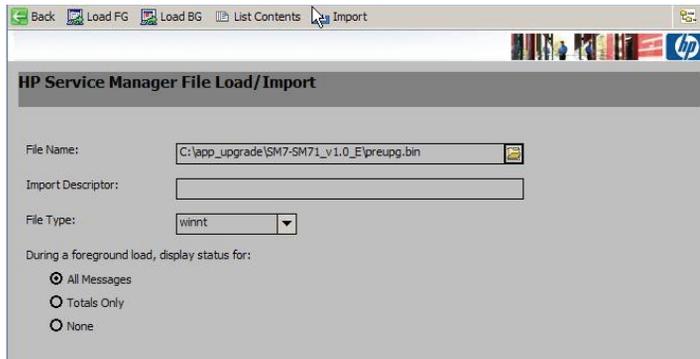
8 Type the fully qualified path to `preupg.bin`, including the file name in the File Name field. For example, if you copied the files to a temporary directory, the path might be:

```
Windows: c:\temp\upgrade\preupg.bin
Unix: /tmp/upgrade/preupg.bin
```

9 Click **Load FG**.

Depending on the speed of your system, loading this file should take from one to five minutes. When the operation is complete, Service Manager returns to the main Database Manager form and displays a confirmation message.

> ▶ The server displays a SQL connection error while loading `preupg.bin`. You can safely ignore this SQL connection error. The `preupg.bin` file searches for the data source name on the system on which it was created, and when it cannot find it displays an error message. After the connection attempt times out, it will use the correct data source name defined in the system `sm.ini` file. You can add the parameter **sqlloginretry:0** to your system's `sm.ini` file to prevent it from attempting to reconnect to the missing data source name.

10 Type **load transfer** in the Service Manager client command line. Press **ENTER**.

11 Type the fully qualified path to `transfer.bin` including the final back slash (\) or forward slash (/), depending on your operating system. For example, if you copied the files to a temporary directory, the path might be:

```
Windows: c:\temp\upgrade\
Unix: /tmp/upgrade/
```

Do not include the file name (`transfer.bin`) in this path.

HP Service Manager Upgrade Utility

What is the fully qualified path to the transfer.bin
file?

`C:/app_upgrade/SM7-SM71_v1.0_E/`

(e.g.   C:\upgrade\)

Press "Next" to continue, or "Back" to exit.

HP Service Manager
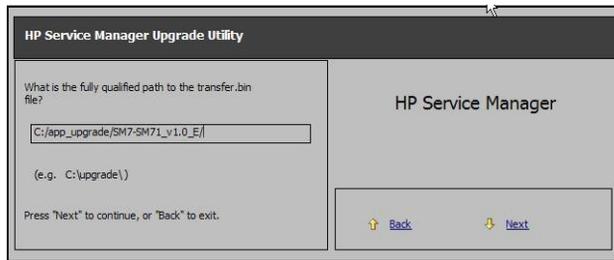
⇧ Back   ⇩ Next

12 Click **Next**. When the process is complete, the system displays a confirmation
    message.

13 Stop the server.

14 Make a checkpoint backup of your data files to enable you to restore from this point,
    if necessary.

15 Restart the server.

Step 2: Run the SQL compare utility

Use the SQL compare utility to see a list of the fields and tables that the upgrade will change. The SQL compare utility compares your existing table and field information with the tables and fields of the Service Manager version you are upgrading to. It will report on any new fields that are to be merged into the existing tables. You can:

- Use the list of the fields produced by SQL Compare to determine whether any fields in your current system differ from those in the new version.

- Use the report to determine which new fields you must add to RDBMS mapped files, if you chose to make the changes manually during the application upgrade.

▶ The application upgrade does not make key changes. You must make any key changes manually. See Key changes on page 140 for the necessary key changes.

Running SQL Compare

The following SQL Compare files install when you install the ServiceCenter Upgrade Utility:

- sqlupgrade.unl

- upgdbdct.dta

SQL Compare returns a message for database dictionary mapping that contains new fields. You can update these database dictionaries to contain the fields specified by the SQL Compare applications before you begin the application upgrade.

▶ Run SQL Compare on the development system.

To run the SQL compare routine:

1   Load `sqlupgrade.unl` into your development system.

2   Type **sm71upgrade** in the ServiceCenter client command line. Press **ENTER**.

3   Click **Run SQL Compare Routine**. A dialog box opens.

4   Type the fully qualified path to upgdbdct.dta including the final back slash (\) or forward slash (/), depending on your operating system. For example, if you copied the files to a temporary directory, the path might be:

```
Windows: c:\temp\upgrade\
Unix: /tmp/upgrade/
```

Do not include the file name (`upgdbdct.dta`) in this path.

SQL Compare returns this message:

```
Process Complete. Please check for any additional messages.
```

The results of the SQL Compare process are stored in the sqlupgrade file. This file resets each time you run SQL Compare.

To view the SQL compare results:

1   Type **sm71upgrade** in the command line. Press **ENTER**.

2   Click **sqlupgrade Records**.

3   Click **Search**. The results display in a record list.

Each database dictionary that requires changes appears as a separate record in the sqlupgrade file. This record also lists the new fields that you must add to the database dictionary, if you are updating your RDBMS mapped system manually.

The sqlupgrade record provides the following information for each field you must add, if you are updating your RDBMS mapped system manually.

Table 1   List of sqlupgrade fields

Field	Description
Field Name	The exact field name to add to the associated P4 database dictionary.
Type	The data type of the field.

Table 1   **List of sqlupgrade fields**

Field	Description
Level	The level where this field resides.
Structure	The structure and array name that you should add to this field to.
Alias of	If this is an alias field, it contains the name of the primary field that it is an alias of. Otherwise this field is blank.

Add new fields

For the new fields to perform correctly, they must exist in the database dictionary and the SQL database. If you are updating your RDBMS mapped system manually, you must add them to the database and update the existing Service Manager SQL mapping. When you update a table in sqlsystemtables, add fields only through the database dictionary. Modifying the SQL mapping damages the file structure of the table.

Determine the correct structure

In most cases, you should add the new field to the descriptor structure. However, sometimes the Structure field contains something other than the word descriptor. When this occurs, add the new field to the appropriate location.

Table 2   Action to take with non descriptor fields

In this instance	Add the field here
The field resides in another structure	If the field is not an array field, you must add the field to the structure listed in the Structure field. For example, if the Structure field reads middle, add the field to the middle structure of the dbdict.
The field is an array	If the field is an array, the field name appears twice in the new field list. The first entry has a data type of array; the second is the data type of the array, such as character or logical. Use the first entry to determine the structure where you should add the array. The Structure field in the second entry reflects both the structure for the array (unless it uses the descriptor structure) and the name of the array itself.
The field is part of an array of structures	If the Structure field lists multiple fields exclusive of an array name, you must add the field to a structured array. To determine the placement in the structured array, follow the list of field names in the Structure/Array from left to right. The first name is the array name and the second is the structure name.

▶ When adding fields to a structured array, add them in the same order as they appear in the sqlupgrade record.

Update field lengths

If you had previously mapped the notification, signatures, upgradepseudolog, or upgradeobjects tables to an RDBMS prior to upgrading to Service Manager 7.10, check the data types of the fields listed below before you begin the application upgrade. The default SQL mapping for these fields did not provide sufficient length for expressions. Change the data type for these columns in your RDBMS to be character-types of at least 255 characters length.

Table 3   List of fields needing a character type at least 255 characters long

Table	Field
signatures	object.name
upgradepseudolog	message
upgradeobjects	object.name
notification	str.condition

Step 3: Run the Upgrade Utility

The Upgrade Utility installs application updates automatically. At any time during this procedure, you can click Next to continue, or click Back to return to previous screens.

To run the ServiceCenter Upgrade Utility:

1 Type **sm71upgrade** in the Service Manager client command line and press **ENTER**.

2 Click **SERVICE PACK** in the Application Upgrade menu.
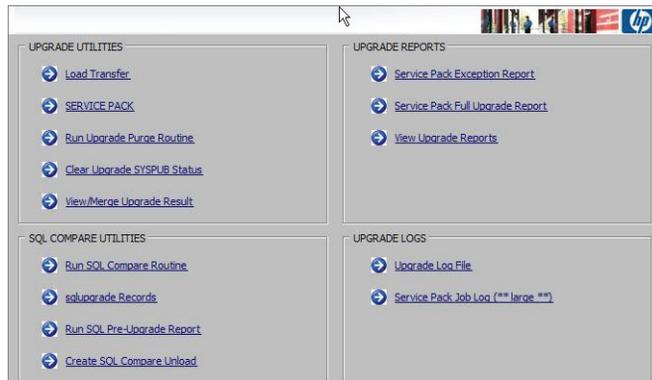
3 Click **Apply an Upgrade**.
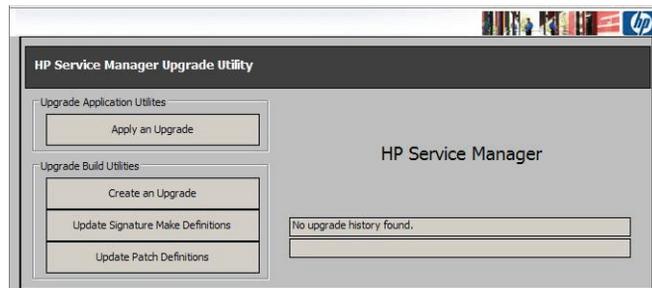
The upgrade utility guides you through questions specific to the upgrade process. You must answer them before the upgrade can begin.

4 Click **Next** to start the upgrade.

5 If this screen displays your correct version number, click **Next**.

If this screen does not display the correct version number, do not continue with the upgrade, instead contact HP Software Customer Support. This could happen because your version is corrupt and the upgrade utility does not recognize it.

6 Specify whether this is a development system. Click **Next**.

— Select **Yes** to apply the out-of-box upgrade to your development system. After this upgrade to your development system, you will perform conflict resolution and create a custom upgrade.

— Select **No** to apply a the custom upgrade to your test or production

— system. You must have already created a custom upgrade.

7 Type the fully qualified path to the upgrade or custom upgrade directory. This path must contain a final forward slash (/) or backslash (/), depending on the operating system.

8 Click **Next**.

9 Select how the upgrade utility should handle objects. Click **Next**.

— If you are applying the out-of-box upgrade to your development system, select **Install HP's Version of the Object Alongside Your Own**. This renames the new file whenever there is a difference between the existing file and the new file. You do not want to overwrite the existing object until you determine what the differences are. The file naming convention is NEW<release><filename>. For example, NEWSM7.1pm.main.

— If you are applying a custom upgrade to your test or production system, select **Replace your version of the object with HP's Version of the object**. This renames the old file if there is a difference between the existing file and the new file. The file naming convention is OLD<release><filename>. For example, OLDSC4pm.main. At this point, the custom upgrade should contain the correct file because you have already resolved the differences.
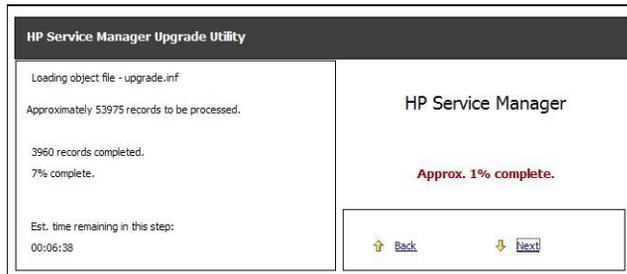
For a more complete explanation of these options, see .

10 The Upgrade Utility always uses External logging. Click **Next**.

 — Click **Yes** to also store the log files internally within Service Manager.

 — Click **No** to use only external logging. External log files reside in the same directory as the upgrade files.

For more information, see Upgrade utility logs and error messages on page 66.

11 Select whether or not to generate To Do records for all existing open records in the major application modules. Click **Next**.

 — Select the **Generate To Do records** check box to generate all open application records.

 — Depending on the number of records in your applications, this step can take a significant amount of time. The record sizes to consider are those in the Change Management, Incident Management, Service Desk, Problem Management, and Request Management applications. If you have a large number of records, you can chose to perform this step at your convenience after completing the upgrade.

 — Do not select the check box if you would like to perform this step later. To run the upgrade for To Do records later, type *aapm.upgrade.todo from the Service Manager client command line.

12 Select the Target SQL Database Type. If you are unfamiliar with the database connected to Service Manager, contact your database administrator.

13 Select the Disposition of Array fields. Click **Next**.

14 Type in the Other RDBMS information. Click **Next**.

15 Choose whether you want the Upgrade Utility to update the RDBMS tables automatically or you want to view and modify the alter statements manually. Click **Next**. For more information, see Modifying the DDL alter statements on page 67.

If you choose to view the alter statements, the upgrade stops at the end of the database dictionary phase, even if you specified the run-to-completion option. You will have the opportunity to modify the alter statements manually.

16 Click **Next** to confirm your selections.

17 A window opens. Click **Yes** to start the upgrade. While the upgrade runs, the Upgrade Utility displays the progress of the upgrade process by indicating which process is running, the progress of record processing, and the completion percentage for the upgrade.

HP Service Manager Upgrade Utility	
Loading object file - upgrade.inf Approximately 53975 records to be processed. 3960 records completed. 7% complete. Est. time remaining in this step: 00:06:38	HP Service Manager **Approx. 1% complete.** ⬆ Back ⬇ Next

During the upgrade process, the Windows Task Manager indicates that Service Manager is not responding. This is normal and does not indicate a problem with the upgrade.

The upgrade utility check points files as they are loaded so that if an error occurs during the data load, you can restart from the last valid table load. To continue, use the log files to correct the error and restart upgrade application. The upgrade utility restarts from where it stopped.

The upgrade utility may display a dialog with the following message after the database dictionary upgrade:

```
Not all dbdicts have been upgraded. Do you want to continue?
```

If this message appears, log off and stop the server. This is necessary to clear the SQL cache. This message indicates that some field types may not have been as the application expected them and could have an impact on the upgrade or regular system operation. Review the upgrade logs for details.

18 When the upgrade process finishes, the upgrade utility displays a completion message. Click **OK** in the message dialog box.

> ▶ If new schedulers are displayed on the status window after the upgrade completes, do not stop the system! Let the schedulers finish the background data upgrade before stopping the system.

19 Review the logs and error messages. For more information, see Upgrade utility logs and error messages on page 66.

20 Test the system, and if it runs satisfactorily purge the upgrade files.

> If you have restarted the upgrade process, do not run the upgrade purge routine again.

Upgrade utility logs and error messages

The upgrade utility creates a set of log files during the upgrade process. These files reside in the same directory as the upgrade files. You can also store a log internally within Service Manager.

Table 4   List of upgrade log files

Log file	Contents
detail.log	Specific information about the upgrade, such as which files are being signed at any time.
exception.log	Information about any exceptions reported by the upgrade.
	The exception.log file may have important messages about data type mismatches that you should resolve, or database dictionaries that it cannot upgrade. If these error messages appear in the except.log file, do not continue the upgrade until you resolve the data type mismatches.
	If there are exceptions in probsummary, problem, cm3r, cm3rpage, cm3t, or cm3tpage files, do not continue the upgrade until you repair them.
upgrade.log	Information about where the upgrade is at any point. This file contains only the main steps of the upgrade.

You can access the post-upgrade reports through the Upgrade menu.

Modifying the DDL alter statements

If you select the option to review the alter statements that the upgrade utility creates, the upgrade stops for each database dictionary that requires an SQL alter statement to be applied.

When the upgrade utility displays an alter statement, you can do the following:

• Allow the upgrade utility to apply the alter statement to the RDBMS exactly as it appears.

• Modify the alter statement manually and allow the upgrade utility to apply that statement.

• Handle the altering of the RDBMS tables manually, and skip the alter statement.

After you view each alter statement and make any changes:

• Click **Proceed** to apply the alter statement.

• Click **Skip** to proceed to the next alter statement without taking any action.

If the alter statement is unsuccessful, the alter window opens. You can edit the statement and try again, or quit the upgrade process.

Step 4: Return the system to normal operation

After the upgrade, the system may exhibit abnormal behavior until you return it to its normal operating environment.

To return to a normal operating environment:

1   Log out.

2   Stop the server.

3   Remove the comment from the system.start entry from the `sm.cfg` file

4   Restart the Service Manager server.

5   Log on.

6   Wait for the background processes to finish.

7   Regenerate your IR keys.

Step 5: Review the upgrade reports

The upgrade utility generates reports that provide feedback about its activities. After the upgrade finishes, you can view them to identify the files on your system that are different from the new files.

This table defines the upgrade reports.

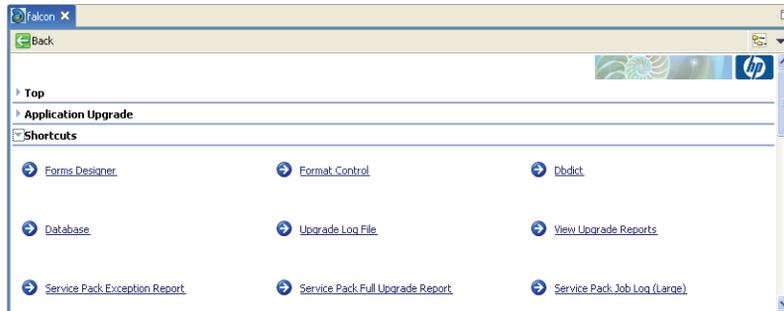**Table 5   List of upgrade reports**

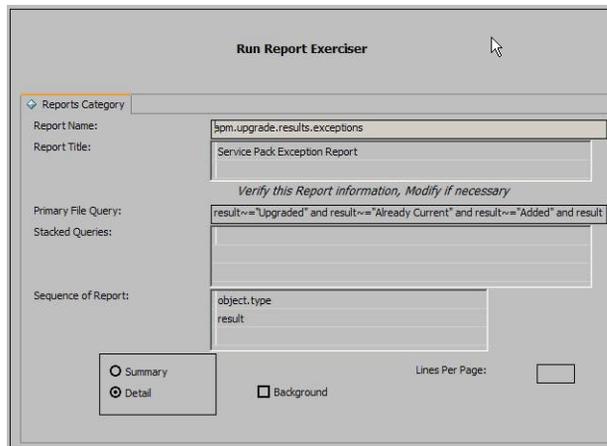| Report Title	Report Name	Description
Service Pack Full Upgrade Report	apm.upgrade.results.full	A list of all objects that the upgrade utility processed, even if they upgraded smoothly. This is a large report that typically contains more than 10,000 lines. The Service Manager parameter maxreportpages default value is 1000. If you do not explicitly set it to a higher number in the sm.ini file, your report terminates after printing 1000 pages.
Service Pack Job Log (Large)	apm.upgrade. job.log	A dump of the upgrade job log. This log is generally not necessary. Use it for debugging or informational purposes.
Service Pack Exception Report	apm.upgrade.results.exceptions	A list of the objects that the upgrade utility could not upgrade automatically. It classifies those components as merged, renamed, or added. If it returns the message: "No records selected by report." then all components upgraded successfully. There are no discrepancies and no records need special attention. If the upgrade utility displays any other messages, they describe upgrade exceptions you must resolve.

View the post-upgrade reports

The post-upgrade reports enable you to identify any conflicts that need to be resolved.

To run a post-upgrade report:

1 Type **sm71upgrade** on the Service Manager client command line. Press **ENTER**.

2 Open the Shortcuts menu and click the report you want to run.

The utility displays the query form for that report.

3 Choose the options for generating the report.

Table 6   List of print options

Option	Definition
Print	If you select the Print option, you can select the printer and schedule when the report prints. If you clear the Print option, Service Manager displays the report. The form only displays this option if you enable server printing.
Background	If you select the Background option, you can select the printer and schedule when the report prints. The report spools in the background, enabling you to return to the client immediately. The scheduler starts the spool process. If you clear the Background option, the report spools in the foreground. The Service Manager client is disabled until the report finishes spooling.

4 Click **Run**.

If you selected Background, The utility displays the report schedule form.
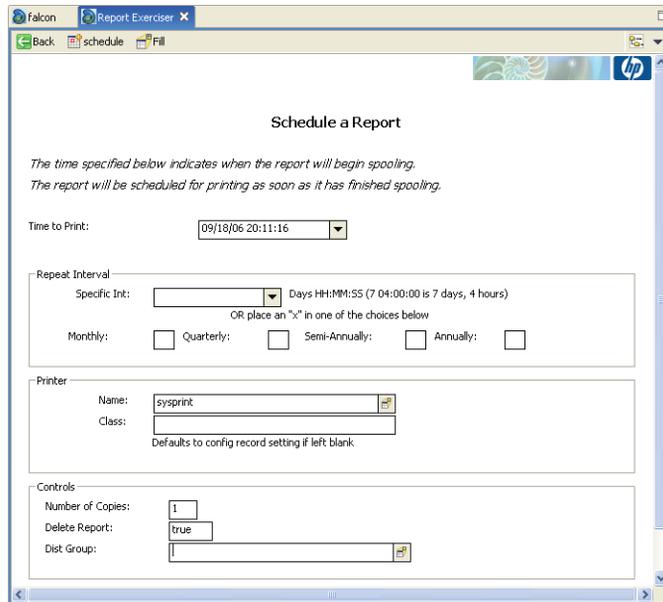
5 Choose the options for scheduling the report.

6   Click **Run**. The utility displays the Report Maintenance form.

7   Click **View** on the Report Maintenance form. The utility displays the spoolheader form.

8   Click **View Report** to open the report.

Retrieve reports

You can run a report once and view it later by selecting it from the print spool.

To retrieve an existing report:

1   Type **spool** in the Service Manager command line. Press **ENTER**.

2   When HP Service Manager displays the Spool Header record, press **ENTER**. (Do not click Find.) Service Manager displays a record list of recent reports in the print spool.

3   Select the report to view.

Step 6: Apply unloads

The unload file, `kmstopword.unl,` should be applied on systems that use double-byte characters for stop words in Knowledge Management. This includes systems that use the Arabic, Russian, and Chinese languages.

Apply `kmstopword.unl` after the Service Manager 7.00 to 7.10 upgrade has been completely finished. Use the Dbdict Utility to apply the unload.

To apply the unload:

1 Click **Menu Navigation->Tailoring->Database Dictionary**.

2 Search for the file, `kmstopword`.

3 Click **Delete Dbdict Record**.

4 Use the File Load/Import utility to load the file.

- Click **Tailoring > Database Manager**.

- Right-click the form and select **Import/Load**.

- Browse to the location of the `kmstopword.unl` file.

- Click **Load FG**.

6 Phase V: Resolving Conflicts

After you complete the initial application upgrade on your development system, you are ready to resolve the differences between your customized files and the new files. You may need to resolve differences in the display components, the database dictionaries, and the applications. Until you resolve these differences, your upgraded system will not work properly.

Topics in this section include:

Resolving conflicts overview

For each file identified in the exception reports, you have these choices:

- Use your custom file instead of the new file.

- Use the new file instead of your customized file.

- Make the same modifications in the new file and discard the old custom file.

When making your decision, consider the following:

- It is not always desirable to overwrite existing files. For example, if you modified an application to add features for your site, you may not want to discard your modifications.

- New application files often have more functions and features that are more attractive than the existing files. Using new files makes your application more robust.

Resources

When you begin the resolution process, assemble your resources and save them as historical data for the next upgrade. The resolution process requires the following resources:

- List of customized files

- Work notes and historical unload archives

- Exception reports

- View/Merge Upgrade Results utility

- Upgrade Results form

- List of files marked as "Reconciled" in the Upgrade Results form

- Display reports

You can use revision tracking to make unload archives for future use.

Make sure to make frequent checkpoint backups of the data files.

Exception report

The exception report lists difference between the new Service Manager signed object types and your existing Service Manager objects. The upgrade utility adds any new objects to your system and does not create an exception report entry. The upgrade utility identifies existing objects in your system where the digital signature has changed. The exception report creates an entry for each changed object in your system. You must decide if the change is significant, or if you can safely accept the difference without detailed investigation of the differences.

Using View/Merge Upgrade Results utility

In the SM7.1 Upgrade Utilities menu there is an option, View/Merge Upgrade Results. This option uses the exception report generated after upgrading a Service Manager (for example an SM7 system to SM7.10) to allow users to view the upgrade results and, in the case of renamed record, merge the data from the renamed records into the active record. The View/Merge Upgrade Results utility displays a side-by-side view of a selected record. The utility allows you to determine which changes you want to include in the new record.

There is also a record keeping capability as part of the View/Merge Upgrade Results utility that enables you to mark each record you update as reconciled. Use the form, Upgrade Results, to mark a record as reconciled, then when you re-run the exception report records that have been marked display the version number of the software in which they were reconciled (for example, SM7.1) in the Reconciled Version column of the exception report. The Upgrade Results form can be used to track all of your changes, including those done manually, without the View/Merge Upgrade results tool.

▶ In the XML records there are instance nodes with record IDs (such as, `recordid="svcCatalog"`). These instance nodes **should not be changed**. Changes to them can break the primary key for a record.

To use View/Merge Upgrade Results

1   Click **View/Merge Upgrade Results** in the Upgrade Utilities menu.
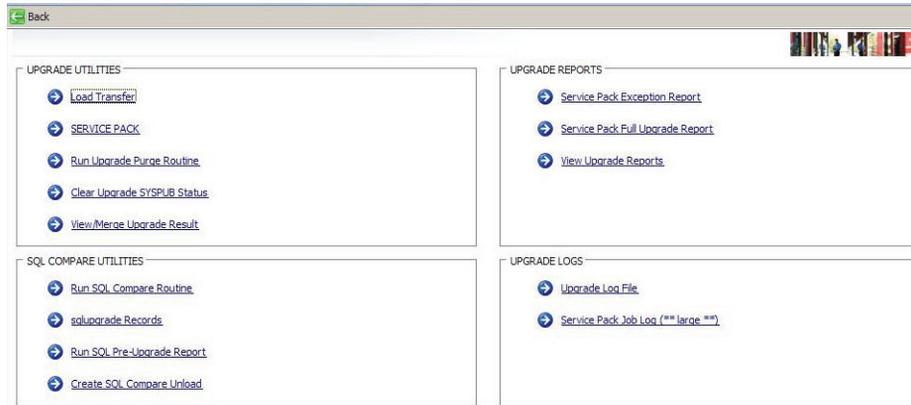
2   Select **Forced** or **Renamed** in the Result text box on the Upgrade Results form.
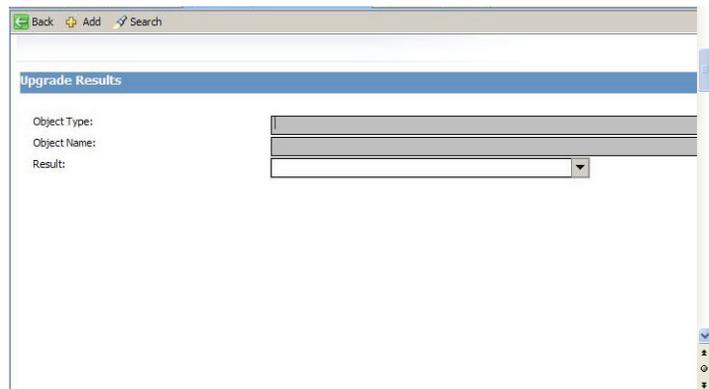
3   Click **Search** to display a list of records with renamed or forced objects. In this
     example, Renamed has been selected.
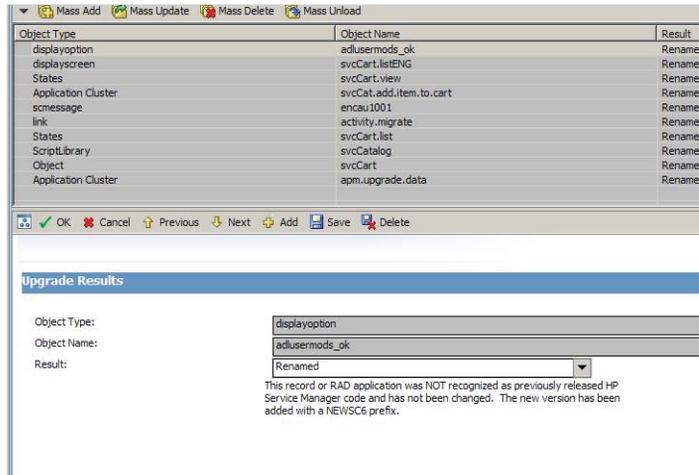
4   Select the record you wish to merge and then click **Merge** in the Options menu list.
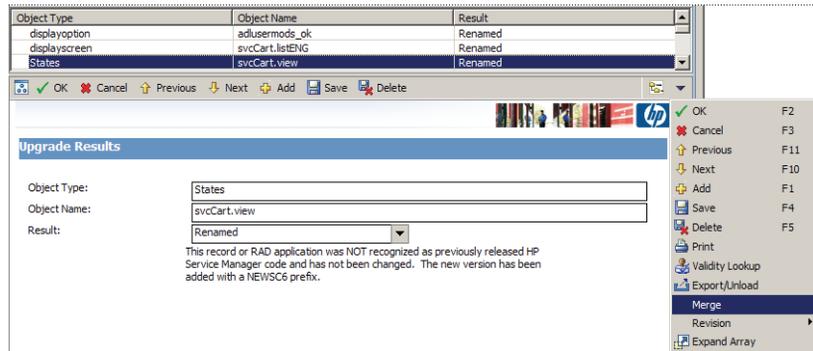
5   In the merge view, select the changes you want included in the new record.
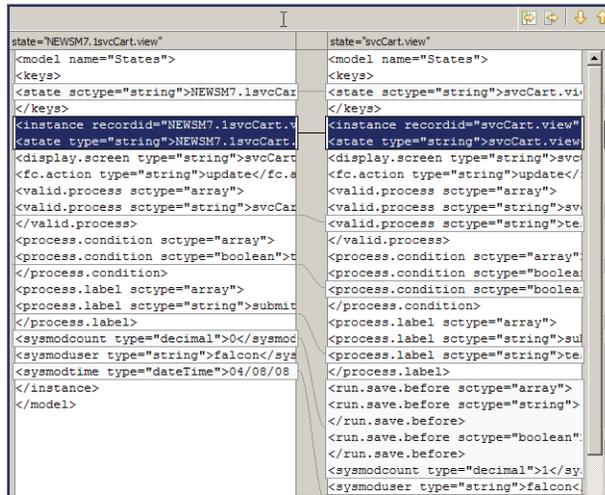
a   Use the buttons at the top of the pane to select individual differences to bring from the new record over to the active record or,

b   You can bring over the new record in its entirety.

6   Click **SAVE**.

7   Continue to examine each record until all records are completed.

Upgrade Results form

The Upgrade Results form enables you to search for changes and display a list of changed objects. You can then select a record in the list and mark the record as changed. You can reconcile a change more than once, but only the most recent reconciliation is retained and recorded. After you have marked a record as reconciled, it appears as reconciled in the exception report along with the version number of the system in which it was reconciled.

To mark a change as reconciled:

Typically, you would use this for ScriptLibrary changes.

1 Click **View/Merge Upgrade Results** in the Upgrade Utilities menu.

2 Select a Result, Renamed for example.
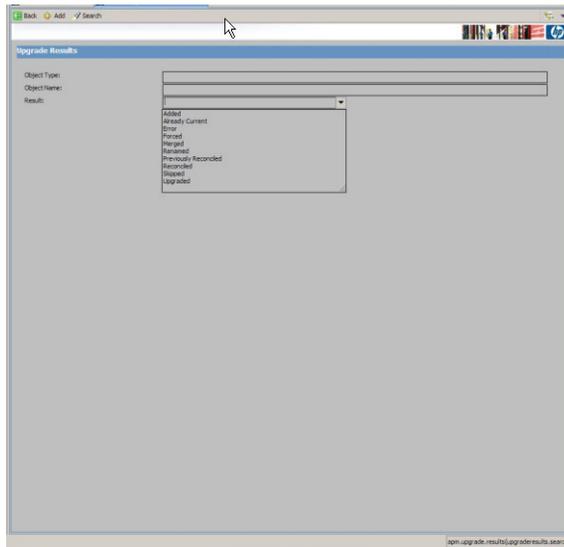
3   Click **Search**.
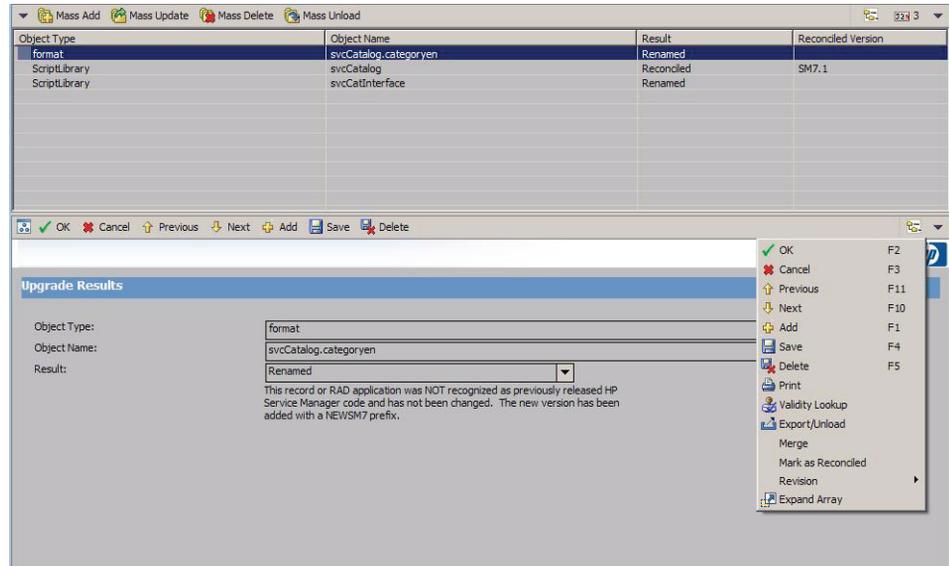
4   Click **Mark as Reconciled** in the list.

5   Continue marking records and searching for changed records until all of changes records are marked.

Step 1: Resolve application data differences

The Upgrade Utility upgrades any object in your file system that is unchanged from the out-of-box version for that release. When the Upgrade Utility encounters an object that is not identical to the original out-of-box version, the upgrade installs the new version as `NEW<vvvnnn>`, where vvv is the release name and nnn is the name of the object, or the upgrade renames your version as `OLD<vvvnnn>`.

The method depends on the choice you made when starting the upgrade process.

- If you chose **Install the HP Version of the Object Alongside Your Own**:

  When the Upgrade Utility does not recognize an object as the original out-of-box version, it leaves that object alone. The upgrade names the new version of the object NEWxxxyyy where xxx is the HP Service Manager Version you are upgrading to and yyy is the name of the object. The new object has the status "Renamed" in the upgraderesults file.

- If you chose **Replace your version of the object with the HP Version**:

  When the Upgrade Utility does not recognize an object as the original out-of-box version, it renames that object as OLDxxxyyy where xxx is the HP Service Manager version you are upgrading from and yyy is the name of the object. The upgrade installs the new version of the object. The new object has the status "Forced" in the upgraderesults file.

You must examine the old object and the new one and decide which of the following options to choose:

- Retain the old version of the object in question
- Accept the new object
- Create a combined object

Most objects, including forms, format control records, and validity records, are relatively easy to check. For example, adding a single user-defined field to a form causes a signature mismatch. Examine the old object and the new one side-by-side, if possible.

Application differences

Customized applications are more complicated. Application differences occur when you customize the RAD code or when you apply a patch to your existing version. The apm.upgrade.results report identifies these application differences. After you produce the report, check each application listed for recommended action. The utility assigns a status to each application. The following table lists the possible status and the recommended action.

Table 1   List of upgrade statuses in the upgrade report

Upgrade status	Definition	Action
Added	The Upgrade Utility added the new object to your system because you did not have an existing object.	No action required.
Error	The Upgrade Utility could not upgrade the object.	Resolve the error. Contact Customer Support for assistance if necessary

Table 1   **List of upgrade statuses in the upgrade report (cont'd)**

Upgrade status	Definition	Action
Forced	The Upgrade Utility renamed your object OLDSM7<object.name> and added the new object as <object.name>. The Upgrade Utility used this option because you chose to replace your version of the object with the HP version of the object. (See Step 3: Run the Upgrade Utility on page 62.)	Select one: **Keep the old version** — No action required. **Keep the new version** — Rename the old version and give the new version the original name. **Merge new and old versions** — Add the new features to the old version. Always make your changes in the old file by manually adding the new features. If you rename the new file to the name of the old file, you will lose data because the data is linked to the logical file number, not the file name.
Renamed	The Upgrade Utility renamed the new object NEWSM7.1<object.name> and added it to your system. The Upgrade Utility used this option because you chose to install the HP version of the object alongside your own. (See Step 3: Run the Upgrade Utility on page 62.)	Select one: **Keep the old version** — No action required. **Keep the new version** — Back up the old version and give the new version the original name. **Merge new and old versions** — Add the new features to the old version. Always make your changes in the old file by manually adding the new features. If you rename the new file to the name of the old file, you will lose data because the data is linked to the logical file number, not the file name.
Reconciled	This status will only appear when the record has been marked as reconciled. (See To mark a change as reconciled: on page 81)	No action required.

Guidelines for resolving differences

Use these guidelines to make conflict resolution decisions:

- If your company does not have a RAD license, always select the new version.

- If your company has a RAD license, identify the differences between the old version and the new version by using the RAD Comparison Utility. Choose one of the following options:

    — Keep the existing version.

    — Keep the new version.

    — Merge the new version with the existing version.

➤ If your company has RAD license and the application has been modified for you specifically, then you should merge changes. A SCR unload can cause applications to show up on the conflict resolution exception report that no longer apply with an upgraded version of the application.

For more information, see Using the RAD comparison utility on page 147.

- If the name of the application starts with apm.upgrade, keep the existing application and delete the NEWvvvnnn version of the application. RAD applications named `apm.upgrade.*` are HP-reserved applications. Do not change them.

- Avoid changing `macro.build.field.array`. If macro.build.field.array is not identical to the new version, keep the new version.

- Never rename the applications `se.view.engine`, `display` or `se.get.object` when resolving conflicts.

- Changes to the **Regen Every** field prevents a global list from upgrading. If comparison to the new version of the global list shows no other changes, keep the new version.

Application dependencies

HP Service Manager applications often use the RAD language to call other applications. For example, the se.view.engine application calls a second application, se.get.object, to update change records. Consequently, the functionality of se.view.engine depends on the functionality of se.get.object.

It is possible to upgrade se.view.engine but not se.get.object. Upgrading only one application can cause parameter differences in a subordinate application and logical dependency differences.

Logical dependences

Logical dependencies usually occur when there is a new feature in a child application that the parent application requires. If you upgrade the parent application but the not the child application, the new feature is not available to the parent application. Resolving logical dependencies usually requires reverting the parent application to the previous version, or upgrading the child application to the latest version.

Parameter differences in subordinate applications

The most common dependency problem arises when the number of parameters in a subordinate application changes. For example, if an application had four parameters but now has five, the old parent application passes four parameters, but the new child application now expects five. If you upgrade the child application but not the parent, the missing parameter can cause errors.

Altered validity table entries

Service Manager processes validity table entries in entry sequential numeric order, such as 1, 2,...n. Most users who implement multiple validity entries add sequence numbers to their entries. Typically, several validity entries exist for a given format with sequence numbers running from 1 to n. Many of the Service Manager default validity entries have sequence numbers of NULL.

When the Upgrade Utility updates your validity entries, it adds Service Manager default validity lookup entries if it does not find an entry with the same key. For example, if your system does not have a validity entry with a sequence number of NULL, and the default validity entry has the NULL sequence number, the Upgrade Utility adds the new entry to your file system.

Although the Upgrade Utility does not remove any of your code, it adds a validity entry that it processes first. As a result, the first validity lookup you see is a default validity entry, rather than one of your modified entries.

To remove invalid validity table entries added by the Upgrade Utility:

1   Type `validity` on the HP Service Manager client command line. Press **ENTER**.

2   Type the name of the file or format in question in the **Files/Formats** field.

3   Click **Search**.

4   Select a field name from the record list.

5   Look for an entry with a sequence number of NULL. Verify that the Upgrade Utility added this entry and that it is not an important part of your validity processing.

6   To remove the invalid validity record, click **Delete**.

Validity processing now starts with your lowest sequence numbered validity entry.

Step 2: Resolve database dictionary differences

When the Upgrade Utility upgrades your database dictionaries, it often finds that your database dictionary version does not match any previous Service Manager versions. For example, this occurs if you add or delete fields.

When the Upgrade Utility discovers these discrepancies, it merges your database dictionary with the new database dictionary. The resulting database dictionary contains all the fields in your original database dictionary, as well as all new fields.

The Upgrade Utility gives these combined database dictionaries a status of Merged. The Merged status does not indicate an error, but shows that the Upgrade Utility found differences in that database dictionary. Database dictionaries mapped to an relational database management system (RDBMS) always have a Merged status because the mapping data within each table is unique and cannot be compared to basic Service Manager data.

Data types

When the Upgrade Utility merges database dictionaries, the data type and the field name must match for success. If the new version of a field is a different type than the type in your system, it chooses your field type. HP recommends that you modify the field to the expected field type.

Duplicate field names are problematic. For example, if you create a scalar field with a user-defined name and the Upgrade Utility attempts to add a structure field with the same name, this discrepancy appears in the except.log file. The Upgrade Utility cannot add the structure and its subordinate fields until you resolve the name duplication. The except.log file lists the names of any fields that are not added because of a duplicate field name.

Create a naming convention with a recognizable unique prefix for all user-defined fields. The Upgrade Utility action will always merge these fields successfully.

Database dictionary differences

After you produce the apm.upgrade.results report, check it for recommended action. The following table lists the possible results and the recommended action.

Table 2   List of upgrade results in the upgrade report

Upgrade result	Definition	Action
Merged	The Upgrade Utility merged the new database dictionary with your current version.	No action required.
Added	The Upgrade Utility added the new database dictionary to your system because you did not have an existing version.	No action required.
Error	The Upgrade Utility could not upgrade the database dictionary.	Resolve the error. Contact Customer Support for assistance if necessary.

Data policy differences

If you modified a data policy record, the Upgrade Utility usually cannot upgrade the associated data policy record. Keep the current version of the database dictionary because it reflects the current state. Investigate the new version of the database dictionary for any changes to data policy that you may want to move to the current database dictionary record.

Step 3: Resolve display component differences

The Service Manager Upgrade Utility upgrades the display components. It reports discrepancies if any of your components differ from the new versions. You must be familiar with the Display application and how it functions to resolve the differences.

Display application

The Display application is a Service Manager RAD application that provides access to RAD features without requiring RAD programming skills or RAD licensing.

If the Display application in your existing Service Manager system is not identical to the new version, rename the current version of the application to `display.old` and rename the `NEWSCxdisplay` version to `display`.

If you access this application through the RAD editor, use the equal sign (=) operator to select only the application in question, such as:
`=display or =NEWSCxdisplay.`

Display screen records

Display screens are individual records identified by a unique screen ID. The displayscreen records define the attributes of a screen and provide access to the individual records for options and events. A display screen is not the same as a form.

⚠️ There are triggers attached to the displayscreen file. Changes to the records in this file affect their associated display options and events.

When making changes to these records, make sure that you do the following:

- Incorporate any changes to the new version of the displayscreen file into the old version of the file.

- Incorporate any changes from the new version of the record (prefixed with `NEWSM7`) into the original displayscreen record.

Duplicate function keys

If you complete the upgrade and find that you have duplicate function keys, you most likely have added display options. Perform conflict resolution on your display options to resolve these conflicts.

Changes to the display component upgrade

Prior to Service Manager 6.2, the Upgrade Utility upgraded the display options and display events differently than other components, such as formats or links. In release Service Manager 6.2 and later, the Upgrade Utility upgrades the display components in the same manner as all other components.

The options table and the events table have a unique identifier, stored in the "ID" field. The upgrade process assigns an ID to every display option following the pattern: `<screen id>_<action>_<number>`, where the screen ID and action are from the display option (or event), and the number is an optional field added when multiple options have the same screen ID and action.

If options that have the same action as others in the same display screen have been added to your system, the upgrade process assigns <number> in the order of the options' GUI option number.

Therefore, if the added option was not the last option in terms of GUI option number, the upgrade process does not add the additional numbers in the ID field in the same order as they would have for an out-of-box system. The upgrade process renames the added option and any option after it (in GUI option order), it does not upgrade them automatically.

To ensure that this type of renaming does not happen in future upgrades, when performing conflict resolution on these options, use the ID of the renamed option, `NEWSM7<screen id>_<action>_<number>` and manually change the identifier of the added options. Rename all of the other options to match the ID of the renamed ones.

When renaming your option, use an identifier to specify that this is a customized option, added for your installation. For an example, an ID might look like: `"apm.edit.problem_do nothing_ACME1"`.

This table gives an example of part of the display screen conflict resolution for apm.edit.problem.

Table 3   **Example confliction resolution for the apm.edit.problem display option**

Screen Id	GUI Action	Upgrade Action	Display Screen option
300	do nothing	update	Name: apm.edit.problem_do_nothing_1 Result: This item was updated correctly. User Action: No action necessary.
400	do nothing	update	Name: apm.edit.problem_do_nothing_2 Result: This item was updated correctly. User Action: No action necessary.
450 This is an option you added.	do nothing	rename	Name: apm.edit.problem_do_nothing_3 Result: This item was renamed. It is your customized option. User Action: Rename this object to give it a unique new name, such as: apm.edit.problem_do_nothing_ACME1 Name: NEWSM7apm.edit.problem_do_nothing_3 Result: This is the new SM7option. User Action: Perform conflict resolution. To perform conflict resolution, open apm.edit.problem and look at the options. Compare this option with apm.edit.problem_do_nothing_3 and NEWSM7apm.edit.problem_do_nothing_3.
500	do nothing	The upgrade ignores this option.	Result: This option does not appear in the reports. User Action: Perform conflict resolution. To perform conflict resolution, open apm.edit.problem and look at the options. Compare this option with apm.edit.problem_do_nothing_3 and NEWSM7apm.edit.problem_do_nothing_3.

Step 4: Make required application changes

The exception report contains many of the application changes you need to make to upgrade your system, but there are certain application changes you need to make to support new features or key changes. Make the changes listed in Application changes required by release on page 121 before testing and backing up your system.

Step 5: Make optional application changes as needed

HP Service Manager includes optional application changes you can make to improve your system performance, usability, or compliance with Information Technology Infrastructure Library (ITIL) conventions. Review the changes listed in Optional application changes on page 135 to implement that changes that benefit your system.

The most system intensive of these optional changes are changes to the system key structure. Service Manager transforms keys to corresponding indexes on your RDBMS. During the application development process, HP Service Manager development may change, add, or delete database dictionary keys. The upgrade does not change any keys automatically because changing keys requires regenerating your index. The amount of time for the regeneration depends on how many records are in the table. You will need to change the keys manually after the upgrade is complete.

Step 6: Test the system

After you resolve all conflicts, test the upgraded system and verify that it functions properly. If there are problems that you cannot resolve, contact HP Customer Support.

Step 7: Back up the system

Make a checkpoint backup of the data files to enable you to restore from this point, if necessary. Refer the documentation for your RDBMS for backup instructions.

7 Phase VI: Performing the custom upgrade

When you begin this phase, you start with an upgraded development system in which any conflicts have been resolved. You are ready to create a custom application upgrade on the development system that you can export to your test or production system.

Topics in this section include:

Custom Upgrade process overview

The custom upgrade consists of three types of files:

- New Service Manager 7.10 application files that replaced old application files

- Customized application files that you retained

- Merged files that combine prior customization with new application functionality

  > The upgrade.dat file has been replaced with a data directory, so you must have a data directory when doing a custom upgrade.

HP recommends that you export to a test environment and run acceptance tests on this version before you move to live production. However, you can run acceptance tests in the same development environment where you ran the upgrade and resolved differences.

The Upgrade Utility uses checkpointing so that if a failure occurs on a table the upgrade restarts at the last good checkpoint. You can also use backups of the Data directory to enable you to retrace your steps if necessary. These are the main steps in performing your custom upgrade. If you are viewing this document online, you can click each step to link to a detailed explanation of that step.

Step 1: Build the custom upgrade

The Service Manager Upgrade Utility automatically exports the custom upgrade for you; however, you must specify a destination with adequate disk space.

To build the custom upgrade:

1 Use the new client to log on to the development system.

2 Use the database dictionary to delete the file scirexpert, if it exists. The upgrade recreates this file automatically.

3 Verify that you have created a CustomUpgrade directory as described in Step 2: Duplicate the production environment on page 41. The Upgrade directory contains the out-of-box upgrade files. You must create the custom upgrade files in a separate directory.

   a Ensure that the CustomUpgrade directory is empty. If you build another custom upgrade, the Upgrade Utility appends to the files existing in the CustomUpgrade directory. It does not overwrite them.

   b Your User ID must have read and write access to the CustomUpgrade directory.

4 Type **sm71upgrade** on the ServiceCenter client command line.

5 Click **SERVICE PACK** in the Upgrade Utilities menu.

6 Click **Create an Upgrade**.
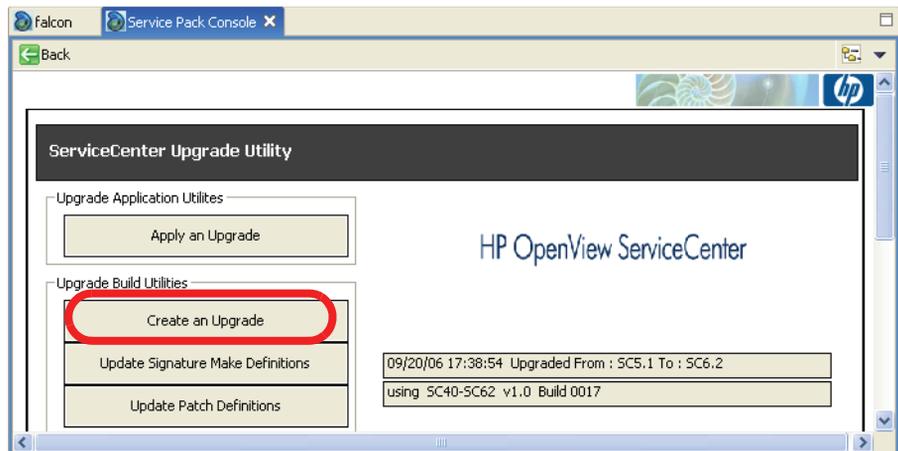
7 Click **Next**.
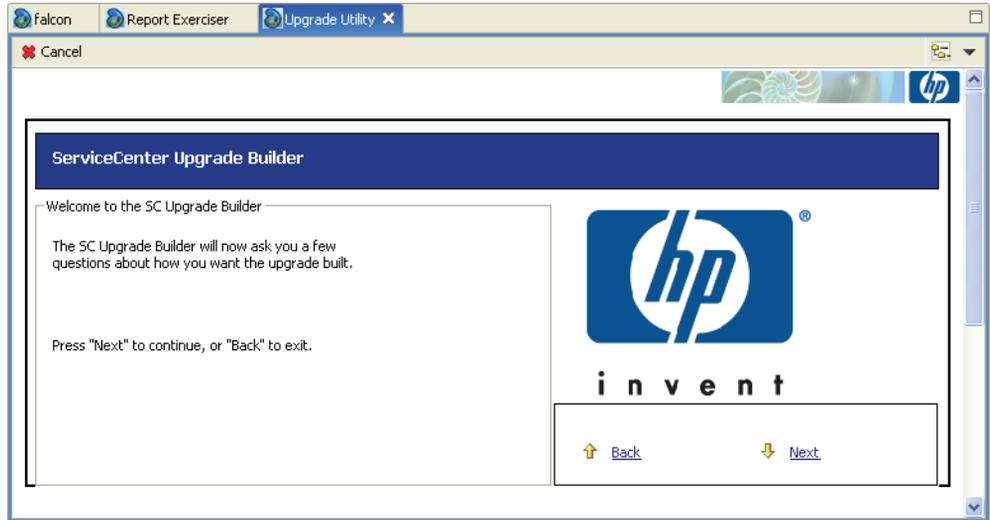
8   Type the name of the release, Service Manager 7.10. Click **Next**.

9   Type a name for the custom upgrade. Click **Next**.

10 Type the fully qualified path to the directory where the Upgrade Utility should create
    and export files (the empty CustomUpgrade directory). Remember this path for later
    use.

    ▶   This should be different than the upgrade directory used to upgrade the test
        system.

    The path must contain a final forward slash (/) or back-slash (\), depending on your
    operating system. Click **Next**.

11 If the Upgrade Utility displays the screen, select the supported languages. Click **Next**.

12 Select the SC6.2 patch file from the menu. Click **Next**.

13 Select **Complete Upgrade Build**. Click **Next.**

    ▶   Choose another option only when directed by HP Customer Support.

14 Click **Yes** to activate internal logging; click **No** to skip internal logging. Click **Next**.

15 Click **Next** to start the Upgrade Builder.

16 A warning message appears.

```
This process will destroy any existing upgrade definitions on
file. Proceed?
```

Click **Yes**. This action destroys any upgrade definitions currently on file and creates the custom upgrade for this system.

▶ The amount of time to create the custom upgrade varies depending on your system resources.

The Service Manager Upgrade Builder monitors the signature process of the build.

The signature process is complete when the Upgrade Utility displays the following message.

```
Finished creating transfer files for the upgrade.
```

17 Copy the upglang.unl and self_service.unl files from the Upgrade directory to the CustomUpgrade directory.

18 Make a checkpoint backup that includes both the data directory and the custom upgrade files.

▶ The transfer.bin created by the Upgrade Utility will be up to 50% larger than the out-of-box version. The upgrade.dta file may be twice as large.

Step 2: Test the custom upgrade

Before you can test your custom upgrade, configure a test environment where you can test all features, especially customized applications. Use the same process you used when configuring a development environment.

Add empty CustomUpgrade and Upgrade directories and complete the steps in Phase II: Preparing for an upgrade on page 29. You must complete the following tasks:

- Verify that you have an archive backup of your production system.

- Complete the tasks to upgrade the server runtime environment. See Tasks for upgrading your server runtime environment on page 45.

- Install the Service Manager Upgrade Utility.

- Run the custom upgrade on the test system.

The test environment should be identical to your production environment in every way.

Tables not upgraded by the Upgrade Utility

The Upgrade Utility does not automatically upgrade all tables and records. The patches file lists the tables and records that are moved into the custom upgrade files. Customizations made to any other table or record will not be part of the custom upgrade.

- To see the changes that were moved into the custom upgrade, refer to the patches table.

- To preserve changes not captured by the Upgrade Utility, unload the customizations and load them into the new system separately.

Run the Upgrade Utility again

The steps to complete a custom upgrade are identical to those to complete an out-of-box upgrade with a few exceptions, such as entering the path to the custom upgrade when asked for the path to the upgrade files. Complete the steps in Phase IV: Running the Upgrade Utility on page 49. You must complete the following tasks:

- Load `preupg.bin` and `transfer.bin`.

- Run the Upgrade Utility.

- Check that background schedulers are finished.

- Note the start and end time. The elapsed time indicates how long the production system will be unavailable during the production upgrade.

- Return the system to a normal operating environment.

The Upgrade Utility repeats the upgrade process, using the mirror image of the production system and custom upgrade files you created in Step 1: Build the custom upgrade on page 101. Verify that the exception reports show no conflicts. You must resolve any remaining upgrade differences.

Test the result with Service Manager

Test the upgraded system with the new Service Manager client. If the upgrade process has any problems, you may need to repeat the steps in Phase V: Resolving Conflicts on page 75 to resolve the remaining differences.

To test the custom upgrade:

1   Use the instructions in the *Service Manager Installation Guide*, to install and configure the Service Manager client for the target version.

2   Use the new Service Manager client to log on.

3   Review the features described in the Service Manager 7.10 Help.

4   Use the new Service Manager client to test the upgraded system thoroughly. Test all features that your users will access. Pay particular attention to areas that were modified on your system.

5   If you plan to use the Web client, you may find subtle differences with forms that need to be corrected.

When the custom upgrade works properly and passes all tests, you can use it to upgrade your production system.

Step 3: Upgrade your production system

The final task of the upgrade process is to apply the custom upgrade that you created to your production system. This process is identical to the one you followed in upgrading your test system.

⛔ Do not apply any upgrade that has not been thoroughly tested to your production system.

When you upgrade the production system:

- The production system is not available to the users while you are applying the custom upgrade.

- Ensure the upgrade files you created are accessible to the production system (the files are located on the same server).

- If you transfer the files to your production system by FTP, set FTP to binary mode.

- You must have adequate disk space.

To upgrade the production system:

1 Ensure that you still have an archive copy of the original production system and all data.

2 Stop the production system, including clients, server, and console.

3 Log on as a system administrator.

4 Prepare the system. Complete the steps in Phase II: Preparing for an upgrade on page 29. The steps for setting up the production environment are identical to those for the development and test environments, with a few exceptions. For example you don't have to duplicate the production environment. You must complete these tasks:

   a Install the Service Manager Upgrade Utility.

   b Make RDBMS preparations.

   c Install the client and server for the upgrade.

   d Complete the tasks to upgrade the server runtime environment. See Tasks for upgrading your server runtime environment on page 45.

5 Complete the steps in Phase IV: Running the Upgrade Utility on page 49. You must complete the following tasks:

   a Load `preupg.bin` and `transfer.bin`.

   b Run the Upgrade Utility.

   c Check that background schedulers are finished.

   d Return the system to a normal operating environment.

The Upgrade Utility repeats the upgrade process again, using the real production system and the custom upgrade files.

6 Use the new Service Manager client to test the upgraded system again.

When you successfully upgrade the production system, your application upgrade is complete.

Step 4: Purge upgrade files

After you successfully applied the custom upgrade to your production system, you can remove all the unnecessary files created by this process from your development system by running the Purge utility. You do not have to purge the files if the custom upgrade was created on a backup of your development system that will no longer be used.

Do not purge the upgrade files if the upgrade fails. Do not purge the upgrade until you have the production system running successfully. If the upgrade fails, you can restart where it stopped. It may be necessary to rebuild a custom upgrade, but you can do so only if the upgrade files have not been purged.

To run the Purge utility:

1   Type *aapm.upgrade.purge on the Service Manager client command line. Press **Enter**.

2   Click **I'm done, and I want to remove the upgrade files completely**.

3   Click **OK**.

Step 5: Train users on updated applications

Before implementing the upgraded system into production, train your users on any new features that they might need. To learn about the new features in Service Manager 7.10, see the Service Manager 7.10 Help. HP Education Services also offers a complete training schedule.

A Server changes required by release

This appendix contains a list of actions required to upgrade your server run-time environment (RTE) from its current version to HP Service Manager 7.10. Refer the section that applies to your current version of ServiceCenter or Service Manager for a list of required upgrade steps.

ServiceCenter 6.2 to HP Service Manager 7.10

You must perform the following actions to upgrade your server run-time environment (RTE) from ServiceCenter 6.2 to HP Service Manager 7.10.

Table 1　　**Changes required to upgrade the server RTE from ServiceCenter 6.2**

Feature	Change	Action required
Data persistence	P4 database removed	Convert to a supported RDBMS

Required configuration file parameter changes

You must make several configuration file (sm.ini) parameter changes to upgrade your system. Some existing parameters have new names. See the New parameter name column for parameters that have been renamed. Other existing parameters are now obsolete and require you to use new functionality. See the Required action column for the new parameter, if any, that replaces the obsolete parameter.

Table 2   **Required configuration file parameter changes**

Previous parameter name	New parameter name	Required action
apiserver		This parameter is obsolete. All client listener processes now connect through dedicated HTTP and HTTPS ports.
cacertpem	truststoreFile	Move your CA certificate from a PEM file to a Java keystore and provide the path to this keystore with the truststoreFile parameter.
certpem	keystoreFile	Move your server certificate from a PREM file to a Java keystore and provide the path to this keystore with the keystoreFile parameter.
clustername		This parameter is obsolete. You can setup a failover configuration by starting a second load balancer process on a separate system.
cstrace		This parameter is obsolete.
debugdtevents		This parameter is obsolete.
debugdtrecords		This parameter is obsolete.
debugdttrace		This parameter is obsolete.
debugdtworld		This parameter is obsolete.
debuglog		This parameter is obsolete.
debugrpc		This parameter is obsolete.

Table 2   **Required configuration file parameter changes**

Previous parameter name	New parameter name	Required action
debugtransport		This parameter is obsolete.
dhpem	keystoreFile	This parameter is obsolete. You can set DH encryption properties when you create your Java keystore.
immediateshadow		This parameter is obsolete. Use your RDBMS utilities for reporting or data back up.
ip_address		This parameter is obsolete. Use AutoPass to specify valid IP addresses for your system.
keepalive		This parameter is obsolete.
odbccharacterarray		This parameter is obsolete.
pkpem	keystoreFile	Move your server private key from a PEM file to a Java keystore and provide the path to this keystore with the keystoreFile parameter.
pkpempass	keystorePass	Move your server private key from a PEM file to a Java keystore and provide the password to this keystore with the keystorePass parameter.
sccluster	group	This parameter has been renamed. Use the new parameter to shutdown your virtual group.
scclusterbindaddress	groupbindaddress	This parameter has been renamed. Use the new parameter to define your server group.
scclustermcastaddress	groupmcastaddress	This parameter has been renamed. Use the new parameter to define your server group.

Table 2   **Required configuration file parameter changes**

Previous parameter name	New parameter name	Required action
scclustername	groupname	This parameter has been renamed. Use the new parameter to define your server group.
scclusterport	groupport	This parameter has been renamed. Use the new parameter to define your server group.
scdebug		This parameter is obsolete. Contact customer support if you need to debug customized custom RAD applications from previous versions.
scemail	emailout	This parameter has been renamed. Use the new parameter to process outbound e-mail from HP Service Manager.
schost	host	This parameter has been renamed. Use the new parameter to shutdown or standby your system.
sctimeramount	heartbeatinterval	This parameter has been renamed. Use the new parameter to keep client connections open.
servletcontainer		This parameter is obsolete. You no longer need this parameter to start client listener processes.
soapaccepttimeout	sessiontimeout	This parameter is obsolete. Web services client connections now use the client time out limits.
soapreceivetimeout	sessiontimeout	This parameter is obsolete. Web services client connections now use the client time out limits.
sqldetect		This parameter is obsolete. The Database Dictionary now automatically detects changes to RDBMS tables and columns.

Table 2　　　**Required configuration file parameter changes**

Previous parameter name	New parameter name	Required action
sqlidentify		This parameter is obsolete.
sqllogintime		This parameter is obsolete.
sqlmodcount		This parameter is obsolete. The Database Dictionary now automatically detects changes to RDBMS tables and columns.
ssl_trustedClientspem	trustedclientsJKS	Move your list of trusted clients from a PEM file to a Java keystore and provide the path to this keystore with the trustedkeystoreJKS parameter.
timeoutlimit	sessiontimeout	This parameter has been renamed. Use the new parameter to set your client's session time out.
validateodbcfieldnames		This parameter is obsolete.

Required web tier configuration file parameter changes

You must make the following web tier configuration file (web.xml) parameter changes to run an upgraded web tier. Some existing parameters have new names. See the New parameter name column for parameters that have been renamed. Other existing parameters are now obsolete and require you to use new functionality. See the Required action column for the new parameter, if any, that replaces the obsolete parameter.

Table 3   Required web tier configuration file parameter changes

Previous parameter name	New parameter name	Required action
sc.autosubmit	autoSubmit	This parameter has been renamed. Use the new parameter to configure your web tier.
sc.cacerts	cacerts	This parameter has been renamed. Use the new parameter to configure your web tier.
sc.compress_soap	compress_soap	This parameter has been renamed. Use the new parameter to configure your web tier.
sc.helpserverhost	helpServerHost	This parameter has been renamed. Use the new parameter to configure your web tier.
sc.helpserverport	helpServerPort	This parameter has been renamed. Use the new parameter to configure your web tier.
sc.honorUrlHost	honorUrlHost	This parameter has been renamed. Use the new parameter to configure your web tier.
sc.honorUrlPort	honorUrlPort	This parameter has been renamed. Use the new parameter to configure your web tier.
sc.host	serverHost	This parameter has been renamed. Use the new parameter to configure your web tier.

Table 3   **Required web tier configuration file parameter changes**

Previous parameter name	New parameter name	Required action
sc.hscale	hscale	This parameter has been renamed. Use the new parameter to configure your web tier.
sc.port	serverPort	This parameter has been renamed. Use the new parameter to configure your web tier.
sc.querysecurity	querySecurity	This parameter has been renamed. Use the new parameter to configure your web tier.
sc.refreshmessages	refreshMessages	This parameter has been renamed. Use the new parameter to configure your web tier.
sc.refreshmessinterval	refreshMessagesInterval	This parameter has been renamed. Use the new parameter to configure your web tier.
sc.ssl	ssl	This parameter has been renamed. Use the new parameter to configure your web tier.
sc.useservertabs	useservertabs	This parameter has been renamed. Use the new parameter to configure your web tier.
sc.viewactivenotes	viewactivenotes	This parameter has been renamed. Use the new parameter to configure your web tier.

Table 3   **Required web tier configuration file parameter changes**

Previous parameter name	New parameter name	Required action
sc.viewclassicmenu	viewclassicmenu	This parameter has been renamed. Use the new parameter to configure your web tier.
sc.viewpromptforsave	viewpromptforsave	This parameter has been renamed. Use the new parameter to configure your web tier.
sc.viewrecordlist	viewrecordlist	This parameter has been renamed. Use the new parameter to configure your web tier.

B Application changes required by release

This appendix contains a list of actions required to update your server applications from their current version of HP Service Manager or ServiceCenter to Service Manager 7.10. See the section matching your current version of the product for a list of the required update steps.

ServiceCenter 6.2 to Service Manager 7.10

You must perform the following actions to update your applications from ServiceCenter 6.2 to Service Manager 7.10.

Table 4   Required changes to update applications from ServiceCenter 6.2

Application	Change	Action Required
ODBC	Service Manager requires starting a legacy Service Manager listener to support existing ODBC connections.	See the *HP Service Manager 7.10 Installation Guide* for information on starting and configuring a legacy Service Manager listener.
Service Level Management	Service Level Agreement records have a new required SLA Type field.	The upgrade utility populates the SLA type field with the value of "Customer" for all existing SLAs.
Service Level Management of Change Management	Service Level Agreement requires a new field (agreement.ids) to be added to the SLA configuration record to track multiple SLAs per change record. Without the new field, each change record can only be associated to one SLA at a time.	1 Log on to HP Service Manager with an administrator account. 2 Click **Service Level Management** > **SLA Administration** > **Configure Application**. 3 Type **cm3r** in the Table Name field, and click **Search**. 4 In SLA ID Field, select **agreement.ids**. 5 Click **Save**.
Service Level Management of Change Management	Service Level Agreement requires a new field (agreement.ids) to be added to the SLA configuration record to track multiple SLAs per change task record. Without the new field, each change task record can only be associated to one SLA at a time.	1 Log on to HP Service Manager with an administrator account. 2 Click **Service Level Management** > **SLA Administration** > **Configure Application**. 3 Type **cm3t** in the Table Name field, and click **Search**. 4 In SLA ID Field, select **agreement.ids**. 5 Click **Save**.

Table 4   Required changes to update applications from ServiceCenter 6.2 (cont'd)

Application	Change	Action Required
Service Level Management of Incident Management	Service Level Agreement requires a new field (agreement.ids) to be added to the SLA configuration record to track multiple SLAs per incident record. Without the new field, each incident record can only be associated to one SLA at a time.	1 Log on to HP Service Manager with an administrator account. 2 Click **Service Level Management** > **SLA Administration** > **Configure Application**. 3 Type **probsummary** in the Table Name field, and click **Search**. 4 In SLA ID Field, select **agreement.ids**. 5 Click **Save**.
Service Level Management of Problem Management	Service Level Agreement requires a new field (agreement.ids) to be added to the SLA configuration record to track multiple SLAs per problem record. Without the new field, each problem record can only be associated to one SLA at a time.	1 Log on to HP Service Manager with an administrator account. 2 Click **Service Level Management** > **SLA Administration** > **Configure Application**. 3 Type **rootcause** in the Table Name field, and click **Search**. 4 In SLA ID Field, select **agreement.ids**. 5 In CI Fields, type **affected.item**. 6 Click **Save**.
Service Level Management of Problem Management	Service Level Agreement requires a new field (agreement.ids) to be added to the SLA configuration record to track multiple SLAs per problem task record. Without the new field, each problem task record can only be associated to one SLA at a time.	1 Log on to HP Service Manager with an administrator account. 2 Click **Service Level Management** > **SLA Administration** > **Configure Application**. 3 Type **rootcausetask** in the Table Name field, and click **Search**. 4 In SLA ID Field, select **agreement.ids**. 5 In CI Fields, type **affected.item**. 6 Click **Save**.

Application	Change	Action Required
Service Level Management of Service Desk	Service Level Agreement requires a new field (agreement.ids) to be added to the SLA configuration record to track multiple SLAs per interaction record. Without the new field, each interaction record can only be associated to one SLA at a time.	1 Log on to HP Service Manager with an administrator account. 2 Click **Service Level Management** > **SLA Administration** > **Configure Application**. 3 Type **incidents** in the Table Name field, and click **Search**. 4 In SLA ID Field, select **agreement.ids**. 5 In CI Fields, type **affected.item**. 6 Click **Save**.
Table change	The application upgrade utility copies the Web services API fields from the datadict table to the extaccess table. After the upgrade, your system will only use the extaccess table to manage Web services.	Review the Web services mappings in the extaccess table and verify that they list the desired service and object names.
Change Management with Web services	The following fields in the cm3r table must be exposed following an upgrade of the RTE: header,assign.dept affected.item closureComments These are now mandatory fields when opening or closing a Change record, so they must be accounted for in any Web services definitions. HP also recommends that the emergency field in the cm3r table be exposed as well, to support emergency Changes.	To expose the fields, add them to the appropriate extaccess records for the cm3r table published in the Web services API. header,assign.dept (suggested Caption=AssignmentGroup) affected.item (suggested Caption=Service) closureComments (suggested Caption=ClosureComments) emergency (suggested Caption=Emergency)

Table 4   **Required changes to update applications from ServiceCenter 6.2 (cont'd)**

Application	Change	Action Required
Incident Management with Web services	The affected.item field in the cm3r table must be exposed following an upgrade of the RTE. This is now a mandatory field when opening or closing an Incident record, so it must be accounted for in any Web services definitions.	To expose the field, add it to the appropriate extaccess records for the probsummary table published in the Web services API. affected.item (suggested Caption=Service)
Key change	device	Key added: Key type: Nulls & Duplicates. Key field(s): ucmdb.id
Key changes	displayeventrev	Key changed to: Key Type: Unique Key field(s): id, sc.revision From: Key Type: Unique Key field(s): screen.id, language, event, event.sig, sc.revision
Key changes	displayoptionrev	Key changed to: Key Type: Unique Key field(s): id, sc.revision From: Key Type: Unique Key field(s): screen.id, language, txt.bank, txt.option, text.sig, sc.revision
Key changes	incidents	Key added: In order to be able to order from catalog from an interaction. Key Type: Nulls & Duplicates Key field(s): cartId

Table 4   Required changes to update applications from ServiceCenter 6.2 (cont'd)

Application	Change	Action Required
Key change	svcCatalog	This must be done so that the Service Catalog search searches both the description and the displayName fields. Key removed: Key Type: IR key Key field(s): description, name.ir

Service Manager 7.00 to 7.10

You must perform the following actions to update your applications from Service Manager 7.00 to Service Manager 7.10.

Table 5   Required changes to update applications from Service Manager 7.00

Application	Change	Action Required
Incident Management	Some form names used to start with the prefix "problem" (for Problem Management). Form names now use the prefix "IM" (for Incident Management). Your existing category records may reference form names that begin with "problem".	Change your category records to point to forms with the prefix "IM" rather than the prefix "problem".
IR Expert	IR Expert indexes now reside in the scirexpert table instead of in external ir.* files.	Regenerate the IR Expert index.
Key changes	The erddef table requires a new Nulls & Duplicates key.	Add a second key for the erddef table. Type: Nulls & Duplicates Fields in key: • distributed
Key changes	The inbox table requires updated fields for the unique key.	Update the fields in the Unique key for the inbox table. Type: Unique Fields in key: • container.id • inbox.name • inbox.type • operator.name
Key changes	The ocmgroups table requires a new Nulls & Duplicates key.	Add a key for the ocmgroups table. Type: Nulls & Duplicates Fields in key: • members

Table 5   Required changes to update applications from Service Manager 7.00 (cont'd)

Application	Change	Action Required
Key changes	The operator table requires a new Nulls & Duplicates key.	Add a key for the operator table. Type: Nulls & Duplicates Fields in key: • contact.name
Key changes	The schedule table requires a new Nulls & Duplicates key.	Add a ninth key for the schedule table. Type: Nulls & Duplicates Fields in key: • application • object • type
Key changes	Device: if using UCMDB	Add a Nulls & Duplicates key if using UCMDB: Key field(s): ucmdb.id
Key changes	displayeventrev	Update the fields in the Unique Key to: Key field(s): id, sc.revision From: Key field(s): screen.id, language, event, event.sig, sc.revision
Key changes	displayoptionrev	Update the fields in the Unique Key to: Key field(s): id, sc.revision From: Key field(s): screen.id, language, txt.bank, txt.option, text.sig, sc.revision
Key changes	svcCatalog (So that the search will search both the description and the displayName fields)	Remove IR Key: Key field(s): description, name.ir
Key changes	Incidents (In order to be able to order from catalog in an interaction)	Add a Nulls & Duplicates Key: Key field(s): cartId

Table 5　　Required changes to update applications from Service Manager 7.00 (cont'd)

Application	Change	Action Required
Link records	The Link record form used to display information about the target.file and format name in one column. The Link record form now needs separate columns for the target file name and the target format name.	Update the Link record form to view the target file name and the target format name in separate columns.
ODBC	Service Manager requires starting a legacy Service Manager listener to support existing ODBC connections.	See the *HP Service Manager 7.10 Installation Guide* for information on starting and configuring a legacy Service Manager listener.
Operators	HP Service Manager 7.10 requires that every operator have a contact record. You can still have a contact record without a corresponding operator record.	Review your existing operator records and create a contact record for every operator record that does not already have one. HP Service Manager 7.10 provides four wizards that you can use to create the missing contact or operator records. These wizards are available from the Options menu of the contacts.g and operator.g forms. • Create Operator - Creates an operator record from a contact record. • Mass Create Operators - Creates a batch of operator records from a batch of contact records • Create Contact - Creates a contact record from an operator record. • Mass Create Contacts - Creates a batch of contact records from a batch of operator records. Disable the "Synchronize contacts with operators" option in the System Information Definition before running these wizards. You can customize how the wizards map fields from the contacts and operator tables by editing the `createUsers` JavaScript in the script library.

Table 5   Required changes to update applications from Service Manager 7.00 (cont'd)

Application	Change	Action Required
Problem Management (formerly Root Cause)	Profile records now contain settings to control access to Problems, Problem Tasks, and Known Errors.	After you update your applications the PROBLEM MANAGER profile will be the only profile with privileges to access Problem Management Tasks, Known Errors, or any new options added for Problems. Update your profile records to enable users to view and modify records within Problem Management.
Service Catalog	The ApprovalDef table supporting Service Catalog approvals has been converted to a system table. You should not change the name of this table. The upgrade utility adds this table but does not populate it with any approval definition records.	Add approval definition records to support your Service Catalog approval process.
Service Desk (formerly Service Management)	The Service Desk application uses three different fields to list contacts for new calls: • Contact for this call (callback.contact) • This call is for (contact.name) • Reported By (alternate.contact) Once you select a contact (callback.contact), the other two fields, if left blank, default to the callback.contact. There is no longer a check box for the alternate.contact field; this field is now always populated.	Specify the contact for the callback.contact, contact.name, and alternate.contact fields for each existing Service Desk ticket in your system.

Table 5   Required changes to update applications from Service Manager 7.00 (cont'd)

Application	Change	Action Required
Service Level Management	Service Level Agreements have a new field that indicate the start date of the agreement. HP Service Manager uses the agreement start date to determine whether an SLA is valid. The start date may also be needed for accurate reporting.	The upgrade utility populates the start date field with the date January 1, 2001, for all existing SLAs. Review each SLA and provide the correct start date in the "SLA in effect from" field.
Service Level Management	Service Level Management uses configuration records for each application to determine the fields and conditions that are used during SLA processing. The upgrade populates the Response States for the Incident Management application with information from the pmstatus table. Each record in the pmstatus table has a sequence number, and this sequence determines the order of the states in the Response State Progression array field of the probsummary slamodulecontrol record (the SLA configuration record for the Incident Management application).	Review the Response State Progression field and modify or remove any states that do not apply to your implementation. Ensure that the states in this field are listed in the order that they should occur during the life cycle of an incident ticket. (For example, Closed comes after Open).
Service Level Management	Service Level Management uses application profiles to determine user rights within the module. After the upgrade, all users will have the DEFAULT profile which has limited permissions.	Review the available profiles, and edit them or add new ones as necessary. Update your SLM users to have the correct profile.

Table 5   Required changes to update applications from Service Manager 7.00 (cont'd)

Application	Change	Action Required
Service Level Management	Service Level Management uses application profiles to determine user rights within the module. After the upgrade, no operator will have an assigned SLA profile. All operators will use the DEFAULT profile, which has limited permissions.	Update the operator records of each operator who needs access to SLA, and give them appropriate permissions.
Service Level Management	The "target" field was removed from the sla table in the out-of-box system. Upgraded systems will still have a data policy record with a validation rule for this field.	Remove the validation entry for the "target" field in the datadict record for the sla table.
Table change	The deviceparent table has been removed and replaced by the cirelationship table. The upgrade utility does not delete your pre-existing deviceparent records but rather converts each deviceparent record into a cirelationship record as follows: • Field Name in deviceparent -> Field Name in cirelationship • parent ->logical.name • logical.name ->1 in related.cis • relationship ->relationship.type • port.no ->port.no • comments ->comments • logical.name ->relationship.name • outage.dependency ->True • outage.threshold ->1 After the upgrade, your system writes new records to the cirelationship table.	After you convert data to cirelationship, check each record and make sure information converted correctly. Do not purge the deviceparent table until all of the records have been converted successfully to the new table. You can manually set fields that were not in the deviceparent, such as outage.dependency and outage.threshold. The cirelationship table is one too many. You can consolidate multiple record in deviceparent into 1 record in cirelationship table. We provide a consolidation wizard. You can launch the wizard by selecting Consolidate Relationships from the options menu when viewing a CI.

Table 5   Required changes to update applications from Service Manager 7.00 (cont'd)

Application	Change	Action Required
Service Level Management	Service Level Agreement records have a new required SLA Type field.	The upgrade utility populates the SLA type field with the value of "Customer" for all existing SLAs.
Change Management with Web services	These fields in the cm3r table must be exposed following an upgrade of the RTE: header,assign.dept affected.item closureComments These are now mandatory fields when opening or closing a Change record, so they must be accounted for in any Web services definitions. HP also recommends that another field in the cm3r table be exposed as well, to support emergency Changes.	To expose the fields, add them to the appropriate extaccess records for the cm3r table published in the Web services API. header,assign.dept (suggested Caption=AssignmentGroup) affected.item (suggested Caption=Service) closureComments (suggested Caption=ClosureComments) emergency (suggested Caption=Emergency)
Incident Management with Web services	The affected.item field in the cm3r table must be exposed following an upgrade of the RTE. This is now a mandatory field when opening or closing an Incident record, so it must be accounted for in any Web services definitions.	To expose the field, add it to the appropriate extaccess records for the probsummary table published in the Web services API. affected.item (suggested Caption=Service)

C Optional application changes

This appendix lists all the optional application changes you can make to improve your system performance, usability, or compliance with ITIL conventions:

Table changes

The following sections list the data definition changes to tables and columns that the upgrade utility makes when updating your applications to Service Manager 7.10. See the section for your current version of ServiceCenter or Service Manager for a cumulative list of table changes.

Table changes from ServiceCenter 6.2 to HP Service Manager 7.00

The upgrade utility makes the following data definition changes to tables and columns updating your applications from ServiceCenter 6.2 to Service Manager 7.00.

Table 6   Table changes made updating from ServiceCenter 6.2

Application	Change	Suggested action
Configuration Management	The upgrade utility adds the following fields to the environment table. • cm3SubscriptionAdd – lists the Change Management category to use when a user adds a subscription to a configuration item. • cm3SubscriptionRemove – lists the Change Management category to use when a user removes a subscription to a configuration item. • subscribeMode – lists the application that manages user subscriptions to configuration items.	Review the Configuration Management environment record and select the subscription features you want to enable.

Table 6   Table changes made updating from ServiceCenter 6.2 (cont'd)

Application	Change	Suggested action
Change Management	Now supports the use of multiple SLAs for change records.	If you currently use Event Services that have event maps against the agreement.id field, update the event map or create a new one that maps to the agreement.ids field (array of numbers).
		If you currently expose the aggreement.id field in any existing *extaccess* records, change the field to agreement.ids.
Incident Management	Now supports the use of multiple SLAs for incidents.	If you currently use Event Services that have event maps against the agreement.id field, update the event map or create a new one that maps to the agreement.ids field (array of numbers).
		If you currently expose the aggreement.id field in any existing *extaccess* records, change the field to agreement.ids.
		If you are currently using SLA for incidents, add the affected.item field to the CI Fields section of the SLA Integration Information record in the incidents table.

Table 6   Table changes made updating from ServiceCenter 6.2 (cont'd)

Application	Change	Suggested action
Problem Management	Now supports the use of multiple SLAs for problem records and problem tasks.	If you currently use Event Services that have event maps against the agreement.id field, update the event map or create a new one that maps to the agreement.ids field (array of numbers).
		If you currently expose the aggreement.id field in any existing *extaccess* records, change the field to agreement.ids.
		If you are currently using SLA for problem records, add the logical.name field to the CI Fields section of the SLA Integration Information record in the rootcause table.
		If you are currently using SLA for problem tasks, add the logical.name field to the CI Fields section of the SLA Integration Information record in the rootcausetask table.
Service Desk	Now supports the use of multiple SLAs for interactions	If you currently use Event Services that have event maps against the agreement.id field, update the event map or create a new one that maps to the agreement.ids field (array of numbers).
		If you currently expose the aggreement.id field in any existing *extaccess* records, change the field to agreement.ids.

Name changes

The following sections list the name changes that the upgrade utility makes when updating your applications to Service Manager 7.10. See the section for your current version of ServiceCenter for a cumulative list of name changes.

Name changes from ServiceCenter 6.2 to Service Manager 7.00

The upgrade utility makes the following name changes updating your applications from ServiceCenter 6.2 to Service Manager 7.00.

Table 7   **Name changes made updating from ServiceCenter 6.2**

Application	Change	Suggested action
Incident Management	The label "Configuration Item" renamed to "Affected Item" on the create and search incidents forms.	Review customized menus, formats, and messages that refer to Configuration Items to determine if they need to be changed to match the new naming convention.
Service Desk	The label "Configuration Item" renamed to "Affected Item" on the create and search interactions forms.	Review customized menus, formats, and messages that refer to Configuration Items to determine if they need to be changed to match the new naming convention.

Key changes

You can perform the following key changes to optimize the performance of your upgraded system. All of these changes are optional. See the section for your current version of ServiceCenter or Service Manager for a list of performance tuning key changes.

Key changes from ServiceCenter 6.2 to Service Manager 7.00

The following key changes are for ServiceCenter 6.2 to Service Manager 7.00.

Table 8   Key changes to optimize performance from ServiceCenter 6.2

Table	Change	Suggested action
cm3r	Add a new Nulls & Duplicates key	Add a key for the cm3r table. Type: Nulls & Duplicates Fields in key: • logical.name
dept	Add a new Nulls & Duplicates key	Add a key for the dept table. Type: Nulls & Duplicates Fields in key: • serviceRequesters
extaccess	Add a new No Duplicates key	Add a key for the extaccess table. Type: No Duplicates Fields in key: • object.name

Table 8   Key changes to optimize performance from ServiceCenter 6.2 (cont'd)

Table	Change	Suggested action
extaccess	Update the Unique key	Update the fields in the Unique key for the extaccess table. Type: Unique Update fields in key to: • object.name • service.name
inbox	Update the Nulls & Duplicates key	Update the fields in the Nulls & Duplicates key for the inbox table. Type: Nulls & Duplicates Update fields in key to: • inbox.type • inbox.class • operator.name • audience.type • groups • role
incidents	Add a new Nulls & Duplicates key	Add a key for the incidents table. Type: Nulls & Duplicates Fields in key: • affected.item
kmknowledgebaseupdates	Update the Unique key	Update the fields in the Unique key for the kmknowledgebaseupdates table. Type: Unique Update fields in key to: • id
ocmq	Add a new Nulls & Duplicates key	Add a key for the ocmq table. Type: Nulls & Duplicates Fields in key: • logical.name

Table 8   Key changes to optimize performance from ServiceCenter 6.2 (cont'd)

Table	Change	Suggested action
producttype	Update the Unique key	Update the fields in the Unique key for the producttype table. Type: Unique Update fields in key to: • category • subcategory • product.type • company
wfwenvironment	Add a new Unique key.	Add a key for the wfwenvironment table. Type: Unique Fields in key: • name
wfwenvironment	Remove a No Duplicates key.	Remove a key for the wfwenvironment table. Type: No Duplicates Fields in key: • name

Key changes from ServiceCenter 6.2 to Service Manager 7.10

The following key changes are for ServiceCenter 6.2 to Service Manager 7.10.

Table 9   Key changes to optimize performance from ServiceCenter 6.2

Table	Change	Suggested action
cm3r	Add a new Nulls & Duplicates key.	Add a key for the cm3r table: • Key type: Nulls & Duplicates • Key field(s): middle, logical.name

Table 9   Key changes to optimize performance from ServiceCenter 6.2 (cont'd)

Table	Change	Suggested action
counters	Add a new Unique key.	Update the fields in the Unique key for the counters table. Key changed to: • Key Type: Unique • Key field(s): table.name From: • Key Type: Unique • Key field(s): table.name, column.name
dept	Add a new Nulls & Duplicates key.	Add a Nulls & Duplicates key for the dept table. • Key type: Nulls & Duplicates • Key field(s): serviceRequesters
incidents	Add a new Nulls & Duplicates key.	Add a Nulls & Duplicates key for the incidents table • Key type: Nulls & Duplicates. • Key field(s): affected.item
kmknowledgebaseupdates	Add new Unique key.	Add a Unique key for the kmknowledgebaseupdates table. • Key Type: Unique • Key field(s): id
ocmq	Add a new Nulls & Duplicates key.	Add a Nulls & Duplicates key for the ocmq table. • Key type: Nulls & Duplicates • Key field(s): logical.name

Table 9   Key changes to optimize performance from ServiceCenter 6.2 (cont'd)

Table	Change	Suggested action
producttype	Update a Unique key.	Update the fields in the Unique key for the producttype table. Key changed to: • Key type: Unique • Key field(s): category, subcategory, product.type, company"} From: • Key Type: Unique. • Key field(s): category, subcategory, product.type, company, variable1, variable2, variable3
notification	Update a Unique key.	Update the fields in the Unique key for the notification table. Key changed to: • Key Type: Unique • Key field(s): id From: • Key Type: Unique • Key field(s): name, str.condition
datadict	Removed a Nulls & Duplicates.	Remove the Nulls & Duplicates key type in the datadict table. • Key type: Nulls & Duplicates • Key field(s): {}
extaccess	Add a new No Duplicates key.	• Add a No Duplicates key for the extaccess table. • Key type: No Duplicates. • Key field(s): "object.name"

Table 9   Key changes to optimize performance from ServiceCenter 6.2 (cont'd)

Table	Change	Suggested action
extaccess	Update a Unique key.	Update the fields in the Unique key for the extaccess table. Key changed to: • Key type: Unique • Key field(s): object.name, service.name From: • Key type: Unique • Key field(s): name,
inbox	Update a Nulls & Duplicates key.	Update the fields in the Nulls & Duplicates key for the inbox table. Key changed to: • Key type: Nulls & Duplicates • Key field(s): inbox.type, inbox.class, operator.name, audience.type, groups, role From: • Key type: Nulls & Duplicates • Key field(s): inbox.type, inbox.class, operator.name, groups
Object	Remove a Nulls & Duplicates key.	Remove the Nulls & Duplicates key for the Object table. • Key type: Nulls & Duplicates • Key field(s): {}

Key changes from Service Manager 7.0 to Service Manager 7.10

The following key changes are for Service Manager 7.0 to Service Manager 7.10.

Table 10   Key changes to optimize performance from Service Manager 7.00

Table	Change	Suggested action
counters	Key change	Update the fields in the unique key to: • Key field(s): table.name • From: Key field(s): table.name, column.name
incidents (for search)	Key change	Add a Key: • An IR key with Key field(s): description, resolution
notification	Key change	Unique Key updated to: • Key field(s): id • From: Key field(s): name, str.condition

D Using the RAD comparison utility

HP Service Manager bundles the RAD Comparison Utility with the Service Manager Upgrade Utility to help you examine the differences between the new and old versions of an application. The RAD Comparison Utility displays a list of differences between elements, such as panels and lines that do not match. Use this utility to examine those applications where you must decide to keep or replace the old version or merge objects into a single version. You can view the comparison results online or print them. You can also view the old and new versions of each panel.

Topics in this section include:

The RAD Comparison Utility

To access the RAD Application Comparison utility, type `agcompare` on the HP Service Manager client command line. Press **Enter**.
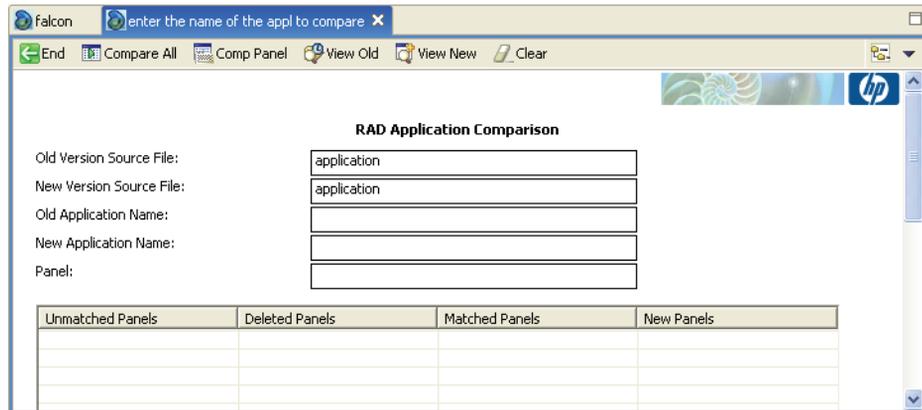
Toolbar buttons

The following table describes the functions in the Toolbar.

Table 1   RAD comparison utility toolbar options

Button	Definition
End	Returns you to the previous screen.
Compare All	Compares all the panels of the new version of the application named with all the panels of the old version.
Comp Panel	Compares the old and new versions of a single panel named in the Panel field of the Compare Application form.
View Old	Displays the old version of the panel named in the Panel field of the Compare Application form.

Table 1   **RAD comparison utility toolbar options (cont'd)**

Button	Definition
View New	Displays the new version of the panel named in the Panel field of the Compare Application form.
Clear	Clears all data from the panel comparison fields (Unmatched, Deleted, New, and Matched), and prepares the utility to perform another comparison. This option does not affect the old or new RAD applications.
Print All	Sends all currently displayed panel records to the printer.

Fields

The following table describes the fields on the RAD Application Comparison form.

Table 2   **RAD comparison utility field descriptions**

Field	Definition
Old Version Source File	Specifies the logical file containing the application records (RAD panels) referenced by **Old Application Name**. The default value of this field is `application`.
New Version Source File	Specifies the logical file containing the application records (RAD panels) referenced by **New Application Name**. The default value of this field is `application`.
Old Application Name	Specifies the name of the application that resides in **Old Version Source File**. The old version source file is the model for the application specified by **New Application Name**.
New Application Name	Specifies the name of the application that resides in **New Version Source File** that you want compare with the application specified by the **Old Application Name**.
Panel	Specifies the panel the utility should use for comparison checking. You can view the specified panel in the old or new version of the application.

Table 2   RAD comparison utility field descriptions (cont'd)

Field	Definition
Unmatched panels	A list of panel names where there are differences between the old and new versions of the RAD application. Warning: Do not make manual changes to the content of this field.
Deleted panels	Contains a list of panel names in the old version of the application that are not in the new version. Warning: Do not make manual changes to the content of this field.
Matched panels	Contains a list of panel names where there are no differences between the old and new versions of the RAD application. Warning: Do not make manual changes to the content of this field.
New panels	Contains a list of all panel names in the new version of the application and not in the old version. This field is set to allow for modifications and easy tabbing. Warning: Do not make manual changes to the content of this field.

⚠ By default, Service Manager contains one Application Library. If you find it necessary to maintain two or more Application Libraries, you must allocate and control the functionality of these files, and define the routines necessary for the exchange of data between files.

Comparing applications

You can compare single panels or all panels in an application.

To compare all panels:

1   Access the RAD Application Comparison utility.

2   Type the appropriate data in the **Old Application Name** and **New Application Name** fields. Depending on the choice you made during the upgrade process, either the new version will have the name NEW<vvvnnn>, where vvv is the release name and nnn is the name of the object, or the old version will have the name OLD<vvvnnn>. For example, a new check.field.name could be NEWSC6.2check.field.name, or the old one could be OLDSC4check.field.name.

3   Click **Compare All**. Service Manager displays summary lists of Matched Panels, Unmatched Panels, Deleted Panels, and New Panels where they apply. Make selections from these lists and display additional panel information by choosing options in the Toolbar.

4   You can view old and new versions of application panels, review the detailed comparison results for panels in the Unmatched Panels list, and print all the panels.

5   To compare a different application, click **Clear** before proceeding. This resets all controls before processing begins.

Printing a report

To print a report of the Comparison Results, click **Print All**. The comparison report contains the following information.

Table 3   Comparison report items

Reported Item	Definition
Matched Panels	Summary of all panels that matched.
Unmatched Panels	Summary of all panels showing a change, printouts of the old and new versions of each unmatched panel, and the comparison results of each panel.
Deleted Panels	Summary of all deleted panels and a printout of each deleted panel.
New Panels	Summary of all new panels and a printout of each panel.

Comparing single panels

You can compare single panels before or after comparing the entire application.

To view a detailed comparison of a single panel after comparing the entire application:

•   Select the desired panel in the **Unmatched Panels** column. Click **Comp Panel**.

To view a detailed comparison of a single panel before comparing the entire application:

•   Type the name of the panel in the **Panel** field. Click **Comp Panel**.

The Detail Listing of Differences form displays the names of the Old and New Version Source Files and the names of the Old and New Application Name fields. The following figure shows the Detail Listing of Differences report.

Array and scalar field differences

When the old and new versions of an array or scalar field have differences, the information appears in the following format.

```
Line xx:action prompt
```

where:

xx is the element number of the array that the Upgrade Utility deleted or added.

action is the change to an element of an array. The old version of the element is considered deleted and the new version of the element is considered added. Changes to array or scalar field elements are accompanied by the phrases ADDED TO and DELETED FROM.

prompt is the name of the prompt on the RAD panel that corresponds to the array or scalar field that the Upgrade Utility changed.

Viewing versions of a panel

Click **View Old** or **View New** from the Detail Listing of Differences form or from the RAD Comparison form after a complete comparison. Type the name of the panel in the **Panel** field or select the panel from one of the lists.

Continuation lines

If the contents of a scalar field or an element of an arrayed field are greater than 72 bytes, the comparison utility uses continuation lines. Continuation lines appear in viewable and printed comparison results.

All continuation lines start with three asterisks (*******). The last two characters of a detail line appear as the first two characters of the next continuation line.

ERD Creator definitions

You can use the ERD Creator to determine which files are associated with an application.

To obtain the list of files associated with an application:

1 Type db on the Service Manager client command line. Press **Enter**.

2 Type erdcreate in the File field. Click **Search**.

3 Click **Search** again to list all the records in this table. Database Manager displays a list of Service Manager applications.

4 Double-click any application. Service Manager displays a list of files related to the application you selected.

Index

Symbols

*aapm.upgrade.purge, 108

A

acceptance tests, 100

apiserver, 113

apm.upgrade.*, 86

apm.upgrade.results, 84, 90

application key, 128

applications
 display, 20

application upgrade, options, 43

array field differences, 153

autoSubmit, 117

C

cacertpem, 113

cacerts, 117

case insensitive database systems, 40

certpem, 113

cirelationship table, 132

client, 105
 documentation, 21
 installation, 21
 upgrade, 107
 web, 105

client/server, 22

clustername, 113

commands
 *aapm.upgrade.purge, 108
 agcompare, 148
 db, 154
 scxupgrade, 101
 validity, 88

compress_soap, 117

continuation lines, 153

cstrace, 113

custom upgrade
 acceptance tests, 26
 backups, 25
 building, 105
 creating, 99
 development system, 26
 disk space, 22, 23
 files, 100
 path, 104
 production, 106, 107
 purge files, 108
 rebuilding, 108
 testing, 104, 105
 test system, 26