Peregrine

# ServiceCenter

# Upgrade Implementation Utility

## For upgrading versions A9802 and later to SC51

PART NO: SCT-5.1-ENG-01022-00233

Peregrine
SYSTEMS

If you need technical support for this product, or would like to request documentation for a product for which you are licensed, contact Peregrine Systems, Inc. Customer Support by email at support@peregrine.com.

If you have comments or suggestions about this documentation, contact Peregrine Systems, Inc. Technical Publications by email at doc_comments@peregrine.com.

This edition applies to version 5.1 of the licensed program.

Peregrine Systems, Inc.
Worldwide Corporate Headquarters
3611 Valley Centre Drive San Diego, CA 92130
Tel 800.638.5231 or 858.481.5000
Fax 858.481.1751
www.peregrine.com

# Contents

# Introducing the Upgrade Implementation Utility Guide

Welcome to the Upgrade Implementation Utility guide. This chapter provides an overview of this guide, and lists the knowledge requirements and the related ServiceCenter documentation.

This guide documents the ServiceCenter Upgrade Utility (SC Upgrade). The ServiceCenter Upgrade Utility is used to update your ServiceCenter Applications (A9802 or later) to the current release of SC51. This guide refers to the upgrade utility as SC Upgrade.

Some information about upgrading the ServiceCenter RTE (SC 5.1) is available in this guide, but more detail is given in the ServiceCenter Installation guide for your platform.

If you are running a ServiceCenter application version of A9801 or older, see the ServiceCenter 4 Upgrade Utility Guide to upgrade to a more recent version.

This guide contains the following chapters and appendices:

- *ServiceCenter Upgrade Process Overview* on page 15 — provides an overview of the upgrade process. It explains the upgrade path, and the stages of the upgrade process. It describes how the upgrade works, and how the Run-time Environment and applications are impacted by the upgrade. It describes what to consider when upgrading a customized system.
- *Outline of SC Upgrade Steps* on page 39 — provides a outline of the upgrade process, which can serve as an Upgrade Checklist.

- *Planning the ServiceCenter Upgrade* on page 29 — provides steps for planning the upgrade of your ServiceCenter system, including system and disk requirements; system knowledge requirements; the planning necessary for each stage of the upgrade; and a summary of the upgrade process.
- *Preparing the Development System* on page 55 — provides preparation information for the upgrade, including installing the SC Upgrade Utility and options for RDBMS users.
- *Upgrading Your System* on page 69 — provides step-by-step procedures for upgrading your display components and the rest of your system with the SC Upgrade Utility, discusses methods for conflict resolution, and is the starting point in the procedures for those upgrading from A9802 or later.
- *Building a Custom Upgrade* on page 127 — provides step-by-step procedures for creating a custom upgrade from your *Development* system to apply to your *Production* system.
- *Troubleshooting* on page 177 — provides solutions to problems you may encounter with the upgrade.
- *RAD Comparison Utility* on page 153 — discusses the software management tool that compares versions of the same RAD application.
- *Adding ITIL Functionality* on page 165 — provides a list of the unload files and impacted records, should you want to add ITIL functionality.
- *Using the SQL Compare Utility* on page 169 — provides instructions for using a set of applications that help to determine the database changes necessary to support an upgrade without converting SQL files back to P4.
- *Procedures You Need to Know to Run the Upgrade* on page 143 — provides instructions for procedures that you will use during the upgrade process, including loading files, stopping schedulers, and monitoring the upgrade.

# Knowledge Requirements

It is essential that you read this entire document and be familiar with the procedures before actually proceeding with an upgrade. The procedures must be followed in the sequence provided. If you do not follow this sequence, the upgrade could fail.

Only an experienced system administrator who is fully trained in ServiceCenter and familiar with the customization at your site should attempt to perform the upgrade.

The experienced system administrator has a thorough knowledge of their operating system and of the ServiceCenter P4 file system, and a thorough understanding of the base utilities used with ServiceCenter.

**Important:** If you are unsure of any procedures below; if you think you may not have the administrative experience necessary to run the upgrade; or if you are new to administering ServiceCenter, please contact Peregrine Systems Customer Support. Contact information is provided in *Troubleshooting* on page 177.

The system administrator should know:

- how the ServiceCenter file system operates.
- how the application files function.
- how to compare records within ServiceCenter.
- ServiceCenter's Rapid Application Development (RAD) environment. If you are administering a system with customized RAD applications, then knowledge of RAD, is required.

**Note:** If you are unfamiliar with RAD and are administering a system with customized RAD applications, contact the RAD programmer who maintains your system modifications, usually Peregrine Systems customer support.

The utilities used the most are:

- Database Manager
- Database Dictionary
- Display Application
- Forms Designer (GUI) or Format Manager (text mode).

If data is stored in an RDBMS:

- You should be knowledgeable about that database. If you are not knowledgeable about the database, get the assistance of your database administrator.
- You should know how the ServiceCenter file system functions with the database in which your ServiceCenter data is stored.

## Related Documentation

In addition to this guide, other ServiceCenter documents are referenced and should be available during the upgrade process:

- *System Tailoring Guide, Volumes 1, 2, and 3* — for Database Dictionary, Database Manager, Forms Designer, and the Display application.
- *Data Management and Administration Guide* — for RDBMS information, if your system is mapped to an RDBMS.
- *Request Management Guide* - for further explanation of Request Management, if used at your site.
- *Installation Guides* — Windows platforms, Unix or OS/390, depending on which platform that your ServiceCenter server is running, for the RTE upgrade
- *Release Notes* — for specific information on what is included with this release.

# Sample Screens and Examples

The sample screens and examples included in this guide are for illustration only, and may differ from those at your site.

# Contacting Education Services

Training services are available for the full spectrum of Peregrine Products including ServiceCenter.

Current details of our training services are available through the following main contacts or at:

http://www.peregrine.com/education

| | |
|---|---|
| Address: | Peregrine Systems, Inc.<br>Attn: Education Services<br>3611 Valley Centre Drive<br>San Diego, CA 92130 |
| Telephone: | +1 (858) 794-5009 |
| Fax: | +1 (858) 480-3928 |

# CHAPTER 1 ServiceCenter Upgrade Process Overview

This chapter provides an overview of the upgrade process. It explains the upgrade path, and the stages of the upgrade process. It describes how the upgrade works, and how the Run-time Environment and applications are impacted by the upgrade.

This chapter is divided into the following sections:

- *The Upgrade Path, by Version and Language* on page 16 — lists the path, materials, and documentation necessary to upgrade the various versions of ServiceCenter.
- *The Upgrade Process* on page 19 — describes the upgrade process.
- *How the Upgrade Process Works* on page 21 — discusses how the SC Upgrade Utility functions, including an explanation of digital signatures. The upgrade time frame is also discussed.
- *How the Upgrade Impacts the Run-time Environment and Applications* on page 22 — explains the Run-time Environment (RTE) and application components of the ServiceCenter system.
- *The Impact of Customization on the Upgrade Process* on page 27 — explains the conflicts that arise while running SC Upgrade if your ServiceCenter applications were modified from the original system supplied by Peregrine Systems, Inc. This section includes a list of the upgrade log files that can be used to track the process.
- *Overview of Implementing ITIL Best Practices* on page 28 — discusses implementing ITIL Best Practices on ServiceCenter.

# The Upgrade Path, by Version and Language

The upgrade path for ServiceCenter applications and Run-time Environment (RTE) depends on the software version from which you are upgrading and the destination version.

ServiceCenter applications are currently packaged in three language groupings:

- English only applications.
- English, plus French, Italian, German, and Spanish (EFIGS).
- English, plus Japanese (EJ).

The tables in the following sections define the upgrade path for each of these language groupings, as well as identify the required documentation and software.

- See *Upgrade path for systems with English-only applications* on page 17 for upgrading systems with English-only applications.
- See *Upgrade path for systems with English/French/Italian/German/Spanish (EFIGS) applications* on page 18 for upgrading systems with EFIGS applications.
- See *Upgrade path for systems with English/Japanese (EJ) applications* on page 19 for upgrading systems in Japanese.

After ServiceCenter 4.0, you should always use the most current RTE version available, regardless of the applications version you are using. Instructions for upgrading the RTE can be found in *Upgrading the Run-time Environment* on page 150 in this guide, and in the Installation guides for various platforms.

## Upgrade path for systems with English-only applications

| To upgrade from this version... | to this version: | Use this path, documentation and software |
|---|---|---|
| RTE ServiceCenter 3.0<br>Pre-A9802 applications | Any later version | Upgrade to ServiceCenter 3.0 and A9802, then upgrade to the destination version(s), as described below. |
| RTE ServiceCenter 3.x A9802 applications | 4.0.5 RTE<br>SC4.0.5 Applications | First, upgrade to the 4.0 RTE and SC4 applications, using the *SC4.0 Upgrade Utility Implementation Guide* and the 4.0/SC4 upgrade software. (Alternatively, you could upgrade the RTE to 4.0.5 directly.)<br>Then, update the RTE to 4.0.5, if you have not already done so.<br>Finally, apply the SC4.0.5 applications service pack using the *SC4.0.5 Service Pack Upgrade Utility Implementation Guide* and the application service pack software. |
| RTE ServiceCenter 4.0 SC4 applications | 4.0.5 RTE<br>SC4.0.5 Applications | Update the RTE to 4.0.5 using the 4.0.5 software, as explained in the installation documentation for your operating system.<br>Then, upgrade the applications to SC4.0.5 by applying the service pack using the *SC4.0.5 Service Pack Upgrade Utility Guide* and the application service pack software. |
| RTE ServiceCenter 4.0.5 and higher<br>SC4.0.5 applications | 4.1 RTE<br>SC4.0.5 Applications | Since ServiceCenter 4.1 is an RTE-only release, update the RTE using the 4.1 installation software and installation guide for your operating system.<br>For this RTE, the most recent applications are SC4.0.5. |
| RTE ServiceCenter 3.x or 4.x<br>A9802 or higher applications | 5.0 RTE<br>SC5 Applications | Upgrade to ServiceCenter 5.0 and SC5 using the *SC5 Upgrade Utility Implementation Guide* and the 5.0/SC5 upgrade software. |
| RTE ServiceCenter 3.x or 4.x<br>A9802 or higher applications | 5.1 RTE<br>SC5.1 Applications | Upgrade to ServiceCenter 5.1 and SC5.1 using the *SC5 Upgrade Utility Implementation Guide* and the 5.1/SC5.1 upgrade software. |

### Upgrade path for systems with English/French/Italian/German/Spanish (EFIGS) applications

| To upgrade from this version... | to this version: | Use this path, documentation and software |
|---|---|---|
| RTE ServiceCenter 3.0<br>Pre-A9802 applications | Any later version | Upgrade to ServiceCenter 3.0 and A9802, then upgrade to the destination version(s), as described below. |
| RTE ServiceCenter 3.x A9802 applications | 4.0 RTE<br>SC4.0 Applications | Not applicable - ServiceCenter 4.0 (SC4) is available in English only. See the upgrade path to ServiceCenter 4.1 (SC4.0.5). |
| RTE ServiceCenter 3.x A9802 applications | 4.1 RTE<br>SC4.0.5 Applications | Update the RTE to 4.1 using the 4.1 software, as explained in the installation documentation for your operating system.<br>Then, upgrade the applications to SC4.0.5 by applying application upgrade using the *SC4.0.5 Upgrade Utility Implementation Guide for Languages* and the applications upgrade software for EFIGS. |
| RTE ServiceCenter 4.0.7 and higher<br>SC4.0.5 applications | 4.1 RTE<br>SC4.0.5 Applications | Since ServiceCenter 4.1 is an RTE-only release, update the RTE using the 4.1 installation software and installation guide for your operating system.<br>For this RTE, the most recent applications are SC4.0.5. |
| RTE ServiceCenter 3.x or 4.x<br>A9902 or higher applications | 5.0 RTE<br>SC5 Applications | Update the RTE to 5.0 using the 5.0 software, as explained in the installation documentation for your operating system.<br>Then, upgrade the applications to SC5 by applying application upgrade using the *SC 5.0 Upgrade Utility Implementation Guide* and the applications upgrade software for EFIGS. |
| RTE ServiceCenter 3.x or 4.x<br>A9902 or higher applications | 5.1 RTE<br>SC5.1 Applications | ServiceCenter 5.1 (SC5.1) is not yet available in non-English versions. |

### Upgrade path for systems with English/Japanese (EJ) applications

| To upgrade from this version... | to this version: | Use this path, documentation and software |
|---|---|---|
| RTE ServiceCenter 3.0<br>Pre-A9802 applications | Any later version | Upgrade to ServiceCenter 3.0 and A9802, then upgrade to the destination version(s), as described below. |
| RTE ServiceCenter 3.x A9802 applications | 4.0 RTE<br>SC4.0 Applications | Not applicable - ServiceCenter 4.0 (SC4) is available in English only. See the upgrade path to ServiceCenter 4.1 (SC4.0.5). |
| RTE ServiceCenter 3.x A9802 applications | 4.1 RTE<br>SC4.0.5 Applications | Update the RTE to 4.1 using the 4.1 software, as explained in the installation documentation for your operating system.<br>Then, upgrade the applications to SC4.0.5 by applying application upgrade using the *SC4.0.5 Upgrade Utility Implementation Guide for Languages* and the applications upgrade software for EJ. |
| RTE ServiceCenter 4.0.7 and higher<br>SC4.0.5 applications | 4.1 RTE<br>SC4.0.5 Applications | Since ServiceCenter 4.1 is an RTE-only release, update the RTE using the 4.1 installation software and installation guide for your operating system.<br>For this RTE, the most recent applications are SC4.0.5. |
| RTE ServiceCenter 3.x or 4.x<br>A9902 or higher applications | 5.0 RTE<br>SC5 Applications | ServiceCenter 5.0 (SC5) is not available in Japanese. |
| RTE ServiceCenter 3.x or 4.x<br>A9902 or higher applications | 5.1 RTE<br>SC5.1 Applications | ServiceCenter 5.1 (SC5.1) is not yet available in non-English versions. |

# The Upgrade Process

The two parts of ServiceCenter, the applications and the binaries (Run-time Environment) are upgraded separately. The RTE is upgraded first, using the installation media and guides for your platform. The application upgrade is done after the RTE has been upgraded. The ServiceCenter Upgrade Utility (SC Upgrade) is used to upgrade the applications for the ServiceCenter system while minimizing the impact on that *Production* system. SC Upgrade was designed to minimize the amount of time that your production ServiceCenter system has to be down.

**Note:** Installing the SC Upgrade utility on your *Development* system is not the same as running the upgrade. The installation simply loads the files that create a Custom Upgrade which you run on your system.

Because ServiceCenter is a complex and customizable system, most upgrades do not run without conflicts the first time. For that reason, the upgrade is first run on a *Development* system to resolve conflicts and create a custom upgrade, and the custom upgrade is tested on a *Development* system before the custom upgrade can be applied to your *Production* system.

Although upgrading a ServiceCenter installation can be a lengthy process, your *Production* system need not be off-line while developing the custom upgrade you will apply to the system. One of the first stages of the upgrade process is to create a copy of your *Production* system that you will later use to develop and test custom upgrade. Because the development and testing is not done on the *Production* system, the *Production* system continues to run normally during this process.

After testing this custom upgrade, it can then be applied to your *Production* system with little interruption in service. Only minimal conflict resolution and cleanup are necessary at this stage, since your earlier work is already included in the upgrade.

Before beginning, make sure you have appropriate servers and space. You will need to keep several backups of your original production environment, as well as at least two full running systems at all times.

As well as your production environment, plan to have:

- A *Development* system to develop the custom upgrade on.
- A *Test* system to test the custom upgrade on.
- Several backups which can be used to restore to previous conditions as necessary.

**Important:** Make frequent backups. That way, if something goes wrong, you won’t have to start over completely. You can start from the last backup instead.

# How the Upgrade Process Works

The standard ServiceCenter installation is used to upgrade the ServiceCenter RTE (Run-time Environment), also known as the ServiceCenter binaries. Upgrading the RTE must be completed before the applications can be upgraded. If you have not upgraded your RTE, refer to the instructions in *Upgrading the Run-time Environment* on page 150.

The SC Upgrade is a collection of utilities that allows you to upgrade from ServiceCenter applications from a release of A9802 or later to release SC51. The task of the upgrade utility is to replace old application-related files with an updated version.

Upgrading a set of ServiceCenter applications is more complicated than it might at first appear. Because ServiceCenter applications have often been modified to meet your business needs, the application upgrade routines must not be permitted to overwrite your customized applications. SC Upgrade determines whether or not an application-related file has been customized.

If the file has not been customized, the new version is copied to the system. If a given application has been customized, SC Upgrade simply places a copy of the new application on the system to compare to the customized version. The customized files are not overwritten. Using conflict resolution, you then build a custom upgrade to update the customized applications.

## How SC Upgrade Compares New Files to Old Files

SC Upgrade uses *digital signatures* to determine if an application has been modified from the original Peregrine Systems application. Signaturing allows an entire object to be reduced to a small, digital signature.

The SC Upgrade includes the current release code and the digital signatures of every object from all ServiceCenter releases that are A9802 and later. The upgrade utility compares the signature of each object in your file system against the entire library of Peregrine Systems signatures from the previous releases. Matching signatures indicate that the object in your system has not been modified, and it can be upgraded as it is.

Signatures that do not match indicate that the original object has been modified. The new object is copied to your system, but the older object is not overwritten. The modifications are saved and are added later in the process during conflict resolution.

This approach provides an upgrade path from previous releases to the current release, and only one set of source code needs to be shipped. By comparing digital signatures, customized applications are not accidently replaced by newer, non-customized applications.

**Note:** Signatures do not check the data written to non-system files, such as Incident tickets, display options, and events. That data is preserved in your database. To learn more about resolving conflicts in the Display application, refer to *Step 2. Conflict Resolution of the Display Components* on page 105.

## How the Upgrade Impacts the Run-time Environment and Applications

SC Upgrade is an *application* upgrade utility that upgrades ServiceCenter applications, not the ServiceCenter Run-time Environment. This section explains these pieces of the system and outlines the differences between them.

You can upgrade your Run-time Environment independently of your applications. ServiceCenter allows you to run a new RTE release level with previous level applications. However, it is often impossible to run a new release of ServiceCenter applications with a previous level RTE release. Before running the SC Upgrade utility, you must complete an RTE upgrade. This procedure is provided in *Upgrading the Run-time Environment* on page 150.

Although Peregrine Systems released the SC 5.1 RTE concurrently with the SC51 application release, it is possible to use the SC 5.1 RTE with any prior application release, such as A9701 or even A9601. Updated RTE releases are designed to run with both the new applications and the older applications.

The SC applications, however, often need to be run on a binary version of at least as new as they are, so you cannot necessarily use an older RTE with a newer version of an SC application release. Peregrine Systems is constantly adding new capabilities to its RTE. As these features become available, ServiceCenter applications are enhanced to take advantage of the new RTE features. For example, the A9902 applications were developed with the ServiceCenter 3.0 Service Pack 2a RTE. The A9902 application release will not run properly under the 2.1 binary (RTE) release.

## Release Designation

Releases are designated by a three-decimal number, indicating version as follows:

- *X.0.0* - major release: Application and RTE enhancements
- *0.X.0* - minor release: RTE enhancements only, however, some minor releases may also contain Application enhancements.
- *0.0.X* - maintenance release: Bug fixes only

Refer to the Peregrine Systems, Inc., Customer Support web site (http://support.peregrine.com) for the latest compatibility matrix to see on which version of a platform the client runs.

## The Run-time Environment

The engine for the ServiceCenter system is the Run-time Environment (RTE), also referred to as the *binaries*. The RTE is a collection of core executables that run both on the server and on any client machines that require access to ServiceCenter. (OS/390 (MVS)) customers running in 3270 mode and Unix clients running the system directly off the server do not have client applications.)

The RTE is compiled and runs natively on your server or client platform, interacting with the server or client operating system. The RTE interprets the ServiceCenter applications and translates the application requests into appropriate actions on a specific platform.

# Applications

The term *applications* refers to the ServiceCenter applications and the related configuration files. Users interact with the ServiceCenter system through the applications, which dictate the manner in which ServiceCenter behaves. Incident Management, Change Management, and Inventory Management are examples of ServiceCenter applications.

Applications are stored within the ServiceCenter database file system in a series of database dictionaries (dbdicts) named *format*, *application*, *code*, and *enclapplication*. Other supporting data for these applications are stored in a variety of other database dictionaries, such as *formatctrl*, *validity*, and *environment*.

Since ServiceCenter allows you to add new fields to the dbdicts , and new dbdicts to the system, the ServiceCenter application upgrade can and often does affect the dbdicts.

If you would like a full report on what fields will be added to your dbdicts by SCUpgrade, run the SQL Compare Utility (You can do this even if you are not mapped to a RDBMS). For more information about the SQL Compare Utility, see *Using the SQL Compare Utility* on page 169.

At the most basic level, applications are simply data inside the ServiceCenter file system. SC Upgrade replaces old files with new files. Provisions are made for customized systems so that modified files are not lost.

## Application Version Numbering

Starting in ServiceCenter 4.0, applications are designated as SC*x*, where *x* is the version number. For example, SC4.

In ServiceCenter 3 and earlier versions, application releases were designated in the form *Ayyrr* where *yy* = the last two digits of the year of the release and *rr* = the release number for that year. For example, the first application release of 1999 was A9901.

## Application Files that are Upgraded

During the upgrade process, you load a file called transfer.bin. During the upgrade process, this file upgrades a series of database files for the applications. The following patch record lists the upgraded database files. You can use the Database Manager to view these files.

### Patch Record

| | |
|---|---|
| applicationfields | cm3messages |
| cm3profile<br>(Only the DEFAULT record is updated.) | cmcontrol |
| company<br>(Only the DEFAULT record is updated.) | counters |
| currency | datadict |
| datamap | dbdict |
| ddescript | displayevent |
| displayoption | displayscreen |
| environment | erddef |
| eventmap | eventregister |
| format | formatcontrol |
| globallists | help |
| info | jcl |
| joindefs | language |
| link | macrodef |
| menu | msgclass |
| notification | number |
| object | ocmoptions |
| pmenv<br>(Only the DEFAULT record is updated.) | process |

| | |
|---|---|
| querystored<br>(Querystored records whose names start with probsummary are not upgraded.) | RAD applications |
| rcenv | report |
| reportquery | scmessage |
| screlconfig | scripts<br>(Only scripts that start with pm, ocm, or cm are upgraded.) |
| slacontrol | sqlwords |
| states | subtotals |
| system events | triggers |
| tzfile | validity<br>(Validity files with names that start with prob or upgrade are not upgraded.) |

## Root Cause Analysis and Scheduled Maintenance

If you are upgrading from a version prior to ServiceCenter 4.0, two new modules were added in SC4: Root Cause Analysis and Scheduled Maintenance.

If you are upgrading a system from a release prior to SC4, these modules are installed during Phase II of the upgrade process. For more information on this process, see *Phase II: Applying the Upgrade* on page 75.

If you are upgrading an SC4 or later system, Root Cause Analysis and Scheduled Maintenance are updated with the other modules.

For details on Root Cause Analysis, see the *User's Guide* and *Application Administration Guide*.

# The Impact of Customization on the Upgrade Process

ServiceCenter is a very customizable system. Most customers have made significant changes to the functionality and behavior of their systems.

Because of these customizations, upgrading ServiceCenter is not an entirely automated process. As a system administrator, you should expect to spend a significant amount of time after the upgrade is completed to test the upgraded system and resolve any conflicts that arise.

**Important:** Do not further customize your ServiceCenter *Production*, *Test*, or *Development* systems during the upgrade process. Changes to any system could cause the custom upgrade to fail when it is applied to the *Production* system.

Customization affects the upgrade process because of the interdependency of various parts of the system. If an original Peregrine Systems application is upgraded and a second modified application on which the first is dependent is not upgraded, the two applications may not interact properly.

**Note:** In practice, most replacements of this nature continue to function perfectly. However, a small number do not. Therefore, you must test the system and ensure that it is functioning properly.

SC Upgrade treats each application as a unique entity and upgrades all the pieces that are unmodified Peregrine Systems code. It does not automatically upgrade items that have been modified from the original out-of- box system.

If SC Upgrade fails to upgrade any application because of modifications to the out-of-box system, you must make one of the following choices:

1. Use your custom version.
2. Use the new version.
3. Carry your customizations forward into the new version through conflict resolution. (See *Phase III: Conflict Resolution* on page 100.)

When making your decision, consider these points:

- It is not always desirable to overwrite your customized files with the latest version of Peregrine Systems code. For example, if you modified an application to add needed features for your site, you may not want to overwrite your modifications.

- If Peregrine Systems Customer Support helped you debug (and therefore modify) an application, you will probably want to upgrade to the latest Peregrine Systems version of the application with its fixes and new features. You can check the customer support website to confirm whether or not your SCR got into the release. See *Searching for Details of SCRs* on page 181.

# Overview of Implementing ITIL Best Practices

ServiceCenter 4.0 incorporated workflows and best practices based on the Information Technology Infrastructure Library (ITIL).

If you are upgrading applications prior to SC4, the ITIL functionality is not installed automatically with the upgrade. You can add this functionality by loading a series of files to update Incident and Change Management.

The module known as Problem Management in releases prior to SC4 was renamed Incident Management to more clearly draw its correspondence to ITIL workflows.

**Important:** Loading these files is not required. Only load them if you desire the ITIL functionality. Loading these files causes any category or group record with corresponding names to be overwritten. Existing files may be renamed to prevent this occurrence.

For a list of the files and corresponding impacted records, see *Adding ITIL Functionality* on page 165.

CHAPTER 2

# Planning the ServiceCenter Upgrade

The SC Upgrade utility allows you to implement the latest ServiceCenter application features without losing any customizations made on your system. It provides the current enhancements of the ServiceCenter applications, and includes fixes to problems present in the previous release.

Before proceeding with the upgrade, you need to plan the process for your system. This chapter provides a guide for planning your upgrade and an outline of the upgrade process.

**Important:** By careful planning, you avoid surprises in the process.

This chapter is divided into the following sections:

# Upgrade Requirements

It is essential that you read this entire document and be familiar with the procedures before actually proceeding with an upgrade. You must follow the sequence of procedures provided in order to successfully upgrade your ServiceCenter system.

Only an experienced system administrator who is fully trained in ServiceCenter and familiar with the customization at your site should attempt to perform the upgrade. For a complete explanation of the Knowledge Requirements for the ServiceCenter Upgrade, see *Knowledge Requirements* on page 11.

**Warning:** Suspend all development and tailoring activity until upgrade is complete!

## System Requirements

The following ServiceCenter system requirements must be met to perform the upgrade process:

- *Operator rights* — You must have ServiceCenter system administrator rights to complete the upgrade process. You also must have read and write access to the system on which you will run the *Development* system.
- *System version* — Your current ServiceCenter application release level must be A9802 or later. If your *Production* system is running an application version that pre-dates the A9802 release, you need to bring your system up to at least that version. These procedures are outlined in the ServiceCenter 4 *Upgrade Implementation Guide*.
- *RTE* — You must upgrade your ServiceCenter RTE to SC 5.1 before running the SC Upgrade utility. Upgrading the RTE is described in *Upgrading the Run-time Environment* on page 150.

**Important:** The ServiceCenter application upgrade should not be started until after the RTE (binary) upgrade is complete.

- *Operating system* — The operating systems on your server and client systems must meet the minimum release levels prescribed for ServiceCenter. Remember to check the requirements for all client systems that may be attached to your ServiceCenter server.
- *Database* — If ServiceCenter is operating with an RDBMS, ensure that the release level you are running is compatible with the latest release of ServiceCenter. Refer to the Peregrine Systems, Inc., Customer Support web site (http://support.peregrine.com) for the latest compatibility matrix listing operating system and RDBMS requirements.

## Shared Memory

SC Upgrade requires you to check your system's shared memory prior to starting the upgrade process

▶ Update the shared memory using the shared memory parameter, which is stored in the ServiceCenter initialization file (sc.ini or PARMS).

Make sure you allocate enough shared memory for the upgrade process. Peregrine Systems recommends at least 32 megabytes. However, if you have a large database, you may need to allocate more shared memory to accommodate it during the upgrade process.

**Note:** If you have converted your database to an RDBMS, Peregrine recommends that you set your shared memory to at least 64 megabytes.

**Important:** For OS/390 systems, the upgrade process will take as much CPU time as possible. If you are running other processes at the same time, you may wish to allocate a lower priority to the upgrade process to allow resources for the other jobs.

## Disk Space

Ensure that adequate disk space is available for both the *Development* system and the upgrade of the *Production* system. You must have enough extra space allocated to your ServiceCenter files for the information to be loaded onto your server.

**Important:** If adequate disk space is not available during the upgrade, the upgrade will fail.

The scdb.db1 file in your ServiceCenter file system increases in size up to 200 MB during the upgrade process due to the amount of new information loaded by the utility. The scdb.asc file may increase up to 16 MB.

This extra space can be partially reclaimed at the end of the upgrade process by running the LFMAP utility on your scdb.db1 file (pool 3). See the *P4 File System Utility (SCDBUTIL)* section in the *Database Management and Administration Guide* for instructions on running LFMAP.

- **For the system as shipped:**

  **10 KB per application + 2 KB per form (format) + 40 MB**

  There are 2,000 applications in ServiceCenter using 20 MB of space. Approximately 27,000 forms make up another 45 MB.

  This formula should be adequate, unless your system has large, complex forms and/or applications. Or, you have added substantially to any of the files listed in the patch record (for example, formatctrl, knowledge, menu, or link). If that is the case, use the option for customized systems.

- For customized systems:
  - 150 MB of free drive space for moderate customization.
  - 200 MB of free drive space for heavy customization.

Allocation of disk space for the custom upgrade is discussed in *Preparing to Build the Custom Upgrade* on page 128.

## Customization, RDBMS, and RAD Applications

The upgrade process impacts multiple aspects of the ServiceCenter system. Besides upgrading the standard ServiceCenter applications, an upgrade can impact the RDBMS on which ServiceCenter is running as well as any customized files or RAD applications.

If your ServiceCenter system has been customized, the upgrade process takes those modifications into account. Files impacted by the upgrade are listed in *How the Upgrade Impacts the Run-time Environment and Applications* on page 22.

**Note:** A list of customized files is useful for conflict resolution and building the custom upgrade.

If ServiceCenter is mapped to an RDBMS, certain mappings and tables could be impacted. Contact the database administrator for assistance, and to discuss the impact on the RDBMS.

If RAD applications have been modified, a comparison may be needed between the existing application and the update. This procedure is covered in *RAD Comparison Utility* on page 153. Also, the RAD programmer may be able to supply information about which RAD applications have been modified.

## Backups

Before beginning, make sure you have appropriate servers and space. You will need to keep several backups of your original production environment, as well as at least two full running systems at all times.

As well as your production environment, plan to have:

- A *Development* system to develop the custom upgrade on.
- A *Test* system to test the custom upgrade on.
- Several backups which can be used to restore to previous conditions as necessary.

**Important:** Make frequent backups. That way, if something goes wrong, you won't have to start over completely. You can start from the last backup instead.

## Checking for Known Issues

Known issues are documented in knowledge entries accessible through Peregrine's CenterPoint Web site. Check the Customer Support Website for Upgrade issues before beginning the upgrade process.

### Searching for Known Upgrade Issues

You can view a list of known issues by searching the knowledge base on Peregrine's CenterPoint Web site at:

http://support.peregrine.com/

### After logging in with your login and password:

1. Select **Go** for **CenterPoint**.
2. Select **ServiceCenter** from **My Products** at the top of the page.
3. From under **Knowledge Search** on the left, select **Go** for **Advanced Search**.
4. Enter `5.1 Upgrade` as the **Search Criteria**.

   You can further restrict the search by entering additional search criteria if desired.

   If you know the Document ID for a specific knowledge entry (for example, one shown in the following table), you can use that as search criteria to retrieve a specific article. Type in the Document ID enclosed in double quotes, for example, “S7F-W16-8GSN”.
5. Select **ServiceCenter** (**KB & Documentation**) as the product in number 1.
6. Select **Search the Knowledgebase** in number 2.
7. Click Search. The list of entries that match the criteria you selected will be generated.
8. Click on a knowledge entry to open it.

# Planning the Development Environment

The *Development* system is the system on which you will develop your custom upgrade. Before you can develop a custom upgrade, you must set up your development environment:

- Locate a machine on which you can create a *Development* system.
- Check for adequate disk space for the *Development* system, a backup of the *Development* system, and the installation of the SC Upgrade utility. For more information, see *Disk Space* on page 31.
- Ensure that the operating system version is the same for the *Development* system and the *Production* system.

## Set Up the Development Environment

After you have secured a *Development* system, you will need to create a backup of the *Production* system from which you will build the custom upgrade. The custom upgrade is used in the last stage of the upgrade process to upgrade the *Production* system.

After you have made a backup of the *Production* system, do not make any modifications to the *Production* system. Any changes to the applications made after that point will not be reflected in the *Development* system and will cause a conflict when the custom upgrade is applied to the *Production* system.

**Warning:** If a modification must be made to the *Production* system after the *Development* system is created, a new *Development* system must be created, and the upgrade process must be restarted.

### Upgrade the RTE

Before the applications can be upgraded, the RTE must be brought up to the current version. The RTE is upgraded from the standard ServiceCenter installation media: a CD for Windows operating systems and Unix; a cartridge for OS/390 (MVS).

If the RTE upgrade works properly and passes all tests, copy the production data to this system and use it as your *Production* system. Having the RTE upgraded and running while you develop the custom application upgrade lessens the time your system will have to be down for the application upgrade. It also allows time for any RTE issues to come up, and ensures that you are only changing one thing at a time.

# Developing the Custom Upgrade

SC Upgrade replaces ServiceCenter application-related files from previous versions with the updated files for the new version. Because you can tailor ServiceCenter to meet your business needs, conflicts can arise when trying to replace the old files used with customized applications.

## Upgrade the Development Environment

To begin the upgrade process, load a backup of the *Production* system onto the *Development* system. Use *copies* of this *Development* system to apply the upgrade, resolve conflicts, and build a custom upgrade. The custom upgrade is tested against a fresh copy of the *Production* system (the *Test* system).

Be sure to use a *copy* of your *Development* system during the development and testing processes. By using a copy of the *Development* system, should any problems occur, you will not have to make another backup.

## Resolve Conflicts

Conflicts may need to be resolved in the display components, the Database Dictionary, and the applications. While doing conflict resolution, keep copious notes and use Revision Tracking to make unload archives. Make frequent backups. A backup should be made after each major change. For more information on resolving conflicts, see *Phase III: Conflict Resolution* on page 100.

**Important:** Conflict resolution is the most important part of the Service Center Applications upgrade. Without it, your upgrade will not work properly.

## Create a Custom Upgrade

After you have resolved the conflicts, you will create the custom upgrade with the copy of the *Development* system that you have been using. The custom upgrade is later used to upgrade the *Production* system.

Before creating the custom upgrade, you must create a new directory to store the custom upgrade. After that destination directory has been created, ensure that the ServiceCenter *Development* system has *read-write* access.

After the upgrade is complete, test it thoroughly. See *Testing the Custom Upgrade* on page 36.

# Testing the Custom Upgrade

Before the custom upgrade can be applied to the *Production* system, it must be tested. You will use a copy of the *Production* system for this purpose.

**Important:** Test all features that your users will utilize.

## Set Up the Test Environment

Before you can test a custom upgrade, you must set up your testing environment. This is the same process as setting up the development environment. The *Test* system can be on the same machine as the *Production* system, if there is enough room. For more information on setting up the *Test* environment, see *Planning the Development Environment* on page 34 and *Creating a Development or Test System* on page 56.

## Test the System

Test the Custom Upgrade by applying it to the Test system. If there are any problems, with the upgrade process, go back to *Resolve Conflicts* on page 36 and repeat the conflict resolution process until the custom upgrade works properly. For more information, see *Testing Your Custom Upgrade* on page 137.

## Become Familiar with New Features

Once you have successfully applied the custom upgrade to the *Development* system, familiarize yourself with the new features in the applications. If you are unfamiliar with a feature, refer to the appropriate ServiceCenter documentation.

For a list of enhancements and new features, refer to the Release Notes shipped with this product. You can also get the Release Notes and a list of resolved issues from the Peregrine Systems Customer Support web site at http://support.peregrine.com.

# Applying the Custom Upgrade

When the custom upgrade works properly, and passes all tests, apply the custom upgrade to your *Production* system using the ServiceCenter Upgrade utility, and the instructions in this guide. See *Upgrading Your Production System* on page 137.

## Plan the Implementation of the Custom Upgrade on Your Production System

To upgrade your *Production* system, you must consider two items:

- Training your users on new features.
- Applying the upgrade to the *Production* system.

## Train Users on Updated Applications

Before the upgraded system is brought into use, users need to be trained on any the new features.

For a listing of ServiceCenter courses available, go to: http://www.peregrine.com/, and select Education.

## Apply the Upgrade to Your Production System

Most of the work in the upgrade process is done in the development phase on the *Development* system. Once that work is done, the process of applying the custom upgrade to the Production system is relatively simple.

**To upgrade the production system:**

1. Complete testing of the custom upgrade. (*Planning the Development Environment* on page 34.)
2. Complete training of the users. (See *Train Users on Updated Applications* on page 38.)
3. Check the disk space availability on the production server.
4. Plan the shutdown of the *Production* system.
5. Advise the users.
6. Shut down the *Production* system.
7. Apply the custom upgrade.
8. Restart the server.
9. Advise the users.

# Adding ITIL functionality

ITIL functionality was added to ServiceCenter in version 4. If you are upgrading from a version earlier than ServiceCenter 4 and want to add ITIL functionality to your system, refer to *Adding ITIL Functionality* on page 165. This functionality is optional and not part of the upgrade process itself.

# 3 Outline of SC Upgrade Steps

CHAPTER

This chapter provides an outline of the upgrade process. This outline can serve as an Upgrade Checklist.

It has been divided into the following sections:

- *Planning the Upgrade* on page 40
- *Preparing for the Upgrade* on page 40
- *Developing the Custom Upgrade* on page 42
- *Testing the Custom Upgrade* on page 47
- *Applying the Custom Upgrade* on page 53
- *Adding ITIL functionality* on page 54

# Planning the Upgrade

By planning your upgrade, you avoid surprises in the process. This topic is covered in detail in *Planning the ServiceCenter Upgrade* on page 29.

# Preparing for the Upgrade

Before beginning this upgrade you must be running at least A9802. If your system is preA9802, upgrade to A9802 before proceeding. See *System Requirements* on page 30.

**Warning:** Suspend all development and tailoring activity until upgrade is complete!

## Setting Up the *Development* Environment

Before you can develop a custom upgrade, you must set up your development environment. For more information on setting up the *Development* environment, see *Creating a Development or Test System* on page 56.

**To set up the Development environment:**

1. Identify a server to use for the *Development* environment. See *Planning the Development Environment* on page 34.

   **Note:** Do not use the production server. Do use the same operating system and database environment.

2. Insure adequate memory is available. See *Shared Memory* on page 31.
   - Insure adequate disk space is available and accessible. See *Disk Space* on page 31.
   - Account for SQL tables if applicable. See *Customization, RDBMS, and RAD Applications* on page 32, and *Preparing Systems Mapped to RDBMS* on page 66.
   - Make sure that there is enough space for frequent backups. See *Backups* on page 33.

3. Install *SC5.latest* on the *Development* system.

4 Make a backup of the ServiceCenter system in your production environment to create an original_backup archive, then move the archive to the backups folder. See *Creating a Development or Test System* on page 56.

5 Install a copy of the original_backup on the *Development* system. See *Creating a Development or Test System* on page 56. (Copy scdb.* and ir.* files from the Data folder in the Production system to the Data folder in the *Development* system, overwriting existing files.)

6 Add a new folder to the primary folder in the *Development* system (on the same level as RUN and Bitmaps) called *Upgrade*. See *Installing the SC Upgrade Utility* on page 60.

7 Add a new folder to the primary folder in the *Development* system (on the same level as RUN and Bitmaps) called *Backups*.

8 Add a new folder to the primary folder in the *Development* system (on the same level as RUN and Bitmaps) called *Customupgrade*.

9 Copy the files from the Application Upgrade CD to the *Upgrade* folder. See *Installing the SC Upgrade Utility* on page 60.

10 Provide for connectivity to all ServiceCenter interfaces (for conflict resolution and preliminary testing phase). See the *Client/Server Installation Guide* for your platform.

11 If mapped to an RDBMS, create a duplicate database environment on the development server. See *Database Management and Administration.*

12 Upgrade the *Development* system RTE to ServiceCenter 5.1 (SC 5.1) using the ServiceCenter installation media following the instructions given in the ServiceCenter Installation Guide for your platform. *Upgrading the Run-time Environment* on page 150 also provides these instructions.

13 Test the system and verify that all ServiceCenter features your company uses are functioning properly. If there are any problems, contact customer support. See the ServiceCenter 5.1 Release Notes for new functionality you may wish to use.

14 If the RTE upgrade works properly and passes all tests, copy the production data to this system and use it as your *Production* system.

   a Bring down the ServiceCenter *Production* system.

   b Make a *full* backup of the ServiceCenter system in your *Production* system. See *Making a Backup of the Production System* on page 58.

   c Copy scdb.* and ir.* files from the Data folder in the *Development* environment, overwriting the files in the *Production* system.

**Note:** You can also create a separate folder for the OUT OF BOX data and move the scdb.* files to that folder. If you overwrite them and need to refer to them during the upgrade, they can be copied from the WIN\DATA folder on the SC 5.1 Upgrade CD. Be sure to remove the **Read Only** property.

d Restart the *Production* system.

**Note:** Having the RTE upgraded and running while you develop the custom application upgrade lessens the time your system will have to be down for the application upgrade. It also allows time for any RTE issues to come up, and ensures that you are only changing one thing at a time.

15 Make a *full* backup of the ServiceCenter system in your *Development* environment, called RTE_Backup, then move the archive to the *Backups* folder. See *Making a Backup of the Production System* on page 58.

# Developing the Custom Upgrade

SC Upgrade replaces ServiceCenter application-related files from previous versions with updated files. Because you can tailor ServiceCenter to meet your business needs, conflicts can arise when trying to replace the old files used with customized applications.

## Upgrading the *Development* Environment

During this stage, you apply the upgrade to your *Development* system. SC Upgrade detects where conflicts occur during the upgrade. Applying the upgrade starts with *Phase I: The Preliminary Steps* on page 71.

### To upgrade the *Development* system applications:

1 Modify the sc.cfg file in the *Development* environment to “comment-out” the system.start entry. See *System Preparation* on page 71.

2 Analyze and clean up the *Development* system. See *System Preparation* on page 71.

a Run LFSCAN and check output for errors. Use Scan & fix to correct errors, or contact Customer Support for assistance. See *System Preparation* on page 71.

b Extract file sizing information from the LFSCAN output and import it to Excel or Access.

**Note:** The sizing information is found in the LFSCAN output, starting with this Title line:

```
--------Index---------- -----Data----
```

See *System Preparation* on page 71.

c Start the SC 5.1 server console. See *System Preparation* on page 71.

**Note:** To ensure that only one process is started, check the sc.cfg file in the ServiceCenter RUN folder to make sure that the system.start entry has been commented out.

d Start the SC 5.1 express client. See *System Preparation* on page 71.

e Log in as a SysAdmin user. See *System Preparation* on page 71.

f Make sure the **Client Side Load/Unload** feature is disabled. See *System Preparation* on page 71.

g P4 users - analyze file sizes and allocate data to new pools as necessary. See step b on page 42, in this section, and the P4 Troubleshooting chapter of Database Management and Administration.

**Note:** If new pools are configured and data is moved, backup after this step.

h SQL users - make sure the dbdicts for cm3r, cm3rpage, cm3t, and cm3tpage are assigned to pools with sufficient space. See the P4 Troubleshooting chapter of *Database Management and Administration*.

**Note:** If pools are reassigned, backup P4 files after this step.

i Reset transient data:

- msglog
- syslog
- mail
- eventout
- eventin
- devaudit

(See *System Preparation* on page 71.)

j Remove any records left over from previous upgrades (primary key starts with NEW* or OLD*).

k Correct the data type differences as necessary, updating data where types have changed. See *Database Dictionary Changes (for pre-A9901 systems only)* on page 79.

3 If your application level is A9802, add Multilingual support, and add RDBMS support if necessary. See *Version A9802 Preparation* on page 72.

- Load **upglang.unl** from the *Upgrade* folder. See *Version A9802 Preparation* on page 72.
- Execute **apm.upgrade.language** from the ServiceCenter command line. See *Version A9802 Preparation* on page 72.
- Check to be certain that the **format** dbdict assigns "syslanguage" to field #7 and help assigns "syslanguage" to field #23.
- If you are upgrading ServiceCenter 3 and use DB2Universal as your database, remove PRGNDB from the Table and Index spaces on **sql.options**. See *RDBMS Support* on page 74.
- Backup the system to **data_after_language_upgrade** and copy the archive to the *Backups* folder. See *Making a Backup of the Production System* on page 58.

4 Run LFSCAN again, correct errors using LFSCAN and fix, then run LFMAP. See *System Preparation* on page 71.

5 Compress or zip the files to create a **baseline_after_cleanup** backup, then move it to the *Backups* folder. See *Making a Backup of the Production System* on page 58.

**Important:** This is the backup you will use if you must reapply the SC51 upgrade from the beginning.

6 Start the SC 5.1 console. See *System Preparation* on page 71.

**Note:** To ensure that only one process is started, check the **sc.cfg** file in the ServiceCenter RUN folder to make sure that the system.start entry has been commented out.

7 Start a SC 5.1 express client. See *System Preparation* on page 71.

8 Log in as a SysAdmin user. See *System Preparation* on page 71.

9 Make sure the **Client Side Load/Unload** feature is disabled. See *System Preparation* on page 71, and *Step 3. Conflict Resolution of the Database Dictionaries* on page 110.

10 Load **preupg.bin** from the *Customupgrade* directory (created in step 9 on page 41). See *Step 1: Loading preupg.bin and transfer.bin* on page 77.

11 On the ServiceCenter command line, enter load transfer. See *Step 1: Loading preupg.bin and transfer.bin* on page 77.

**Warning:** Do not use Database Manager to load `transfer.bin`

12 You can reduce the amount of time it takes to run the upgrade by removing the IR keys from the `probsummary`, `cm3r`, `cm3rpage`, `cm3t` and `cm3tpage` files before applying the upgrade. See *System Preparation* on page 71.

**Note:** All of the records in these files will be modified during the upgrade. If you are mapped to SQL the upgrade may create new temporary IR files, which can prolong the upgrade time.

13 Run the SC Upgrade Utility. On the ServiceCenter command line, enter `SC51upgrade`. See *Upgrading Your System* on page 69 and *Running the Application Upgrade* on page 85. (The SC Upgrade Utility was installed in *Setting Up the Development Environment*, step 9 on page 41.)

**Important:** If you encounter problems that cause the upgrade process to *stop*, contact customer support immediately.

14 If using Request Management on a pre-SC4 system, set up Stockrooms. See *Creating Stockrooms for Multiple Locations (for pre-SC4 systems only)* on page 81.

**Note:** This step is optional. If you do not perform this step, a stockroom record will be added for each record in your location table.

15 Compress or zip the files to create a `data_after_adding_stockrooms` backup, then move it to the *Backups* folder. See *Making a Backup of the Production System* on page 58.

16 Select **SERVICE PACK** from the Upgrade menu. See *Running the Application Upgrade* on page 85.

17 Select **Apply an Upgrade** from the Upgrade Utility. See *Running the Application Upgrade* on page 85.

18 Answer the Wizard questions. See *Using the SC Upgrade Wizard* on page 85.

**Important:** Be sure to specify the termination character at the end of the upgrade path name.

19 Verify the upgrade information, then start the upgrade. See *Using the SC Upgrade Wizard* on page 85.

20 If a message appears after the dbdict portion has completed saying that some of the dbdicts could not be upgraded, check the except.log file in the upgrade folder. See *Tracking the Upgrade Process* on page 150.

- If exceptions are in system tables, do not continue the upgrade until the data type mismatches have been resolved. See *Database Dictionary Changes (for pre-A9901 systems only)* on page 79.
- If exceptions are found in probsummary, problem or cm3r, cm3rpage,cm3t or cm3tpage files, you should repair them before continuing with the upgrade. See *Tracking the Upgrade Process* on page 150, and step 13 on page 93.

21 Once the upgrade completes (the background processes are finished), and if you have removed them, (step 12 on page 45) add the IR keys back to probsummary and cm3* files. See the IR Expert chapter of *Database Management and Administration*.

22 If you are upgrading from a version earlier than SC4, regen all other IR keys. See the IR Expert chapter of *Database Management and Administration*.

## Resolving Conflicts

Conflicts may need to be resolved in the display components, the Database Dictionary, and the applications. During this stage, potential conflicts are resolved, allowing your *Production* system to be safely upgraded, without losing the customizations that you have implemented.

**Important:** Conflict resolution is the most important part of the Service Center Applications upgrade. Without it, your upgrade will not work properly.

While doing conflict resolution:

- Keep copious notes and use Revision Tracking to make unload archives. You will need these later!
- Make frequent backups during the conflict resolution process. A backup should be made after each major change. See *Making a Backup of the Production System* on page 58.

For more information on resolving conflicts, see *Phase III: Conflict Resolution* on page 100.

### To resolve the conflicts:

1 Follow the procedures in:
   - *Step 1: Running Post Upgrade Reports* on page 102.
   - *Step 2. Conflict Resolution of the Display Components* on page 105
   - *Step 3. Conflict Resolution of the Database Dictionaries* on page 110
   - *Step 4: Conflict Resolution of the Data* on page 111

2 When Conflict Resolution is complete, test the upgraded system and verify that it is functioning properly. If there are problems, Contact Peregrine Customer Support.

3 Once the system has been thoroughly tested, make a backup that contains the Data directory, and call it the `Upgraded` backup. Copy the archive to the backups folder. See *Making a Backup of the Production System* on page 58.

## Creating a Custom Upgrade

After you have resolved the conflicts, you will create the custom upgrade with the copy of the *Development* system that you have been using. The custom upgrade is later used to upgrade the *Production* system.

Before creating the custom upgrade, you must create a new directory to store the custom upgrade. After that destination directory has been created, ensure that the ServiceCenter *Development* system has *read-write* access.

### To build a Custom Upgrade:

1 Use the ServiceCenter Upgrade utility and follow the instructions *Building a Custom Upgrade* on page 127.

2 Copy the Upgrade files to the *CustomUpgrade* folder.

3 Create a backup that includes both the data directory and the directory containing the custom upgrade files. Call it the `CustomUppgrade_backup` Copy the archive to the *Backups* folder.

**Note:** The `transfer.bin` will be up to 50% larger than the one on the upgrade CD, and the `upgrade.dta` file may more than double in size.

## Testing the Custom Upgrade

Before the custom upgrade can be applied to the *Production* system, it must be tested. Run your tests on a fresh backup of the *Production* system.

## Setting Up the *Test* Environment

Before you can test a custom upgrade, you must set up your testing environment. This is the same process as setting up the *Development* environment. The *Test* system can be on the same machine as the *Production* system, if there is enough room. For more information on setting up the *Development* environment, see *Setting Up the Development Environment* on page 40 and *Creating a Development or Test System* on page 56.

### To set up the *Test* environment:

1 Identify a server to use for the *Test* environment. See *Planning the Development Environment* on page 34.

**Note:** Do not use the production server. Do use the same operating system and database environment.

2 Insure adequate memory is available. See *Shared Memory* on page 31.

- Insure adequate disk space is available and accessible. See *Disk Space* on page 31.
- Account for SQL tables if applicable. See *Customization, RDBMS, and RAD Applications* on page 32, and *Preparing Systems Mapped to RDBMS* on page 66.
- Make sure that there is enough space for frequent backups. See *Backups* on page 33.

3 Install SC5.latest on the Test system.

4 Make a backup of the ServiceCenter system in your production environment to create an original_backup archive, then move the archive to the backups folder. See *Creating a Development or Test System* on page 56.

5 Install a copy of the original_backup on the *Test* system. See *Creating a Development or Test System* on page 56. (Copy scdb.* and ir.* files from the Data folder in the *Production* system to the Data folder in the *Test* system overwriting existing files.)

6 Add a new folder to the primary folder in the *Test* system (on the same level as RUN and Bitmaps) called *Upgrade*. See *Installing the SC Upgrade Utility* on page 60.

7 Add a new folder to the primary folder in the *Test* system (on the same level as RUN and Bitmaps) called *Backups*.

8 Add a new folder to the primary folder in the *Test* system (on the same level as RUN and Bitmaps) called *Customupgrade*.

9 Copy the files from the *CustomUpgrade* folder on the *Production* system. to the *CustomUpgrade* folder on the Test system. See *Creating a Custom Upgrade* on page 47.

10 Upgrade the *Test* system RTE to ServiceCenter 5.1 (SC 5.1) using the ServiceCenter installation media following the instructions given in the ServiceCenter Installation Guide for your platform. *Upgrading the Run-time Environment* on page 150 also provides these instructions.

11 Provide for connectivity to all ServiceCenter interfaces (for conflict resolution and preliminary testing phase). See the *Client/Server Installation Guide* for your platform.

12 If mapped to an RDBMS, create a duplicate database environment on the development server. See *Database Management and Administration.*

13 Test the system and verify that all ServiceCenter features your company uses are functioning properly. If there are any problems, contact customer support. See the ServiceCenter 5.1 Release Notes for new functionality you may wish to use.

14 Make a *full* backup of the ServiceCenter system in your *Test* environment, called `RTE_Backup`, then move the archive to the *Backups* folder. See *Making a Backup of the Production System* on page 58.

## Applying the Custom Upgrade to the Test System

**To apply the custom upgrade to the test system:**

1 Modify the `sc.cfg` file in the *Test* environment to “comment-out” the `system.start` entry. See *System Preparation* on page 71.

2 Analyze and clean up the *Test* system. See *System Preparation* on page 71.

a Run `LFSCAN` and check output for errors. Use `Scan & fix` to correct errors, or contact Customer Support for assistance. See *System Preparation* on page 71.

b Extract file sizing information from the `LFSCAN` output and import it to Excel or Access.

**Note:** The sizing information is found in the `LFSCAN` output, starting with this Title line:

```
--------Index---------- -----Data----
```

See *System Preparation* on page 71.

c Start the SC 5.1 server console. See *System Preparation* on page 71.

**Note:** To ensure that only one process is started, check the sc.cfg file in the ServiceCenter RUN folder to make sure that the system.start entry has been commented out.

**d** Start the SC 5.1 express client. See *System Preparation* on page 71.

**e** Log in as a SysAdmin user. See *System Preparation* on page 71.

**f** Make sure the **Client Side Load/Unload** feature is disabled. See *System Preparation* on page 71.

**g** P4 users - analyze file sizes and allocate data to new pools as necessary. See step b on page 42, in this section, and the P4 Troubleshooting chapter of Database Management and Administration.

**Note:** If new pools are configured and data is moved, backup after this step.

**h** SQL users - make sure the dbdicts for cm3r, cm3rpage, cm3t, and cm3tpage are assigned to pools with sufficient space. See the P4 Troubleshooting chapter of *Database Management and Administration*.

**Note:** If pools are reassigned, backup P4 files after this step.

**i** Reset transient data:

- msglog
- syslog
- mail
- eventout
- eventin
- devaudit

(See *System Preparation* on page 71.)

**j** Remove any records left over from previous upgrades (primary key starts with NEW* or OLD*).

**k** Correct the data type differences as necessary, updating data where types have changed. See *Database Dictionary Changes (for pre-A9901 systems only)* on page 79.

**3** If your application level is A9802, add Multilingual support, and add RDBMS support if necessary. See *Version A9802 Preparation* on page 72.

- Load upglang.unl from the *Upgrade* folder. See *Version A9802 Preparation* on page 72.

- Execute apm.upgrade.language from the ServiceCenter command line. See *Version A9802 Preparation* on page 72.
- Check to be certain that the format dbdict assigns "syslanguage" to field #7 and help assigns "syslanguage" to field #23.
- If you are upgrading ServiceCenter 3 and use DB2Universal as your database, remove PRGNDB from the Table and Index spaces on sql.options. See *RDBMS Support* on page 74.
- Backup the system to data_after_language_upgrade and copy the archive to the *Backups* folder. See *Making a Backup of the Production System* on page 58.

4. Run LFSCAN again, correct errors using LFSCAN and fix, then run LFMAP. See *System Preparation* on page 71.
5. Compress or zip the files to create a baseline_after_cleanup backup, then move it to the *Backups* folder. See *Making a Backup of the Production System* on page 58.

**Important:** This is the backup you will use if you must reapply the SC51 upgrade from the beginning.

6. Start the SC 5.1 console. See *System Preparation* on page 71.

**Note:** To ensure that only one process is started, check the sc.cfg file in the ServiceCenter RUN folder to make sure system.start entry has been commented out.

7. Start a SC 5.1 express client. See *System Preparation* on page 71.
8. Log in as a SysAdmin user. See *System Preparation* on page 71.
9. Make sure the **Client Side Load/Unload** feature is disabled. See *System Preparation* on page 71, and *Step 3. Conflict Resolution of the Database Dictionaries* on page 110.
10. Load preupg.bin from the *Customupgrade* directory (created in step 9 on page 41). See *Step 1: Loading preupg.bin and transfer.bin* on page 77.
    - On the ServiceCenter command line, enter load transfer. See *Step 1: Loading preupg.bin and transfer.bin* on page 77.

**Warning:** Do not use Database Manager to load transfer.bin a You can reduce the amount of time it takes to run the upgrade by removing the IR keys from the probsummary, cm3r, cm3rpage, cm3t and cm3tpage files before applying the upgrade. See *System Preparation* on page 71.

**Note:** All of the records in these files will be modified during the upgrade. If you are mapped to SQL the upgrade may create new temporary IR files, which can prolong the upgrade time.

11 Run the SC Upgrade Utility. On the ServiceCenter command line, enter `SC51upgrade`. See *Upgrading Your System* on page 69 and *Running the Application Upgrade* on page 85. (The SC Upgrade Utility was installed in *Setting Up the Development Environment*, step 9 on page 41.)

**Important:** If you encounter problems that cause the upgrade process to *stop*, contact customer support immediately.

12 If using Request Management on a pre-SC4 system, set up Stockrooms. See *Creating Stockrooms for Multiple Locations (for pre-SC4 systems only)* on page 81.

**Note:** This step is optional. If you do not perform this step, a stockroom record will be added for each record in your location table.

13 Compress or zip the files to create a data_after_adding_stockrooms backup, then move it to the *Backups* folder. See *Making a Backup of the Production System* on page 58.

14 Select **SERVICE PACK** from the Upgrade menu. See *Running the Application Upgrade* on page 85.

15 Select **Apply an Upgrade** from the Upgrade Utility. See *Running the Application Upgrade* on page 85.

16 Answer the Wizard questions. See *Using the SC Upgrade Wizard* on page 85.

**Important:** Be sure to refer to the *CustomUpgrade* folder, and to specify the termination character at the end of the upgrade path name. You should select **Replace**.

17 Verify the upgrade information, then start the upgrade. See *Using the SC Upgrade Wizard* on page 85.

18 Once the upgrade completes (the background processes are finished), and if you have removed them, (step a on page 52) add the IR keys back to `probsummary` and `cm3*` files. See the IR Expert chapter of *Database Management and Administration*.

19 If you are upgrading from a version earlier than SC4, regen all other IR keys. See the IR Expert chapter of *Database Management and Administration*.

## Testing the Upgraded System

After the upgrade is complete, test it thoroughly. If there are any problems, go back to *Resolving Conflicts* on page 46 and repeat the conflict resolution process until the custom upgrade works properly. For more information, see *Testing Your Custom Upgrade* on page 137.

Once you have successfully applied the custom upgrade to the *Development* system, familiarize yourself with the new features in the applications. Refer to the ServiceCenter Release Notes for a list of the new features. If you are unfamiliar with a feature, refer to the appropriate ServiceCenter documentation.

**Important:** Test all features that your users will utilize.

# Applying the Custom Upgrade

When the custom upgrade works properly, and passes all tests, apply the custom upgrade to your *Production* system using the ServiceCenter Upgrade utility, and the instructions in this guide. See *Upgrading Your Production System* on page 137.

## Planning the Implementation of the Custom Upgrade on Your Production System

To upgrade your *Production* system, you must consider two items:

- Training your users on new features.
- Applying the upgrade to the *Production* system.

## Training Users on Updated Applications

Before implementing the upgraded system into production, users need to be trained on any new features that they might be using. If you are unsure of where new features are documented, refer to the ServiceCenter Release Notes.

For a listing of ServiceCenter courses available, go to: http://www.peregrine.com/, and select Education.

## Applying the Upgrade to Your Production System

Most of the work in the upgrade process is done in the development phase on the *Development* system. Once that work is done, the process of applying the custom upgrade to the Production system is relatively simple.

**To upgrade the production system:**

1. Complete testing of the custom upgrade. (See *Planning the Development Environment* on page 34.)
2. Complete training of the users. (See *Training Users on Updated Applications* on page 54.)
3. Check the disk space availability on the production server.
4. Plan the shutdown of the *Production* system.
5. Advise the users.
6. Shut down the *Production* system.
7. Apply the custom upgrade.
8. Restart the server.
9. Advise the users.

# Adding ITIL functionality

ITIL functionality was added to ServiceCenter in version 4. If you are upgrading from a version earlier than ServiceCenter 4 and want to add ITIL functionality to your system, refer to *Adding ITIL Functionality* on page 165. This functionality is optional and not part of the upgrade process itself.

# 4 Preparing the Development System

CHAPTER

After you have become familiar with the upgrade process and planned your course of action, you are ready to begin the upgrade process. An overview of the upgrade and planning components are discussed in the first three chapters of this book.

**Note:** Read this entire manual before attempting the upgrade. All steps must be performed in the sequence provided in this manual.

This chapter explains how to create the systems on which you will develop and test the custom upgrade application, and how to install the Upgrade Utility on them.

It has been divided into the following sections:

- *Creating a Development or Test System* on page 56
- *Making a Backup of the Production System* on page 58
- *Creating the New System* on page 60
- *Installing the SC Upgrade Utility* on page 60
- *The SC Upgrade Application Files* on page 65
- *Preparing Systems Mapped to RDBMS* on page 66

# Creating a Development or Test System

You will need to make a copies of your *Production* system to act as *Development* and *Test* systems during the upgrade process. This section provides instructions for building the *Development* and *Test* systems.

### To create a Development or Test system:

1 Make a copy of your *Production* system. See *Making a Backup of the Production System* on page 58.

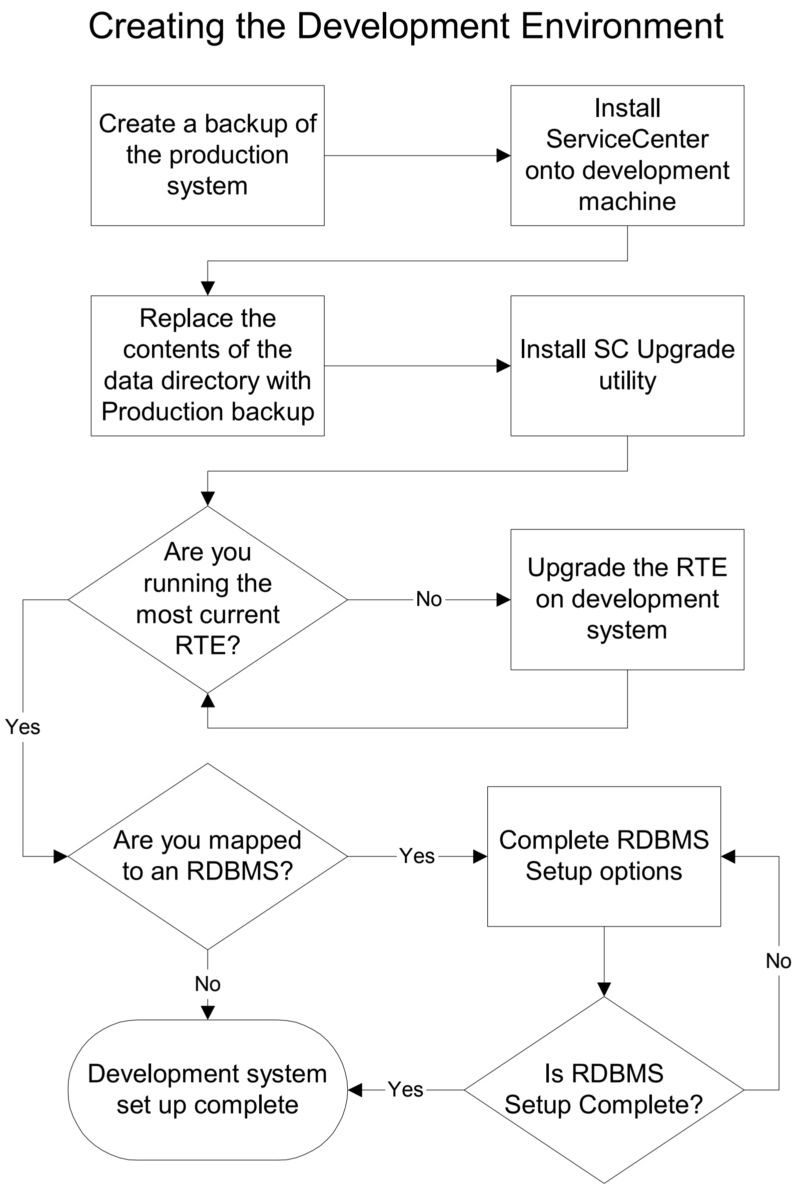
2 Create the new system. See *Creating the New System* on page 60.

3 Install the Upgrade Utility. See *Installing the SC Upgrade Utility* on page 60.

4 If you have any files mapped to an RDBMS, make a copy of that RDBMS. Contact your database administrator to make this backup copy. See *Preparing Systems Mapped to RDBMS* on page 66.

**Important:** DO NOT make changes to your *Production* system (Format Control, Forms Designer, etc.) after copying it for the upgrade process. If it becomes absolutely necessary to make a change, such as to fix a problem, you must make another copy of the *Production* system to use as the *Development* system.

## Flow Chart


Creating the Development Environment
Create a backup of the production system
Install ServiceCenter onto development machine
Replace the contents of the data directory with Production backup
Install SC Upgrade utility
Are you running the most current RTE?
No
Upgrade the RTE on development system
Yes
Are you mapped to an RDBMS?
Yes
Complete RDBMS Setup options
No
No
Development system set up complete
Yes
Is RDBMS Setup Complete?

# Making a Backup of the Production System

There are two methods of doing the backup:

- *Cold Backup* on page 58
- *Hot Backup* on page 59

These are discussed in detail in the *Database Management and Administration* guide.

## Cold Backup

**To make a Cold Backup of your production system:**

1 Locate the ServiceCenter database files you will need to copy. By default these are located in the ServiceCenter DATA directory.

Generally, your database files display in the formats shown below:

| File Description | File Name |
|---|---|
| Database Free Space List | scdb.fre |
| Database Associated Records | scdb.asc |
| Database Logical File Descriptors | scdb.lfd |
| Database Data Records | scdb.db1 |
| Additional Database Data Records | scdb.db2 |
| Additional Database Data Records | scdb.db3 |
| Additional Database Data Records | scdb.db4 |
| Additional Database Data Records | scdb.db5 |
| Additional Database Data Records | scdb.db6 |
| Additional Database Data Records | scdb.db7 |

The first four files MUST be backed up. Back up the other files, if they are present. (These files allow databases to be stored on multiple drives and allow the user to exceed the 2 GB limit imposed by a single data file.)

2 Select and set up the location where you will copy the files.

- If you are using an OS/390 (MVS) or a Unix system, the files can be copied onto a new location on your production machine, if there is enough space.

- If you are using a Windows system, the *Development* system must be created on a system separate from your *Production* system.

3 Notify your users that you are about to bring down the ServiceCenter system.

4 Bring down your ServiceCenter server.

- If you are running on Unix, run the `scstop` script from the command line in your ServiceCenter directory.
- If you are running on OS/390 (MVS), log on to ServiceCenter as an administrative user and issue the `shutdown` command from a command prompt.
- If you are running on a Windows system, bring up the ServiceCenter Console and click **Stop**.

5 Copy the database files listed in step 1 on page 58. These files are used to create your *Development* system.

6 Restart your *Production* system.

- If you are running on Unix, run the `scstart` script from the command line in your ServiceCenter directory.
- If you are running on OS/390 (MVS), submit your JCL to start the service center application.
- If you are running on a Microsoft Windows system, bring up the ServiceCenter Console and click the **Start** button.

7 Make a backup copy of your *Development* system data files, in case you experience problems during the upgrade and need to restart it. This can be compressed and stored on a CD or other storage media.

### Hot Backup

Hot backup cannot be used on a system mapped to RDBMS.

**To make a Hot backup of your production system:**

1 Set Up the Logging Process.

2 Start the Logging Process.

3 Verify that Logging is Enabled.

4 Backup the Database following the steps in *Cold Backup* on page 58.

5 Stop the Logging Process .

# Creating the New System

**To create the new system:**

1. Install the same version of ServiceCenter used on your *Production* system onto your designated *Development* or *Test* machine.
2. Replace the contents of the DATA Directory with the data files you copied in step 5 on page 59.

   This is your *Development* or *Test* system.

# Installing the SC Upgrade Utility

The SC Upgrade installation process copies the files necessary to run the upgrade. The installation process does not start nor perform the upgrade.

This section includes directions for installing SC Upgrade on a Windows, Unix or OS/390 (MVS) system.

Upgrade Installation Considerations:

- Do not try to run the Upgrade directly from the SC Upgrade CD. The upgrade must perform writes to certain files, making running the upgrade from the CD impossible.
- Do not install either the SC Upgrade nor your file system on an NFS-mounted remote partition. Doing so causes serious performance degradation. Even though a properly-configured NFS system can produce I/O bench marks comparable to a local drive array, the real world performance of NFS-mounted partitions plummets, if data is read in many small pieces instead of one large chunk. ServiceCenter in general and SC Upgrade in particular tend to make many small database reads and writes. An NFS-mounted partition is significantly slower than a local drive running for the SC Upgrade process.
- If your *Production* system is running on a previous version of the RTE, upgrade your *Development* and *Test* systems to the SC 5.1 RTE. See *Upgrading the Run-time Environment* on page 150 for instructions.

For SC Upgrade installation instructions, see one of the following:

- *Unix Installation* on page 61.
- *Microsoft Windows Installation* on page 61.
- *OS/390 or MVS Installation* on page 62.

## Unix Installation

### To install SC Upgrade on Unix:

1 Insert the SC Upgrade CD-ROM into the CD drive.

2 Create an upgrade directory on your ServiceCenter server drive.

3 Copy the following upgrade files from the CD-ROM into the new directory that you just created:

```
transfer.bin
upgrade.dta
upgrade.mak
upgdisp1.dta
preupg.bin
sqlupgrade.unl
upgrade.inf
upgrade.str
upgrade.ver
upgdisp2.dta
upgdbdct.dta
upglang.unl
```

See *The SC Upgrade Application Files* on page 65 for descriptions of the files installed.

4 Remove read-only permissions from the files.

5 When the installation is complete:

- If your system is mapped to an SQL database, review the information in *Preparing Systems Mapped to RDBMS* on page 66. After you have reviewed the RDBMS information, proceed to *Upgrading Your System* on page 69.
- If your system is mapped to the P4 database, proceed to *Upgrading Your System* on page 69.

**Note:** If you are running this upgrade on a system that includes the SC3270 interface, you must run the upgrade from a Unix or Windows client. Attempting to run the upgrade from an OS/390 (MVS) client attached via SC3270 causes the upgrade to fail.

## Microsoft Windows Installation

### To install SC Upgrade on Microsoft Windows:

1 Insert the SC Upgrade CD-ROM into the CD drive.

2 Start `File Manager` or `Explorer`.

3 Create an upgrade directory on your local drive.

4 Copy the following upgrade files from the CD-ROM into the new directory that you just created.

preupg.bin
sqlupgrade.unl
transfer.bin
upgdbdct.dta
upgdisp1.dta
upgdisp2.dta
upglang.unl
upgrade.dta
upgrade.inf
upgrade.mak
upgrade.str
upgrade.ver

See *The SC Upgrade Application Files* on page 65 for descriptions of the files installed.

1 Remove read-only permissions from the files.

2 When the installation is complete:

- If your system is mapped to an SQL database, review the information in *Preparing Systems Mapped to RDBMS* on page 66. After you have reviewed the RDBMS information, proceed to *Upgrading Your System* on page 69.
- If your system is mapped to the P4 database, proceed to *Upgrading Your System* on page 69.

## OS/390 or MVS Installation

This section provides the procedures for installing SC Upgrade on an OS/390 (MVS) system.

### Database Considerations

If you are upgrading a DB2 OS/390 (MVS) mapped system, it may be necessary for you to unconvert all your DB2 tables back into P4. The need to convert all the files back into P4 is dependent on how the system was first mapped.

If a LONG VARCHAR data type exists in the m1 table, then DB2 has used all the remaining space in the buffer pool for that table. Because the table is full, additional fields that are added to the P4 database dictionary will not fit in the m1 table. New fields are placed in a new main table (m2). Some of these fields are maintained internally by ServiceCenter and should go into the m1 table for performance reasons. The only method to get the fields into the m1 table is to unconvert the DB2 tables back into P4 for the duration of the entire upgrade then to reconvert them into DB2.

## Preparing to Install SC Upgrade

Before you begin the upgrade, the size of both the scdb.db1 and scdb.asc files must be increased in order to support the upgrade information.

On a default system:

- The scdb.db1 file should be increased in size by 400 cylinders or 325 MB.
- The scdb.asc file should be increased in size by 175 tracks or 10 MB.

On a customized system:

- The scdb.db1 file should be increased the size by 800 cylinders or 700 MB.
- The scdb.asc file should be increased in size by 350 tracks or 20 MB.

It is imperative that you have enough extra space allocated to your ServiceCenter files for information to be loaded. If there is not enough disk space available during the upgrade, the upgrade will fail.

This space can be reclaimed at the end of the upgrade process by running the LFMAP utility on your file system. See the *P4 File System Utility (SCDBUTIL)* section in the ServiceCenter *Database Management and Administration* for instructions on running LFMAP.

## Installing SC Upgrade on OS/390 or MVS

### To install SC Upgrade on OS/390 (MVS):

1 Select a dataset high-level qualifier (HILEV) for use with all unloaded datasets.

2 Select a target disk for unloading the datasets.

Approximately 235 cylinders or 3525 tracks are needed to unload the distribution files from the tape.

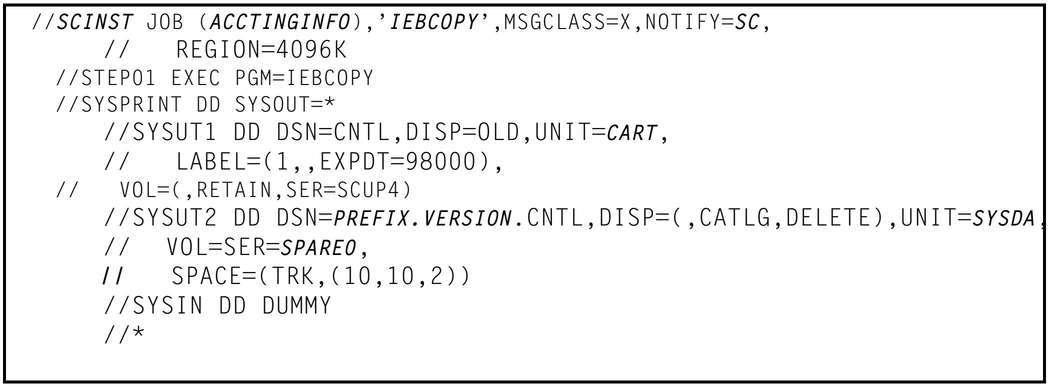
3 Load the CNTL Library:

a Use JCL similar to the example below to unload the first file from the ServiceCenter installation tape.

The first file is a partitioned dataset containing sample JCL members, including the JCL necessary to load the remaining files on the tape.

b Change the items shown in *ITALICS* to conform to your needs.

For *VERSION*, use the version number on the tape label (for example, V4R0M0).



```
//SCINST JOB (ACCTINGINFO),'IEBCOPY',MSGCLASS=X,NOTIFY=SC,
   //   REGION=4096K
 //STEP01 EXEC PGM=IEBCOPY
 //SYSPRINT DD SYSOUT=*
   //SYSUT1 DD DSN=CNTL,DISP=OLD,UNIT=CART,
   //   LABEL=(1,,EXPDT=98000),
 //   VOL=(,RETAIN,SER=SCUP4)
   //SYSUT2 DD DSN=PREFIX.VERSION.CNTL,DISP=(,CATLG,DELETE),UNIT=SYSDA,
   //   VOL=SER=SPARE0,
   //    SPACE=(TRK,(10,10,2))
   //SYSIN DD DUMMY
   //*
```

**Figure 4-1: Submit the job for processing.**

c Mount the tape.

d Verify that the cartridge has been successfully loaded by reviewing the SYSOUT.

4 Load the remaining installation files.

The CNTL partitioned dataset created in the previous steps includes sample JCL for unloading the remaining installation files.

To unload the remaining files:

a Edit the HILEV.VERSION.CNTL member SCLOAD. This member contains notes indicating the changes that must be made.

b Submit the JCL for processing.

c Mount the tape.

d Verify that the cartridge has been successfully loaded by reviewing the SYSOUT.

e At job completion, verify that all job steps issue a return code of 0000.

See *The SC Upgrade Application Files* on page 65 for descriptions of the files installed.

5 When the installation is complete:

- If your system is mapped to an SQL database, review the information in *Preparing Systems Mapped to RDBMS* on page 66. After you have reviewed the RDBMS information, proceed to *Upgrading Your System* on page 69.
- If your system is mapped to the P4 database, proceed to *Upgrading Your System* on page 69.

**Note:** If you are running this upgrade on a system that includes the SC3270 interface, you must run the upgrade from a Unix or Windows client. Attempting to run the upgrade from an OS/390 (MVS) client attached via SC3270 causes the upgrade to fail.

If you are unable to resolve errors associated with one or more job steps, contact Peregrine Systems Customer Support for assistance. Be prepared to provide information about any error messages you received and any steps taken to resolve them.

## The SC Upgrade Application Files

The following table lists the files installed with the SC Upgrade utility to run the upgrade process:

| File name | Contents |
|---|---|
| preupg.bin | Applications and data needed to make changes to the system prior to the upgrade. |
| sqlupgrade.unl (sql.upgrade.unl on OS/390 (MVS) systems) | Applications and data needed to determine new fields that must be added to P4 and RDBMS databases. |
| transfer.bin | The upgrade utility itself and all supporting objects. |
| upgdbdct.dta | Temporary dbdicts needed for the **SQL Compare** process. |
| upgdisp1.dta | Temporary dbdicts needed for the *displayoption* and *displayevent* update. |
| upgdisp2.dta | New *displayoption*, *displayevent*, and *counters* data. |
| upglang.unl | Applications and formats used to prepare a system for multi-language support. |
| upgrade.dta | Upgrade data for all information except dbdicts. |

| File name | Contents |
|---|---|
| upgrade.inf | Upgrade definition file (includes a description of what to upgrade and lists acceptable old signatures). |
| upgrade.mak | Signaturemake file unload (needed on target machine). |
| upgrade.str | Required Database Dictionary upgrades. |
| upgrade.ver | Version stamp for this upgrade. |

# Preparing Systems Mapped to RDBMS

This section discusses the additional steps needed to prepare to run the upgrade if your ServiceCenter files have been mapped to an RDBMS.

If you have converted your ServiceCenter P4 data files to an RDBMS, choose one of the following three options prior to performing the SC Upgrade:

- Let SC Upgrade modify your RDBMS tables for you (highly recommended, except for systems running DB2/MVS). This process is done during Phase II, as explained in step 12 on page 93. Proceed to *Upgrading Your System* on page 69.
- Use the SQL Compare utility to manually update both the P4 database dictionaries and the RDBMS databases before beginning the upgrade process. The procedures for using the SQL Compare utility are outlined in *Using the SQL Compare Utility* on page 169.
- Convert your database back into P4 prior to beginning the upgrade process. When the upgrade process is completed, use the procedures outlined in the SQL guide to reconvert your data files to the RDBMS format. For database conversion instructions, refer to the *RDBMS DB Guide (SQL Support)* chapter in the ServiceCenter *Database Management and Administration Guide*.

**Important:** Converting your database back to P4 is *necessary* for systems running DB2/MVS. Using other procedures impacts the performance of your system after the upgrade is complete.

If you have mapped the signatures, upgradepseudolog, or upgradeobjects tables to SQL, you must check the field lengths of certain fields before you begin the upgrade. The standard default lengths for these fields may not be long enough. You must change the field lengths for these columns in your SQL database to 255 characters.

| Table | Field | Length |
|---|---|---|
| signatures | object.name | 255 |
| upgradepseudolog | message | 255 |
| upgradeobjects | object.name | 255 |

# 5 CHAPTER Upgrading Your System

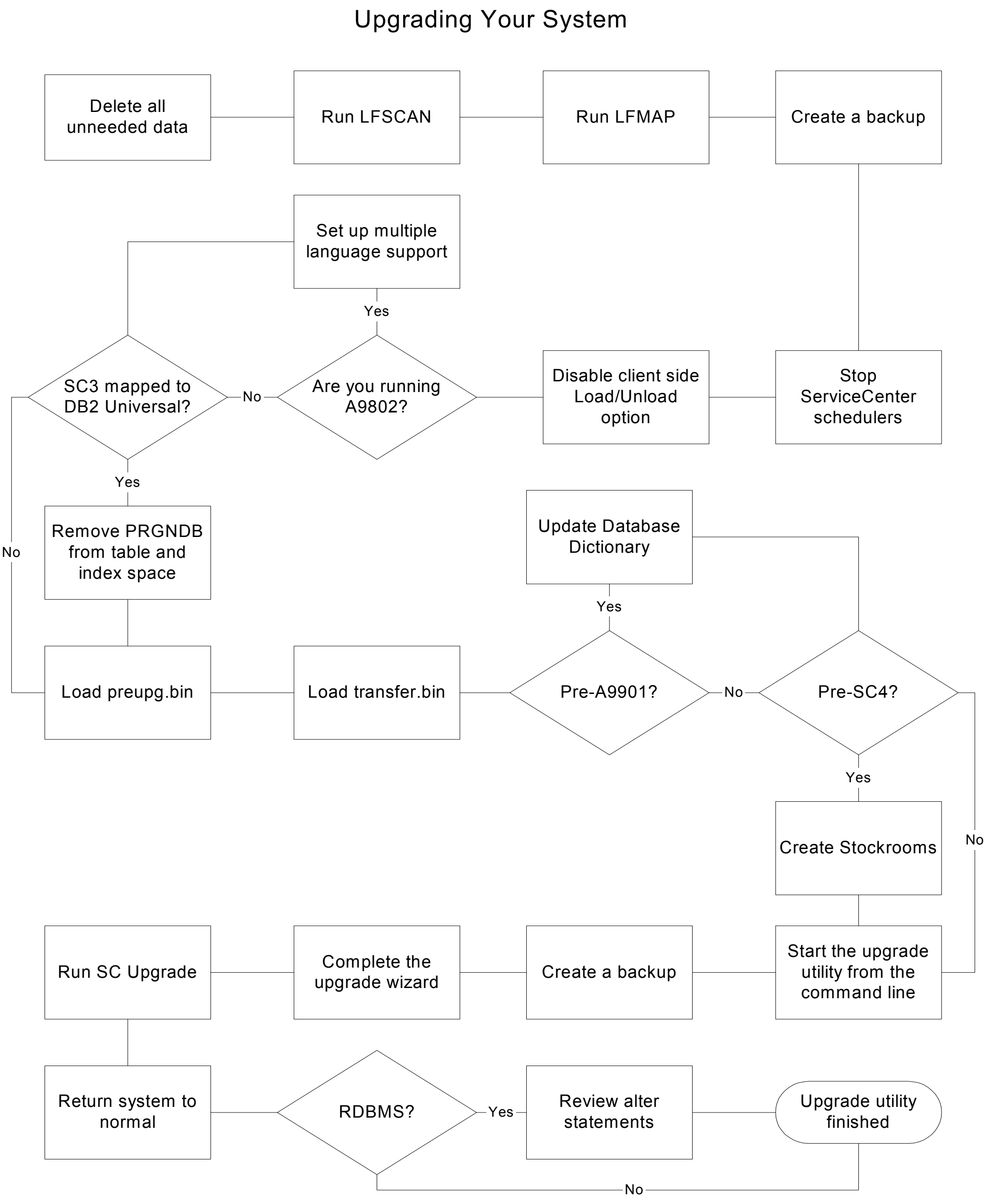
This chapter provides instructions for applying a ServiceCenter Application upgrade and preparing your system for building a custom upgrade.

The instructions in this chapter are to be applied on three different occasions:

- In the *Development* environment while developing the custom upgrade,
- In the *Test* environment while testing the custom upgrade, and
- In the *Production* environment, after the custom upgrade has been built and tested.

This chaper is divided into these sections:

# Upgrade Flowchart



# Phase I: The Preliminary Steps

*Phase I: The Preliminary Steps* is described in the following sections:

- *System Preparation* on page 71 — provides the steps for starting the process.
- *Version A9802 Preparation* on page 72 — provides instructions for loading the applications and data needed to prepare version A9802 and earlier for upgrade. You do not need to do these steps if you have a version later than A9802.

**Important:** These steps must be done in the *Development* environment while developing the custom upgrade, in the *Test* environment, while testing the custom upgrade and again in the production environment while applying the custom upgrade.

## System Preparation

**To get ready to start the upgrade process:**

1 Complete Steps 1 and 2. See *Planning the Upgrade* on page 40, and *Preparing for the Upgrade* on page 40.

2 Delete all data that is no longer needed, such as, msglog, syslog, clocks, old probsummary and problem records, work, and any AUDIT files and related data. Your system should be clean before running the upgrade.

3 You can reduce the amount of time it takes to run the upgrade by removing the IR keys from the probsummary, cm3r, cm3rpage, cm3t and cm3tpage files before applying the upgrade.

**Note:** All of the records in these files will be modified during the upgrade. If you are mapped to SQL the upgrade may create new temporary IR files, which can prolong the upgrade time.

4 Shut down the ServiceCenter Client and Server, if running

5 Run LFSCAN, Option 6, on the backup of your system.

LFSCAN performs a database consistency check on the ServiceCenter database. See the *P4 File System Utility (SCDBUTIL)* section in the ServiceCenter *Database Management and Administration* guide for instructions on running LFSCAN and LFMAP.

Contact Peregrine Customer Support if any inconsistencies are found.

6. Run LFMAP, Option 4, on your system.

   LFMAP repacks data records and index nodes into free space within the same physical file.

7. Make a backup of your *Development* system. If you need to start the upgrade process again, you will have a clean *Development* system available.

8. Prevent the ServiceCenter schedulers from starting up when you start the ServiceCenter client.

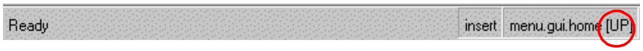
   To do this, edit the sc.cfg file found in the ServiceCenter/RUN directory to comment-out the system.start entry. Alternatively, stop the ServiceCenter schedulers after starting the Client. See *Shutting Down ServiceCenter Schedulers* on page 145.

9. Start the ServiceCenter server for your *Development* system.

10. Log on to the server using an express client. Log in as Sys Admin.

    The SC Upgrade utility is run through a ServiceCenter express client.

11. Be sure that the **Client side Load/Unload** option is disabled. If the client side Load/Unload is enabled, a **U** will be displayed in the right hand corner of the status bar.

    To disable this option, select **File** > **Client side Load/Unload** and ensure that the option is not checked.

**Important:** Before performing any further steps in this chapter, make sure that all schedulers are stopped. Do this by checking the status window.

## Version A9802 Preparation

This section is for version A9802 only. If you are already running version A9901 or later, skip ahead to *Phase II: Applying the Upgrade* on page 75.

The following must be updated before proceeding:

- *Multilingual Support* on page 73
- *RDBMS Support* on page 74

## Multilingual Support

ServiceCenter applications support multiple languages. If you are running application version *A9802*, you must prepare the system for multiple language support.

**Important:** You must perform this step even if you do not plan to run your system in multiple languages.

An upgrade file called upglang.unl has been bundled with the SC51 upgrade utility. This file contains the applications and data needed to allow your current system to run in multiple languages. The upglang.unl file contains a modified version of the login application that is required for multiple language support.

**Note:** The new login application overwrites the version that is currently in your system.

- If you have made changes to the login application and later wish to incorporate them into the SC51 login application, you must copy your login application under a different name. This is necessary to use the SC51 login application after the upgrade is complete.
- You must retrofit any changes you have made in the existing login application in the new SC51 login application.
- You must also have *both* the operator ID and password fields on the login.prompt format. The input for these fields is $user.id and $old.password, respectively.
- Once the upglang.unl file has been loaded, run the associated RAD routines to complete this part of the upgrade.

### To install the multi-lingual support files:

1 Load upglang.unl Use the procedures found in *Loading a File into ServiceCenter* on page 144 to load the upglang.unl file into the ServiceCenter system that is to be upgraded.

If you are upgrading a pre-A9901 system, you may see a series of error messages. See *A9901 Error Messages* on page 79.

2 Enter *aapm.upgrade.language on a command line.

A prompt is displayed with the message: **This process will prepare certain dbdicts for internationalization. Continue?**

3 Click **Yes**.

You are prompted for the default language to run your system.

4 Select the default language from the drop-down list.

**Important:** If running a Japanese system, select **English** for the default language. You can run in Japanese after the upgrade is completed.

5 Click **Next**.

A message is displayed stating: **Process complete. Please inspect any additional messages. You should log off and log back on before continuing.**

- You will log off and log back on at the end of this procedure, not now.
- Active Notes also displays messages, if it is running.

6 Click **OK**.

7 Select an SQL database when prompted (if applicable). Refer to *RDBMS Support* on page 74.

The system prompts you when the process is complete.

8 Log off the current client session and log back on before continuing with the upgrade.

**Note:** This procedure must be completed on the development, test and *Production* systems.

## RDBMS Support

The upgrade language process has added the **syslanguage** field to both the **format** and **help** files. This new field is also part of the indexes for these files. You will need to modify these indexes if either the help, the format file, or both files, are mapped to an RDBMS.

- The **formatm1** table should have two indexes:

| Primary | Secondary |
|---|---|
| syslanguage | syslanguage |
| name | file_name |

- The **helpm1** table will only need its primary index modified:

| | Primary | |
|---|---|---|
| field_name | file_name | format_name |
| syslanguage | term | |

- Using the Database Dictionary utility, you examine the keys of the P4 files (**format**, **help**) if you have any questions.

# Phase II: Applying the Upgrade

*Phase II: Applying the Upgrade* is described in the following sections:

- *Preparing to Upgrade (for ServiceCenter 3 systems mapped to DB2Universal only)* on page 76.
- *Step 1: Loading preupg.bin and transfer.bin* on page 77.
- *Database Dictionary Changes (for pre-A9901 systems only)* on page 79.
- *Creating Stockrooms for Multiple Locations (for pre-SC4 systems only)* on page 81.
- *Step 2: Running the Upgrade Application* on page 83.
- *Step 3: Upgrading System Data* on page 96.
- *Step 4: Returning the System to a Normal Operating Environment* on page 97.
- *Conversion Information (for RDBMS-mapped systems only)* on page 98.

**Important:** These steps must be done in the *Development* environment while developing the custom upgrade, in the *Test* environment, while testing the custom upgrade and again in the production environment while applying the custom upgrade.

Points to be aware of when making an application upgrade:

- Once you have begun these procedures, the system you are upgrading is NOT fully functional until the entire application upgrade is complete.
- Prior to beginning the upgrade process, turn off your screen saver and any power saving options on your computer.

- DO NOT tailor the system on which you create or apply the custom upgrade until the upgrade is complete.
- DO NOT make changes to your *Production* system (Format Control, Forms Designer, etc.) after copying it for the upgrade process. If a change must be made to correct a problem, make another backup of the *Production* system for your *Development* system.
- If the upgrade process fails at any point, you can restart from that point.
- Log on to your *Development* system as an administrator using an *express client* before running any of these procedures.
- Check the log files periodically during the upgrade process to monitor SC Upgrade's progress. See *Tracking the Upgrade Process* on page 150. In OS/390, if you suspect SC Upgrade is hung up, access the job log to see if there is any change in activity.
- During the upgrade process, the Windows Task Manager indicates that ServiceCenter is *Not Responding*. This is normal and does NOT indicate a problem with the upgrade.

## Preparing to Upgrade (for ServiceCenter 3 systems mapped to DB2Universal only)

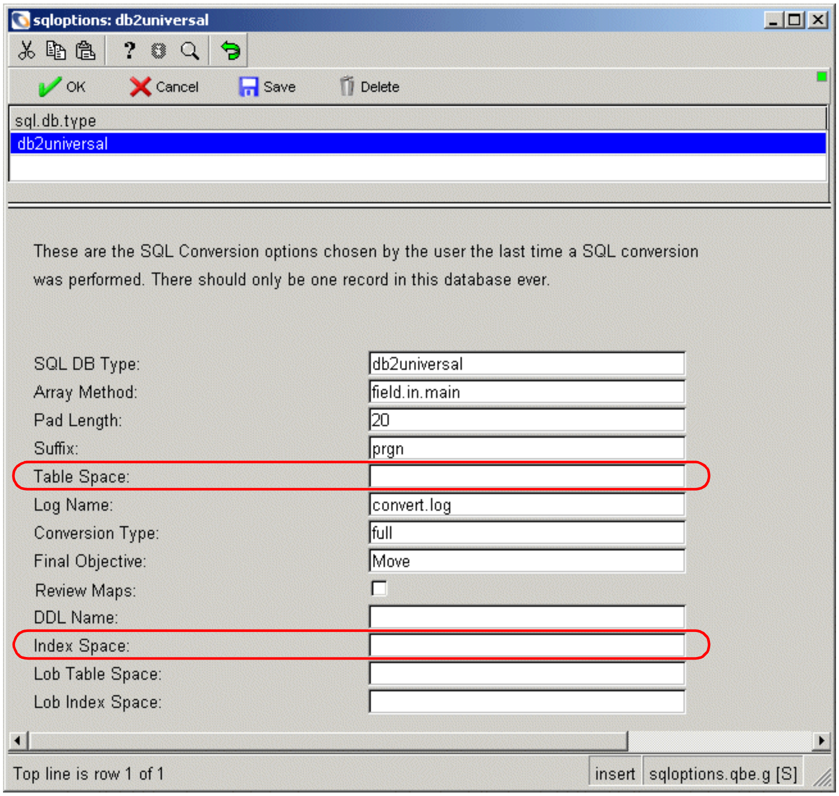
If you are upgrading ServiceCenter 3 and if the system is mapped to DB2Universal, continue with this section. Otherwise, go on to *Step 1: Loading preupg.bin and transfer.bin* on page 77.

SC30xx puts a default PRGNDB name in the sqloptions as the default tablespace name, and the upgrade RTE uses this PRGNDB name for table creation. If PRGNDB appears in the DB2 database, please remove it.

### To remove PRGNDB:

1. Open the file sqloptions using Database Manager.
2. Click **Search.**
3. Select **DB2Universal** from the record list.

The form **sql.options** is displayed.



sql.db.type
db2universal

These are the SQL Conversion options chosen by the user the last time a SQL conversion was performed. There should only be one record in this database ever.

| | |
|---|---|
| SQL DB Type: | db2universal |
| Array Method: | field.in.main |
| Pad Length: | 20 |
| Suffix: | prgn |
| Table Space: | |
| Log Name: | convert.log |
| Conversion Type: | full |
| Final Objective: | Move |
| Review Maps: | □ |
| DDL Name: | |
| Index Space: | |
| Lob Table Space: | |
| Lob Index Space: | |

Top line is row 1 of 1 insert sqloptions.qbe.g [S]

4 Remove **PRGNDB** from the Table Space text box.

5 Remove **PRGNDB** from the Index Space text box.

6 Save the new settings, and exit Database Manager.

## Step 1: Loading preupg.bin and transfer.bin

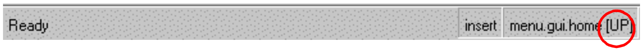
This step should be done regardless of the database type or version you are upgrading from.

An upgrade file named **preupg.bin** is bundled with the SC51 upgrade utility. This file contains the applications and data needed to run the SC Upgrade on your ServiceCenter system. The **transfer.bin** file also is loaded during this procedure.

- If you are upgrading your development or *Test* system, load the preupg.bin file that was included with the Peregrine Systems SC Upgrade media.
- If you are upgrading your *Production* system, use the preupg.bin file that is part of the custom upgrade that you created from your *Development* system.
- If you are applying the custom upgrade to your test or *Production* system, use the files created while building the custom upgrade.
- If you are upgrading a pre-A9901 system, you may see a series of error messages. See *A9901 Error Messages* on page 79.

### To load the files:

1 Be sure that the **Client side Load/Unload** option is disabled. If the client side Load/Unload is enabled, a U will be displayed in the right hand corner of the status bar.



To disable this option, select **File** > **Client side Load/Unload** and ensure that the option is not selected.

2 *Make sure that all schedulers are stopped.* For information on stopping the background schedulers, refer to *Shutting Down ServiceCenter Schedulers* on page 145. You need to stop only the schedulers. The listener can remain running.

3 Using the procedures found in *Loading a File into ServiceCenter* on page 144, load the preupg.bin file into the ServiceCenter system that is to be upgraded.

A message is displayed in the message tray that the load has been completed.

4 Next you will load the transfer.bin file.

5 From the ServiceCenter command line enter load transfer.

**Important:** Use the command line to do this. Do not use the load/unload feature.

A ServiceCenter Upgrade Utility screen prompts you to enter the path where transfer.bin is located. This file was copied to your system when you installed the Upgrade Utility from its media.

6 Enter the fully qualified path for your operating system for Windows (\)and Unix (/), or enter the high level qualifier (.) in OS/390.

**Warning:** The file name (transfer.bin) *must not* be included in the path. Including the file name results in the file not being loaded.

When the process is complete, a message is displayed in the message tray stating: **Transfer files loaded**.

7 Make a clean copy of your *Development* system.

In the event you need to restart the upgrade process, you now have a clean system with the preupg.bin and transfer.bin files.

### A9901 Error Messages

Because the applications have not yet been upgraded to go with the new binaries, you may see several of the following error messages, which may be ignored:

- Query field (*syslanguage*) in (*format*) not defined in the dbdict
- Record add failed
- The record being added contains a duplicate key (*file.load,add.record*)
- file:(globallists) key:(*name=sqlfiles*) (*file.load,add.record*)
- Duplicate key value is: (file:(*globallists*) key:(name=*sqlfiles*)) (*file.load,add.record*)
- sqlExec error: EXEC SQL EXECUTE i17 USING DESCRIPTOR sqlda;
- sqlExec error: sqlcode=-1 errortext=ORA-00001: unique constraint (*ADMIN.FORMATM1_P*) violated
- dbInsert: The record being added contains a duplicate key
- dbInsert: file:(*format*) key:(syslanguage=*de*,name=*apm.make.signatures*)

## Database Dictionary Changes (for pre-A9901 systems only)

If you are upgrading your system from an application version prior to A9901, continue with this section. Otherwise, go on to *Creating Stockrooms for Multiple Locations (for pre-SC4 systems only)* on page 81.

Some database dictionaries contain fields that have changed data types since older versions of ServiceCenter were released. Some tailored systems may also contain fields that have different data types than this upgrade expects.

The SC Upgrade utility cannot upgrade database dictionaries that have fields that do not match the expected data type. You must check these database dictionaries and change the data type of these fields to the expected data type, so that they can be upgraded successfully. The following table lists the database dictionaries in which you may encounter this problem. Not all of these database dictionaries are in every system.

**Table 5-1: Database Dictionary changes for pre-A9901 systems**

| Database Dictionary Name | Field Name | pre-A9901 Data Type | A9901 and later Data Type |
|---|---|---|---|
| benchmark | elapsed.time | Date/time | Number |
| dept | sla.no | Character | Number |
| dept | updated.by | Date/time | Character |
| mail | target | Number | Character |
| pmnotes | number | Number | Character |
| wdSchOptions | wdAutoUsePriority | Logical | Number |
| work | parent.change | Number | Character |

**Note:** If you run an upgrade with fields that have different data types than the upgrade expects, errors are noted in the upgrade logs. You will need to run the upgrade process again after fixing the problem.

The data may be changed automatically by running the **apm.upgrade.fix.data.types** application.

### To run this application from a command line (GUI users only):

1 Enter `*aapm.upgrade.fix.data.types` from a command line.

The data must be modified before the database dictionaries can be modified. Once the application is complete, a prompt is displayed reminding you to change the data types in the database dictionaries.

2 Manually edit the database dictionaries to change the data types of the specified fields.

### To run this application from the RAD Editor (GUI or text mode users):

1. Enter rad on the command line. The RAD editor is displayed.
2. Enter apm.upgrade.fix.data.types in the **Application** field, and press **Enter**.
3. Click the **Test** button in GUI mode or press **F9** in text mode.
   The Application Exerciser is displayed.
4. Press **Enter**. Once the application is complete, a prompt is displayed reminding you to change the data types in the database dictionaries.
5. Manually edit the database dictionaries to change the data types of the specified fields.

## Creating Stockrooms for Multiple Locations (for pre-SC4 systems only)

If you are upgrading from a version prior to ServiceCenter 4, continue with this section. If you are upgrading from ServiceCenter 4 or later, proceed to *Step 2: Running the Upgrade Application* on page 83.

ServiceCenter 4.0 added a Stockroom feature to Request Management. The Stockroom allows each part in the catalog to be placed in a stockroom for the location associated to the part in ServiceCenter. If you are currently running ServiceCenter 4, these stockrooms were created during the installation of that version.

By default, the ServiceCenter Upgrade automatically checks the inventory records and creates a stockroom for each location listed in your ServiceCenter system. A part's quantities are updated for each stockroom, based on that part's location.

Multiple locations can be manually associated with a single stockroom. If you want to have a single stock room to support multiple locations, you must set this up before the upgrade runs.

### To set up multiple locations:

1 Access the SC Upgrade menu by typing `SC51upgrade` (or `SC51upgradetext` in text mode) from a command line and pressing Enter.
The upgrade main menu is displayed.

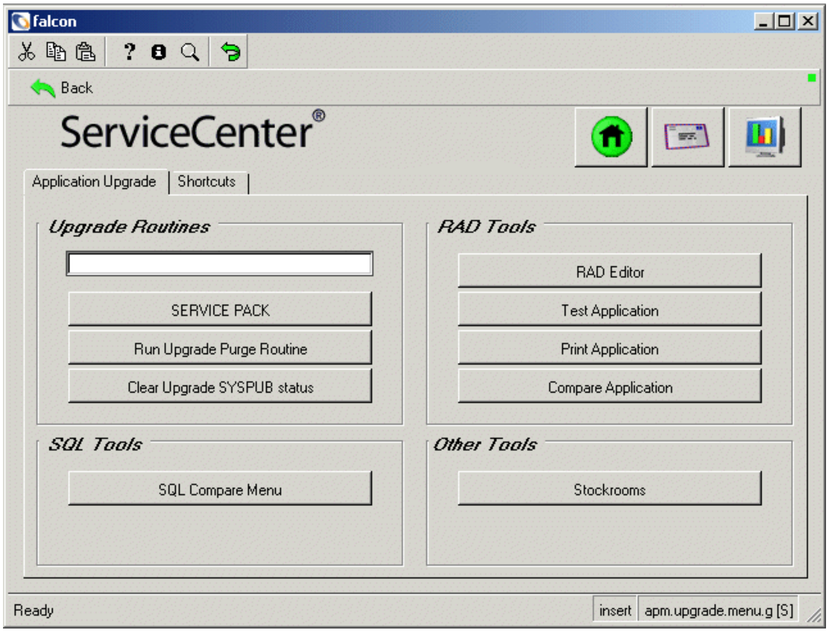


**Figure 5-1: Upgrade Menu — Application Upgrade Tab**

2 From the SC Upgrade menu, select the **Stockrooms** option.

The Stockroom screen is displayed.

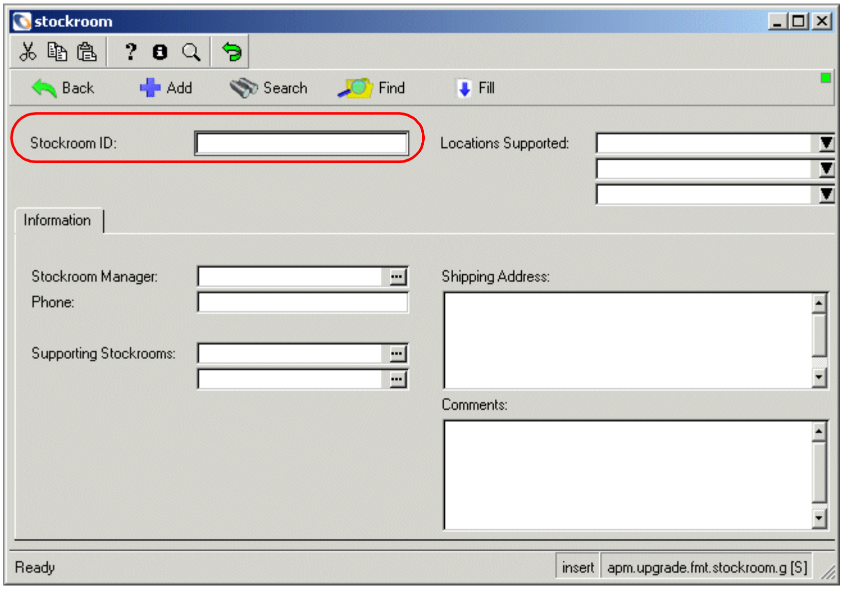


**Figure 5-2: Stockroom Screen**

3 Enter a **Stockroom ID** (for example, Corporate Stockroom).

4 Enter all the locations to be associated to this stockroom from the **Locations Supported** array.

5 Click **Add**.

Repeat for each stockroom that needs to be associated with multiple locations. No action is needed for a stockroom that supports only one location. Single location stockrooms are created automatically when the upgrade is run.

## Step 2: Running the Upgrade Application

This step should be done regardless of the database type or version you are upgrading from.

Next you run the upgrade application to apply the product updates to your system. These updates include the applications added to the new version of ServiceCenter. The applications are installed automatically. If you are upgrading from a version prior to ServiceCenter 4.0, the Root Cause Analysis and Scheduled Maintenance applications are also installed.

## Understanding the Application Upgrade

Once you confirm that the application upgrade should run, it begins upgrading your system. If you are using a GUI client, the upgrade status is provided via on-screen information. You can use the methods outlined in *Monitoring the Progress of the Application Upgrade* on page 148 to monitor the upgrade progress.

### The upgrade is split into three sections:

- The upgrade of the display components.
- The upgrade of the database dictionaries.
- The upgrade of the application data and the purging of the upgrade files.

If the upgrade encounters any problems upgrading the components of any of these sections, it stops automatically at the end of these sections to inform you that it encountered problems.

Whenever the upgrade stops, you have the option of exiting the SC Upgrade process and fixing any problems that may have occurred.

The next time you enter the upgrade (by selecting Apply an Upgrade from the Service Pack menu), SC Upgrade continues where you last left off. Each time you re-enter the upgrade process, you are prompted to confirm the options you want the upgrade to use.

**Note:** When re-entering the upgrade process, it is NOT necessary to run the upgrade purge routine.

Each section of the upgrade should be reviewed to determine if it is necessary to make adjustments due to user customization. While in most cases this determination can be made after the upgrade is complete, it is necessary to correct any problems with the database dictionary upgrade before upgrading the data. Therefore, if the upgrade reports any problems with the database dictionary upgrade, resolve all issues before continuing the data phase of the upgrade.

When the upgrade is complete, it is necessary to investigate the results and correct any conflicts within the system. Refer to *Phase III: Conflict Resolution* on page 100.

## Running the Application Upgrade

### To access the upgrade application:

1 Access the SC Upgrade menu by typing `SC51upgrade` (or `SC51upgradetext` in text mode) from a command line.

2 Press **Enter**.

The upgrade menu is displayed. See Figure 5-1 on page 82.

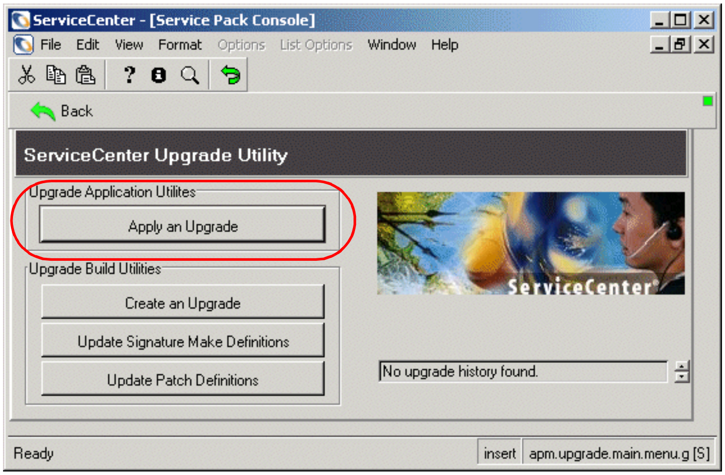
3 From the SC Upgrade menu, select the **SERVICE PACK** option.

The **Upgrade Utility** main menu is displayed. See Figure 5-3 on page 85.

4 Choose the **Apply an Upgrade** option.

- In GUI mode, click the **Apply an Upgrade** button to open the Peregrine Upgrade utility.
- In text mode, select **Apply an Upgrade** (or press **F5**).

The Upgrade Utility Menu is displayed.


ServiceCenter - [Service Pack Console]
File
Edit
View
Format
Options
List Options
Window
Help
Back
ServiceCenter Upgrade Utility
Upgrade Application Utilites
Apply an Upgrade
Upgrade Build Utilities
Create an Upgrade
Update Signature Make Definitions
Update Patch Definitions
ServiceCenter
Ready
insert
apm.upgrade.main.menu.g [S]

No upgrade history found.

**Figure 5-3: Upgrade Utility Menu**

At this point, the SC Upgrade Wizard begins to guide you through a number of questions regarding the upgrade. These questions relate to specific functions of the upgrade process, and must be completed before the upgrade can begin.

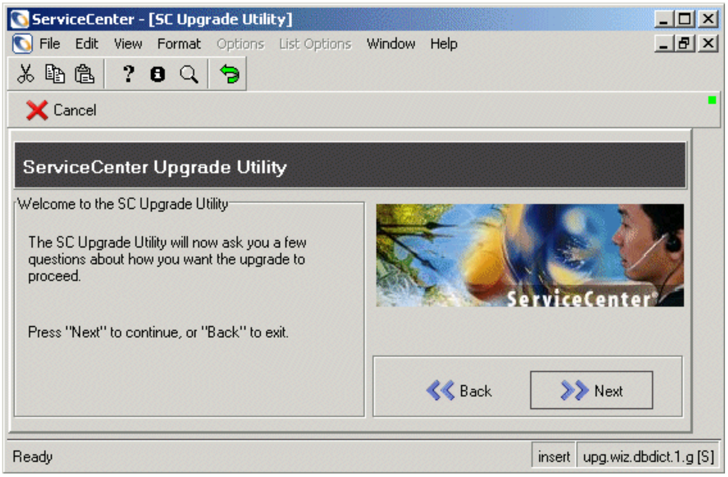
## Using the SC Upgrade Wizard

At any time during this procedure, you can click Next to continue, or click Back to return to previous screens.

### To Use the SC Upgrade Wizard:

1 Select **Apply an Upgrade** option on the Upgrade Utility Menu (Figure 5-3 on page 85).

2 A window asks you to confirm that you are ready to perform the upgrade.


ServiceCenter - [SC Upgrade Utility]
File Edit View Format Options List Options Window Help
Cancel
Back
Next
Ready
insert
upg.wiz.dbdict.1.g [S]

ServiceCenter Upgrade Utility

Welcome to the SC Upgrade Utility

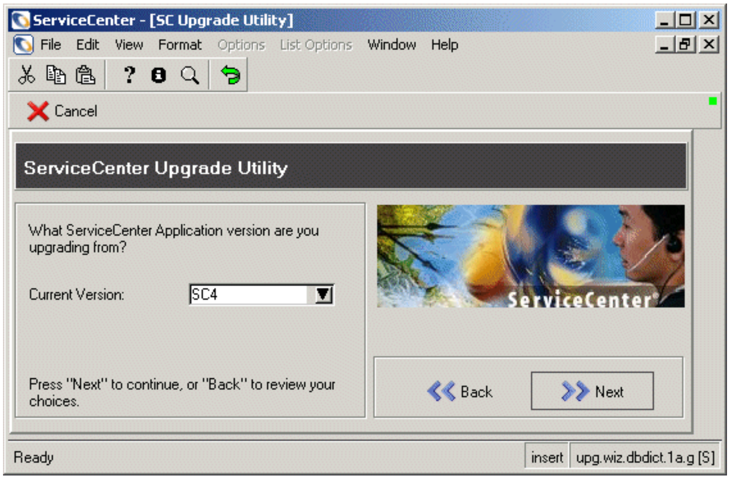
The SC Upgrade Utility will now ask you a few questions about how you want the upgrade to proceed.

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

**Figure 5-4: Upgrade Welcome Screen**

Click **Next** to proceed with the upgrade.

3 A window asks you for the version from which you are upgrading.


ServiceCenter - [SC Upgrade Utility]
File Edit View Format Options List Options Window Help
Cancel
Back
Next
Ready
insert
upg.wiz.dbdict.1a.g [S]

ServiceCenter Upgrade Utility

What ServiceCenter Application version are you upgrading from?

Current Version: SC4

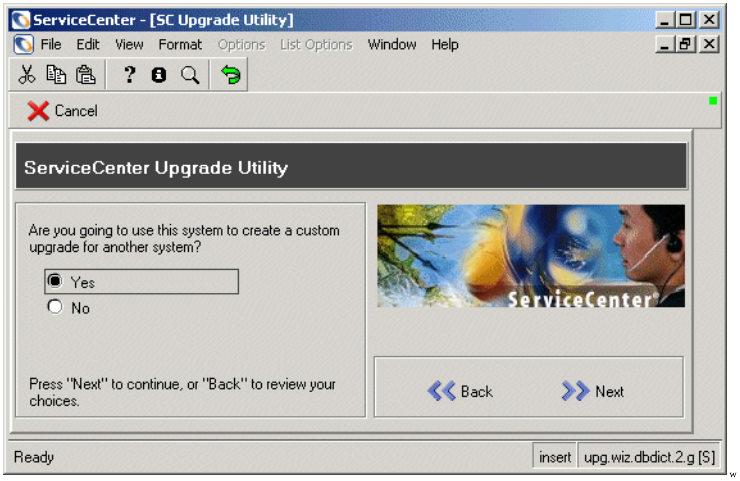
Press "Next" to continue, or "Back" to review your choices.

**Figure 5-5: Version Prompt**

Select the version number from the drop-down list and click **Next**.

**Note:** If you are not sure of the current (old) version, select the oldest version.

4 A window asks if you are going to use this system to create a custom upgrade for another system.



ServiceCenter - [SC Upgrade Utility]
File Edit View Format Options List Options Window Help
Cancel
ServiceCenter Upgrade Utility
ServiceCenter
Back
Next
Ready
insert
upg.wiz.dbdict.2.g [S]

Are you going to use this system to create a custom upgrade for another system?

● Yes

○ No

Press "Next" to continue, or "Back" to review your choices.

**Figure 5-6: System Selection**

If this is your *Development* system, select **Yes**.

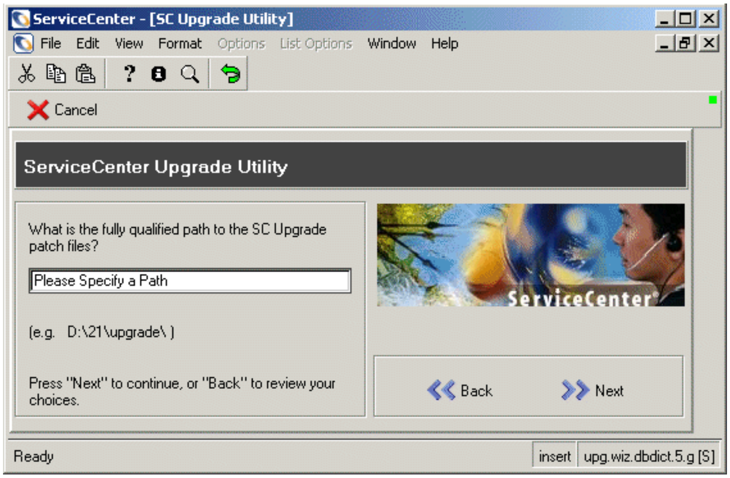
— Or —

If this is your *Production* or *Test* system, select **No**.

**Warning:** If you have not tested your custom upgrade on a *Test* system do not select **Yes**. Instead, select **No** and run it on a *Test* system. If you have not created a *Test* system, go to *Preparing the Development System* on page 55.

If you have already run SC Upgrade on your *Development* system and you have not yet created a custom upgrade, go to *Building the Custom Upgrade* on page 131.

5 A window asks for the fully qualified path to the SC Upgrade patch files.



ServiceCenter Upgrade Utility

What is the fully qualified path to the SC Upgrade patch files?

Please Specify a Path

(e.g. D:\21\upgrade\ )

Press "Next" to continue, or "Back" to review your choices.

Ready | insert | upg.wiz.dbdict.5.g [S]

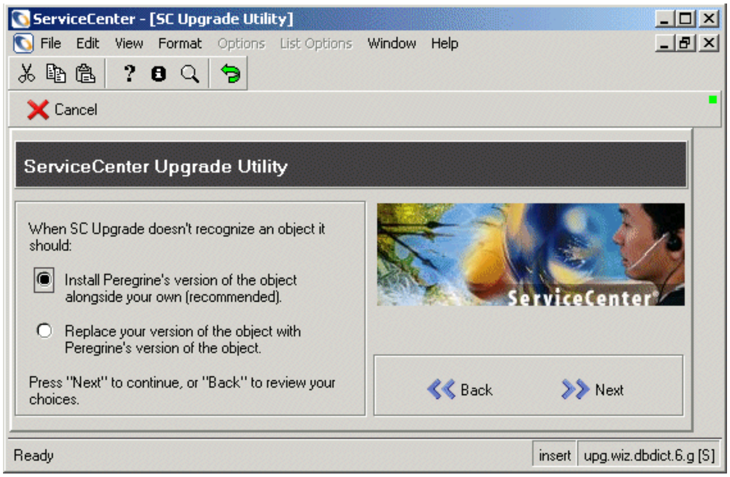
**Figure 5-7: Set the Path to the Upgrade Files**

Enter the final directory qualifier in the path.

**Note:** This should be the same location from which you loaded the preupg.bin and transfer.bin files in *Step 1: Loading preupg.bin and transfer.bin* on page 77.

- *OS/390 (MVS)* — This path must contain a final directory qualifier in your path (for example, SC51.USR.).
- *Unix* — This path must contain a final directory qualifier (slash) (for example, /sc/upgrade/).
- *Windows* — This path must contain a final directory qualifier (back slash) (for example, C:\SC51\upgrade\).

6 A window asks how the upgrade should act when it encounters parts of the file system that you have customized.



When SC Upgrade doesn't recognize an object it should:

● Install Peregrine's version of the object alongside your own (recommended).

○ Replace your version of the object with Peregrine's version of the object.

Press "Next" to continue, or "Back" to review your choices.

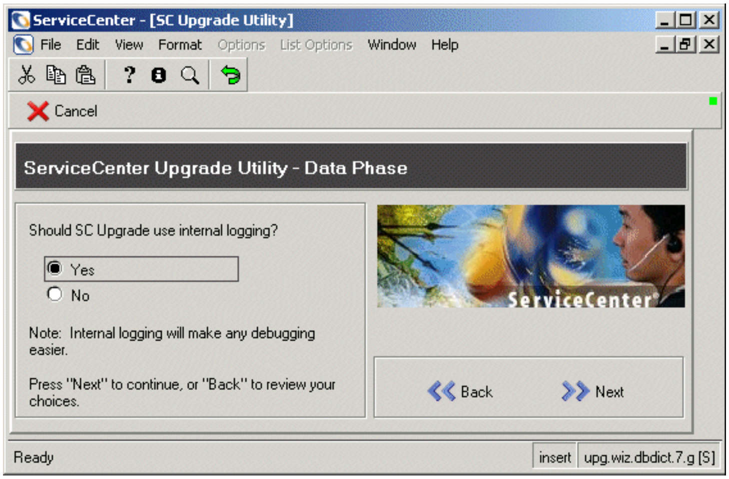
**Figure 5-8: Handling Objects**

Select Install Peregrine's Version of the Object Alongside Your Own to have SC Upgrade rename the Peregrine Systems version of the object to `NEW<release><object name>` (for example, `NEWSC51pm.main`). (Recommended)

— Or —

Select **Replace your version of the object with Peregrine's Version** to have SC Upgrade rename your version of the object `OLDSC4<object name>` (for example, `OLDSC51pm.main`).

7 A window asks if SC Upgrade should use internal logging.



ServiceCenter Upgrade Utility - Data Phase

Should SC Upgrade use internal logging?

● Yes

○ No

Note: Internal logging will make any debugging easier.

Press "Next" to continue, or "Back" to review your choices.

**Figure 5-9: Setting Internal Logging**

Select **Yes** to use internal logging. Internal logging stores the log files within ServiceCenter. Click **Next** to continue.

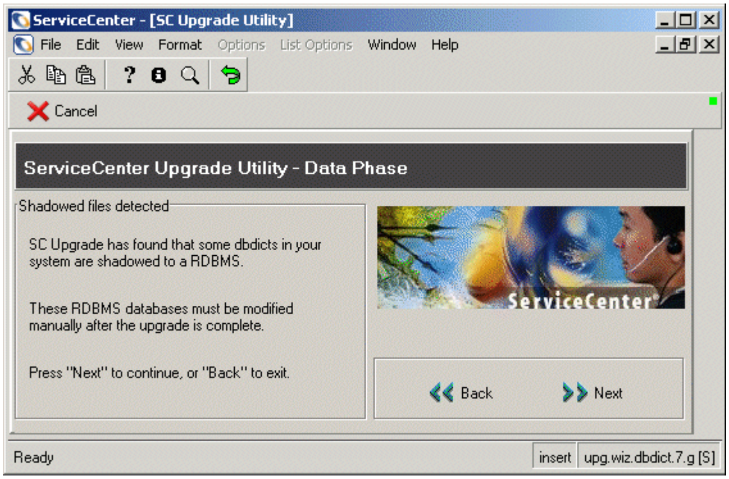
— Or —

Select **No** to use *only* external logging. External logging stores the log files within the same directory as the upgrade files. Click **Next** to continue.

External logging will be used for both options. See *Tracking the Upgrade Process* on page 150 for a list of log files.

8 One of several screens may be displayed at this point.

- If any databases are shadowed to an RDBMS, the warning, shown in Figure 5-10 on page 91, is displayed. SC Upgrade does not make any modifications to the RDBMS for shadowed databases. Click **Next** and proceed to one of the two following bulleted options.
- If your system is mapped to an RDBMS database, proceed to step 9 on page 91.
- If your system is not mapped to an RDBMS database, the upgrade application procedures are complete. Proceed to step 14 on page 95 to continue with the upgrade.


ServiceCenter - [SC Upgrade Utility]
File Edit View Format Options List Options Window Help
Cancel
Back
Next
ServiceCenter

ServiceCenter Upgrade Utility - Data Phase

Shadowed files detected

SC Upgrade has found that some dbdicts in your system are shadowed to a RDBMS.

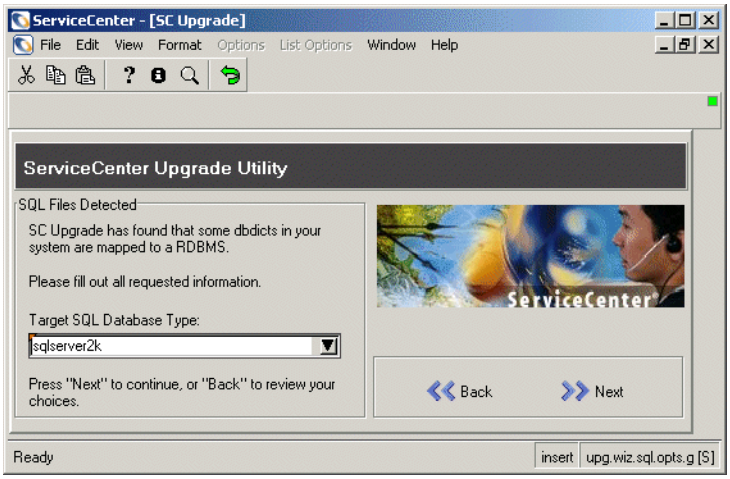
These RDBMS databases must be modified manually after the upgrade is complete.

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

Ready | insert | upg.wiz.dbdict.7.g [S]

Figure 5-10: Shadowed Database Dictionary Files

9 A window asks you for the target SQL Database type (for example, Oracle).


ServiceCenter - [SC Upgrade]
File Edit View Format Options List Options Window Help
Back
Next
ServiceCenter

ServiceCenter Upgrade Utility

SQL Files Detected

SC Upgrade has found that some dbdicts in your system are mapped to a RDBMS.

Please fill out all requested information.

Target SQL Database Type:

sqlserver2k

Press "Next" to continue, or "Back" to review your choices.

Ready | insert | upg.wiz.sql.opts.g [S]

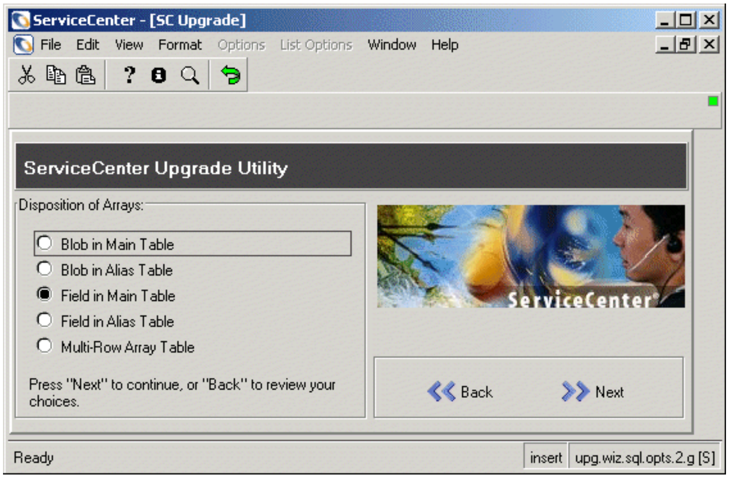
Figure 5-11: SQL Files Detected

Select the target SQL Database type and click **Next**.

**Note:** If you have any problems with the RDBMS portion of this procedure, refer to the *RDBMS Support* section if the ServiceCenter *Database Management and Administration* guide.

10 A window asks you for the disposition of array fields.

**Note:** If you are unfamiliar with the database on which ServiceCenter runs, please get assistance from your database administrator.



Disposition of Arrays:

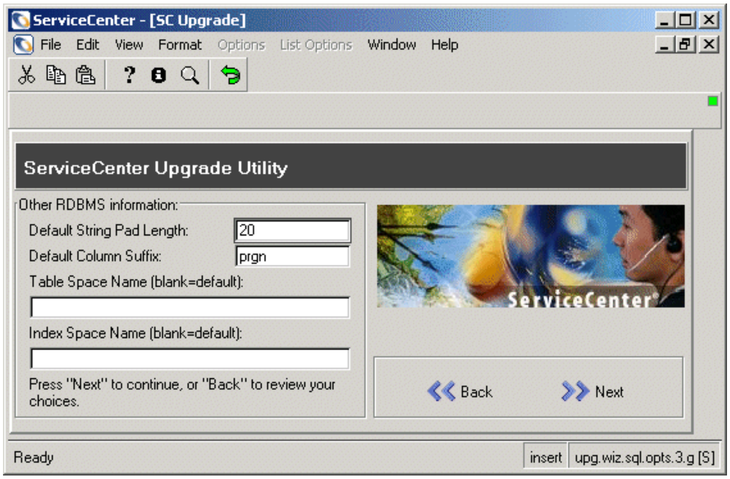
○ Blob in Main Table
○ Blob in Alias Table
● Field in Main Table
○ Field in Alias Table
○ Multi-Row Array Table

Press "Next" to continue, or "Back" to review your choices.

Figure 5-12: Disposition of Arrays

Select the disposition of array fields.

11 A window asks you for other RDBMS information.



Other RDBMS information:

Default String Pad Length: 20

Default Column Suffix: prgn

Table Space Name (blank=default):

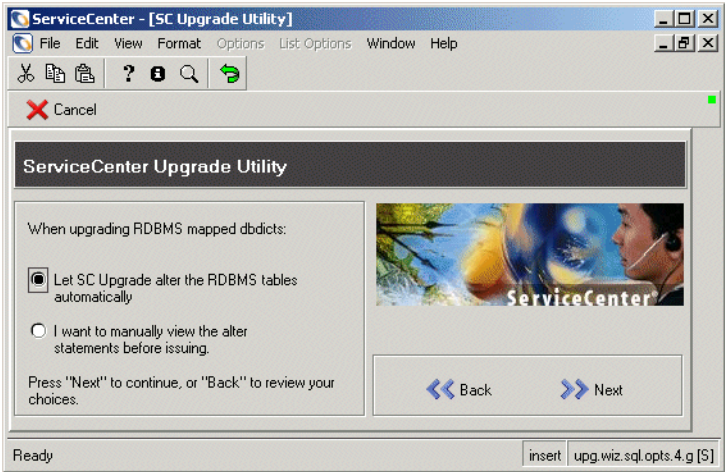
Index Space Name (blank=default):

Press "Next" to continue, or "Back" to review your choices.

Figure 5-13: Default Table Settings

Set the **default string pad length**, **default column suffix**, **Table Space Name**, **Index Space Name**, and then click **Next**. The default is to leave the Table Space Name and Index Space Name blank.

12 A window asks you whether you want RDBMS tables to be automatically updated.



When upgrading RDBMS mapped dbdicts:

● Let SC Upgrade alter the RDBMS tables automatically

○ I want to manually view the alter statements before issuing.

Press "Next" to continue, or "Back" to review your choices.

Ready | insert | upg.wiz.sql.opts.4.g [S]

**Figure 5-14: Selecting Manual or Automatic Upgrade**

Select whether you want SC Upgrade to update the RDBMS tables automatically, or allow you to view and modify the alter statements before they are issued. Click **Next**.

For an explanation of alter statements, see *Conversion Information (for RDBMS-mapped systems only)* on page 98.

- 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.
- If at any point the upgrade has problems trying to modify your SQL mapping, the process pauses and you are prompted with the SQL alter statement that failed. When this occurs, you can change the alter statement and resubmit it, or you can fix the problem outside of ServiceCenter and then skip the alter statement.

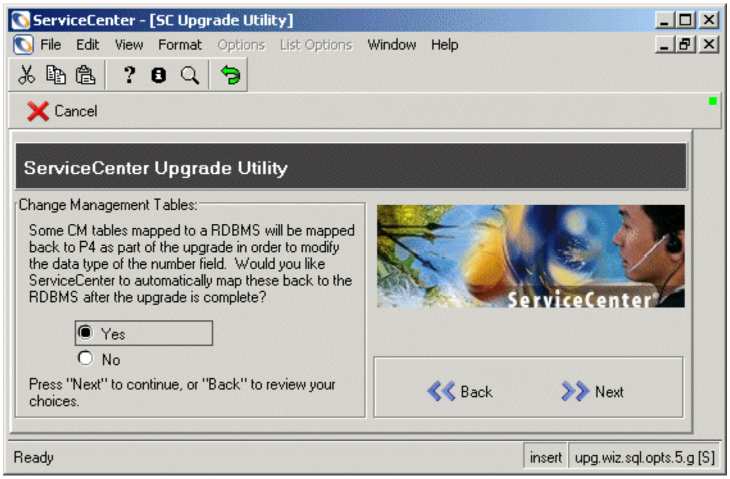
**Note:** If the problem is not fixed, the database dictionary in question remains unstable until the problem is resolved.

13 A window asks if you want the Change Management files (`cm3r`, `cm3t`, `cm3rpage`, and `cm3tpage`) re-mapped automatically after they have been unconverted so that the data type can be changed (see Figure 5-15 on page 94). Currently, there should be no sqlhints for the **Number** field for these files. If there are, you should delete them before running the upgrade.

During the re-mapping, the data from these files will exist in P4 temporarily. You need to ensure that these files exist in data pools that will not exceed the 2GB limit when combined with the Change Management data. You may need to extend them over multiple pools (see the *Database Management and Administration Guide* for details).

You can choose to have the files re-mapped automatically, based on what has been specified in the sqlhints file. However, if you have made mapping changes manually, outside the sqlhints file, you will need to re-map these changes after the upgrade process has finished and the background schedulers are done.

For instructions on checking to see if the background schedulers have finished, see *Step 3: Upgrading System Data* on page 96.



Change Management Tables:

Some CM tables mapped to a RDBMS will be mapped back to P4 as part of the upgrade in order to modify the data type of the number field. Would you like ServiceCenter to automatically map these back to the RDBMS after the upgrade is complete?

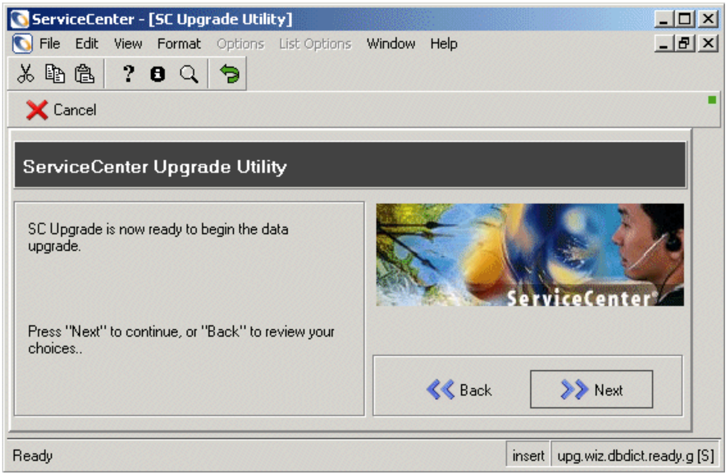
● Yes
○ No

Press "Next" to continue, or "Back" to review your choices.

**Figure 5-15: Change Management Table Mapping**

Select **Yes** for automatic re-mapping or **No** for manual re-mapping later, and then click **Next** to continue.

14 A window asks you to confirm that the data you entered is correct and that you are ready to start the SC Upgrade process.


ServiceCenter - [SC Upgrade Utility]
File Edit View Format Options List Options Window Help
Cancel
ServiceCenter Upgrade Utility
ServiceCenter
Back
Next
Ready
insert
upg.wiz.dbdict.ready.g [S]

SC Upgrade is now ready to begin the data upgrade.

Press "Next" to continue, or "Back" to review your choices..

Figure 5-16: Start Upgrade Prompt

Click **Next** to continue, or click **Back** to review your choices.

15 A window asks if you want to start the upgrade. The window displays the build information of the system you are upgrading.

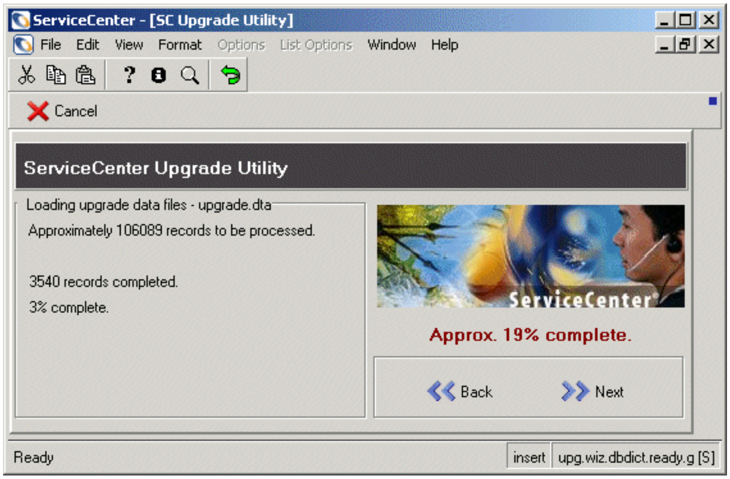
Click **Yes** to start the upgrade.

— Or —

Click **No** if you want to run the upgrade at another time.

If your system is mapped to an RDBMS, the Upgrade Utility will force you to log off after the dbdict upgrade. This action forces a refresh of the SQL cache. To continue, log on and access the upgrade application. The Upgrade Utility will continue from where it left off.

While the upgrade runs, the Upgrade Utility displays the progress of the upgrade process by indicating which process is underway, the progress of the records being processed, the time remaining in the step, and the completion percentage for the upgrade.



ServiceCenter - [SC Upgrade Utility]

ServiceCenter Upgrade Utility

Loading upgrade data files - upgrade.dta

Approximately 106089 records to be processed.

3540 records completed.

3% complete.

Approx. 19% complete.

Ready | insert | upg.wiz.dbdict.ready.g [S]

Figure 5-17: Upgrade Progress

When the upgrade process is finished, a message is displayed stating that the upgrade is complete.

16 Click **OK** in the message box.

17 Once the upgrade completes (the background processes are finished), and if you have removed them, (step 3 on page 71) add the IR keys back to `probsummary` and `cm3*` files. See the IR Expert chapter of *Database Management and Administration*.

## Step 3: Upgrading System Data

This step should be done regardless of the database type or version you are upgrading from. Changes to Incident and Change Management make it necessary to update the actual data within the system. All active incidents and changes will be upgraded during the foreground upgrade process. All inactive or closed incidents and changes and some cost information related to incidents are upgraded by background schedulers. Each of these are run in separate background processes that will automatically be started by the upgrade.When these schedulers are finished, they will automatically log themselves out.

**Background Schedulers Used in the Upgrade Processes**

| | |
|---|---|
| upgrade | Upgrades incidents |
| upgradecm3 | Upgrades changes |
| upgradecost | Upgrades costs related to incidents |

**Warning:** Do not shut down the system or build the custom upgrade until these three schedulers have finished running.

**To determine if the background schedulers have finished:**

1. Open the System Status window by clicking **System Status** on the system administrator's home menu. Once the processes are complete, the background schedulers are no longer displayed in the System Status window.
2. Alternatively, you can view the status of the background upgrade using a ServiceInfo client. The name of the format to open is: apm.upgrade.status.display.

   For information on how to use the ServiceInfo client, see *Monitoring the Progress of the Application Upgrade* on page 148.

## Step 4: Returning the System to a Normal Operating Environment

This step should be done regardless of the database type or version you are upgrading from.

After the upgrade is complete, the system may exhibit abnormal behavior until it is brought back to its normal operating environment. To do this, you need to start all the schedulers that were shut down during the upgrade. Your globallists will be updated the next time you log on to the system.

**Return to a normal operation environment by restarting the schedulers:**

1. Log out.
2. Edit the sc.cfg file found in the ServiceCenter/RUN directory to undo the changes you made when you commented out the system.start entry.
3. Log in again.

Alternatively, start the ServiceCenter schedulers following the instructions in *Starting the ServiceCenter Schedulers* on page 146.

## Conversion Information (for RDBMS-mapped systems only)

If you are upgrading a system mapped to an RDBMS, continue with this section. Otherwise, continue to *Phase III: Conflict Resolution* on page 100.

**Warning:** If you have Change Management files converted to an RDBMS, you must specify the sqldrop:1 parameter in the initialization file (sc.ini or PARMS) in order to upgrade to SC5 or higher applications. If you do not specify the sqldrop:1 parameter, the conversion back to the RDBMS will fail.

### Alter Statements

If you are upgrading a system mapped to an RDBMS and you selected the option to review the alter statements that the SC Upgrade utility creates, the upgrade stops after every upgraded database dictionary that requires an SQL alter statement to be applied. At this time the alter statements are displayed and you may choose to:

- Allow SC Upgrade to apply the alter statement to the RDBMS exactly as it appears.
- Modify the alter statement manually, then allow SC Upgrade to apply that statement.
- Handle the altering of the RDBMS tables manually, having SC Upgrade skip the alter statement.

After you have viewed the alter statement and made any changes that you wish to make:

- Choose the **Proceed** button to apply the alter statement.

— *Or* —

- Choose the **Skip** button to move on without applying.

If the alter statement is unsuccessful, you are returned to the alter screen. You can edit the statement and try again. You can also quit the upgrade process.

**Note:** If you quit the upgrade process before the alter statement is applied successfully, the database dictionary in question becomes unstable until the SQL mapping is corrected.

## System Tables

If, on your system, one of the following tables is mapped to an RDBMS and is not currently a system table, it will be converted into one during the upgrade process.

### ServiceCenter System Tables

| | | |
|---|---|---|
| application | applicationrevision | caldaily |
| category | cm3profile | cm3ralerts |
| cm3rcatphase | cm3talerts | cm3tcatphase |
| datadict | datamap | dbdict |
| displaycache | displayevent | displayeventrev |
| displaymaster | displayoption | displayoptionrev |
| displayscreen | displayscreenrev | dtqueue |
| dtshad | enclapplication | enclapplrev |
| enclapplrevision | environment | eventfilter |
| eventin | eventmap | eventout |
| eventregister | format | formatcontrolrevision |
| formatctrl | formatrevision | globallists |
| help | icmenv | info |
| irqueue | link | linkrevision |
| macro | macrodef | macroheader |
| menu | menucmdlist | menurevision |
| msglog | Object | Objectrevision |
| ocmalertlog | ocmalertpool | ocmapprlog |
| ocmapprpool | ocmcatselect | ocmevents |
| ocmlcat | ocmocat | ocmoptions |
| ocmphaselog | ocmprofile | ocmqcat |
| patches | pmenv | pmnotes |

**ServiceCenter System Tables**

| | | |
|---|---|---|
| Process | Processrevision | sc |
| schedule | scparms | screlconfig |
| scripts | signatures | slacontrol |
| smenv | sqldbdict | sqlqueue |
| Staterevision | States | status |
| systemperform | termtype | tzfile |
| upgdbdict | upginfo | upgrade |
| upgradeapplication | upgradeddbdict | upgradeobjects |
| upgradepseudolog | upgradepsuedolog | upgradestatus |
| validity | | |

# Phase III: Conflict Resolution

The conflict resolution phase of the application upgrade allows you to fix the upgrade conflicts due to customization of the system.

Conflict resolution is the most important part of the Service Center Applications upgrade. Without it, your upgrade will not work properly.

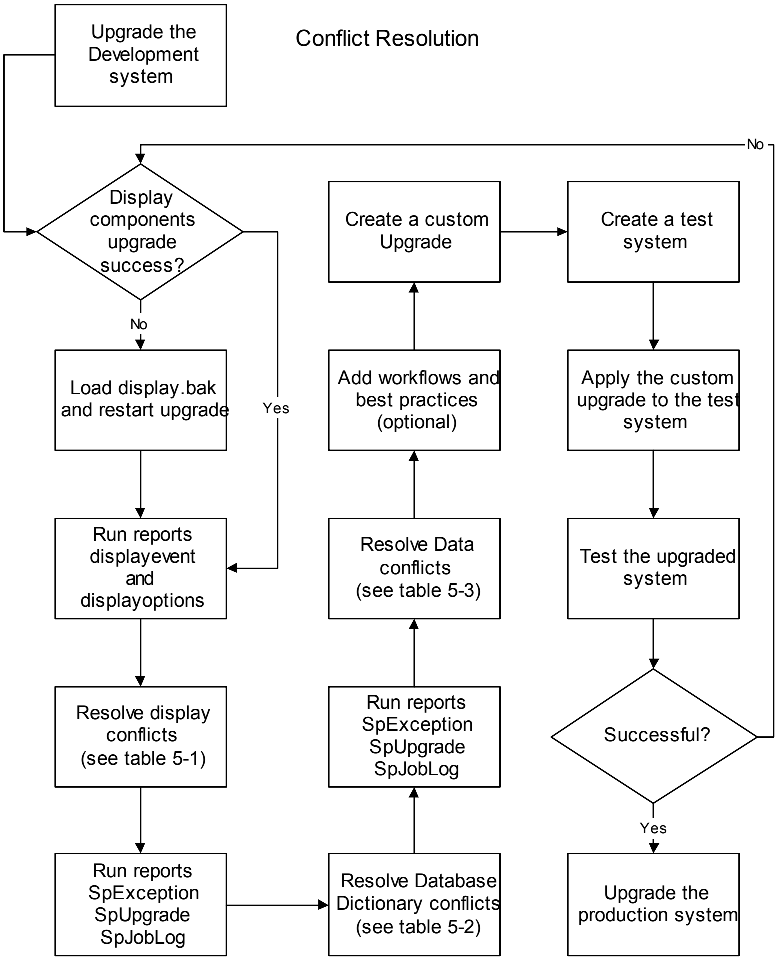
**Note:** If you run into problems while applying the upgrade, refer to *Troubleshooting* on page 177.

*Phase III: Conflict Resolution* is described in the following sections:

- *Step 1: Running Post Upgrade Reports* on page 102.
- *Step 2. Conflict Resolution of the Display Components* on page 105.
- *Step 3. Conflict Resolution of the Database Dictionaries* on page 110.
- *Step 4: Conflict Resolution of the Data* on page 111.

**Important:** These steps must be done in the *Development* environment while developing the custom upgrade, in the *Test* environment, while testing the custom upgrade and again in the production environment while applying the custom upgrade.

# Conflict Resolution Flow


Conflict Resolution
Upgrade the Development system
Display components upgrade success?
No
Yes
Load display.bak and restart upgrade
Run reports displayevent and displayoptions
Resolve display conflicts (see table 5-1)
Run reports SpException SpUpgrade SpJobLog
Resolve Database Dictionary conflicts (see table 5-2)
Run reports SpException SpUpgrade SpJobLog
Resolve Data conflicts (see table 5-3)
Add workflows and best practices (optional)
Create a custom Upgrade
Create a test system
Apply the custom upgrade to the test system
Test the upgraded system
Successful?
No
Yes
Upgrade the production system

## Step 1: Running Post Upgrade Reports

After the upgrade has finished running, run reports using the Report Exerciser to determine what conflict resolution is necessary.

The Shortcuts tab of the Upgrade menu (Figure 5-18 on page 103) provides short cuts to some of the applications needed for Conflict resolution, such as Database Manager (Database button) and Reports.

**Table 5-2: Upgrade Reports**

| Report Name | Report Title | Description |
|---|---|---|
| Results of displayevent upgrade | apm.upgrade.display.event | A report on display options. † |
| Results of displayoption upgrade | apm.upgrade.display.opts | A report on display events. † |
| Service Pack Exception Report | apm.upgrade.results.exceptions | A listing of the objects in the system that SC Upgrade could not upgrade automatically. ‡ |
| Service Pack Full Upgrade Report | apm.upgrade.results.full | A listing of all objects in the system that SC Upgrade processed, even if they upgraded smoothly.‡<br>This is a large report that typically contains more than *10,000* lines.<br>The SC parameter *maxreportpages* defaults to 1000. If you do not explicitly set it to a higher number in the sc.ini or PARMS file, your report terminates after printing 1000 pages. |
| Service Pack Job Log | apm.upgrade.job.log | A dump of the upgrade job log. ‡<br>This log can be useful for debugging purposes with Peregrine Systems Customer Support or simply for informational purposes. Generally speaking, this report is not necessary. |

† The Display reports are discussed in *Step 3. Conflict Resolution of the Database Dictionaries* on page 110.
‡ The Service Pack reports are discussed in *Changes in Change Management* on page 116.

## Viewing Post Upgrade Reports

Reports can be accessed:

- from the Shortcuts tab of the Upgrade menu.
- from the ServiceCenter Command line.

### To access a report:

1 Enter `re` on a command prompt.

2 Select the desired report by entering the name of the report in the Report Name text box. (See page 102 for a list of Upgrade Reports and what they display.)

— *or* —

a Enter `SC51upgrade` on a command line of an express client.

b The upgrade menu is displayed. See Figure 5-1 on page 82.

c Select the **Shortcuts** tab.

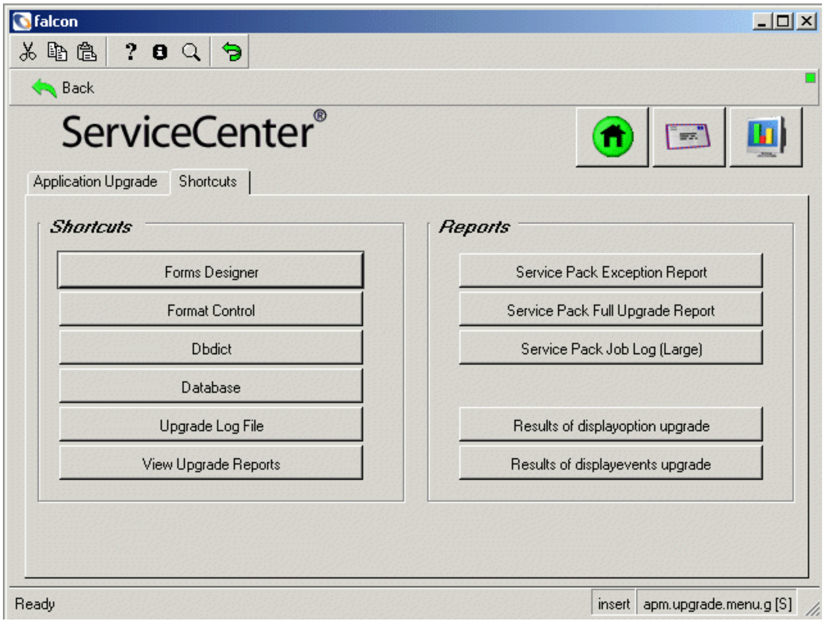
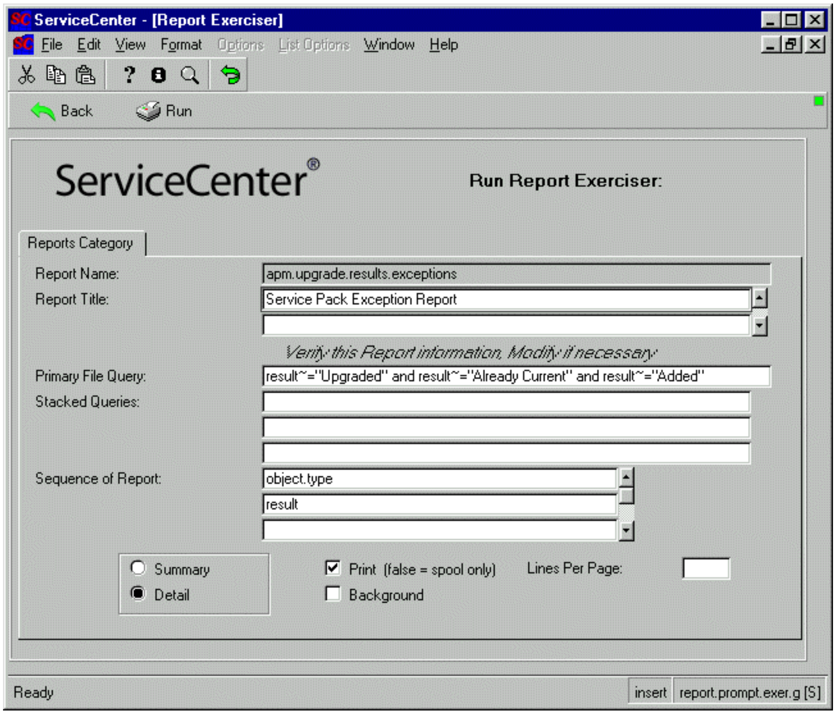


Figure 5-18: Upgrade Menu — Shortcuts Tab

d Select the report by clicking the button for the desired report. (See page 102 for a list of Upgrade Reports and what they display.) The Report Exerciser will open.


ServiceCenter - [Report Exerciser]
File Edit View Format Options List Options Window Help
Back
Run
ServiceCenter®
Run Report Exerciser:
Reports Category

Report Name: apm.upgrade.results.exceptions

Report Title: Service Pack Exception Report

*Verify this Report information, Modify if necessary*

Primary File Query: result~="Upgraded" and result~="Already Current" and result~="Added"

Stacked Queries:

Sequence of Report: object.type<br>result

○ Summary

● Detail

☑ Print (false = spool only)

☐ Background

Lines Per Page:

Ready | insert | report.prompt.exer.g [S]

Figure 5-19: Selecting Report Options

3 If you are familiar with running ServiceCenter reports, you can change any of the options at this point. The default options are usually adequate and need not be changed.

Table 5-3: Report Screen Options

| Option | Definition |
|---|---|
| Print | If the **Print** option is selected, the **Schedule a Report** screen displays to allow you to set a printer and a time for the report to be run and printed. The report is printed on the printer selected.<br>If the **Print** option is NOT selected, the report is displayed in the ServiceCenter window after it runs.<br>**Note:** The print option is only available when the client is set up for server printing. If you have completed the upgrade and the database option. |
| Background | If the **Background** option is flagged, the **Schedule a Report** screen displays to allow you to set a printer and a time for the report to be run and printed. The report spools in the background. The client is returned to your control almost immediately, but the report may not spool for a few minutes until started by the scheduler.<br>If the **Background** option is NOT flagged, the report spools in the foreground. The client is locked until the report is finished spooling. |

4 Click **Run** or press **F1** to start the report.

If the report returns the message *No records selected by report* when run, the components were upgraded successfully. No records exist for that file that need attention. If you get any other message, further action is required.

## Step 2. Conflict Resolution of the Display Components

SC Upgrade upgrades the display components as part of the upgrade process. A conflict can occur if any of these components has been customized. You need to be familiar with the Display application and how it functions to resolve these conflicts. For more information on the Display application, refer to the *Display Application* section in the ServiceCenter *System Tailoring Guide*.

After the display component section of the upgrade has finished running, you can run the displayoption (apm.upgrade.display.opts) and displayevents (apm.upgrade.display.event) reports using the Report Exerciser to determine any additional manual steps that are needed. Accessing and running these reports is described in *Step 1: Running Post Upgrade Reports* on page 102.

## The Post Upgrade Reports for Display Application Conversion

The report for the Display Application phase includes:

- apm.upgrade.display.event (The **Results of displayevent upgrade**)
- apm.upgrade.display.opts (The Results **of displayoption upgrade**)

When SC Upgrade processes the display options, it automatically performs the following actions:

- Copies all current display options with a User Condition defined to a temporary database dictionary called upgradedisplayoption.
- Deletes all current display options that do not have a User Condition defined.
- Loads the new set of SC51 display options into the displayoption file.
- Processes the display options that have been placed in the upgradedisplayoption file.

If both of the display components (display options and display events) were upgraded successfully, you can delete the backup that was created. The backup is saved as display.bak in the same directory that contains your upgrade files.

If you have completed the upgrade and the database option buttons in the system tray have disappeared, the procedure for upgrading the displayoption and displayevent files has failed.

If the display components upgrade procedure failed, load the backup copies of the displayoption and displayevent files onto your *Development* system and run the application again. Refer to *Missing Function Keys* on page 178 in the *Troubleshooting* appendix for more information.

## Resolving Display Application Conflicts

After you have run the report, check the following items to ensure that the customizations were upgraded correctly. See for instructions on how to run a report, see *Step 1: Running Post Upgrade Reports* on page 102.

When processing an upgradedisplayoption file, the system sets the status of each item. The following table lists the possible states and what you should do about them.

Table 5-4: Fields in the Upgrade Display Options and Upgrade Display Events Reports

| Status | Definition | Action |
|---|---|---|
| **Saved** | The displayoption record has been moved to the upgradedisplayoption file, but has not been processed. | If you find a file with this status, contact customer support.<br>See step 2, below. |
| **Moved** | The displayoption record matches an SC51 displayoption and the User Condition from the old option has been moved to the new one. | No action required. See step 3, below. |
| **Added** | The displayoption record does not match any of the new displayoptions, and has been added to the new file. | No action required. See step 4, below. |
| **Not Moved** | The displayoption record does not match any of the new displayoptions, and uses the same Screen ID/Option number as a current option. Therefore, it cannot be added to the new options. | Check why the record was not moved. Decide which new and old features are necessary. Determine what has to be done to resolve the conflict. For example, you might want to add a new graphics option to the old record. See step 5, below. |

### To check the display options or display events report:

1. Open the desired report:
   - **Results of displayoption upgrade** (apm.upgrade.display.opts*)* report or the upgradedisplayoption file
   - Results of displayevent upgrade (apm.upgrade.display.event) report or the upgradedisplayevent file.
2. Check that there are no records with a status of *Saved*.
3. Examine the new display options that have had the user condition *Moved* to ensure that the User Condition is still valid when used by the SC51 applications.

The **Status** field of the old display option contains the unique identifier of the new display option, which may be used to quickly find the proper record.

4 Examine any records that were *Added* to ensure that this option is still valid when used by the new SC51 applications.

The new options should be checked to determine if this option has simply been moved to a new option number.

The unique ID of the *Added* display option is included in the **Status** field.

5 Examine any records that were *Not Moved* to determine whether or not they should be manually added to the new display options using a different option number.

The unique identifier of the display option that caused the conflict is included in the **Status** field.

### The Display application

If the **Display** RAD application has been modified in your ServiceCenter system, the current version of the application should be renamed to `display.old` and the NEW version should be renamed to `display`.

When accessing this application through the RAD editor, the equal sign operator (=) should be used to select only the application in question, such as, either `=display` or `=NEWSC51display`.

**Note:** The Display application provides access to RAD features without requiring RAD programming skills or RAD licensing.

For more information on the Display application, refer to the *Display Application* section in the ServiceCenter *System Tailoring Guide*.

### The Displayoptions file

The `displayoptions` file sets various display options. These options can appear in the Options menu or System Tray buttons in GUI mode, and as F-keys in text mode. RAD applications also can be called from an options definition record.

The SC51 applications have a new set of display options. To install these new options properly into an existing system, the old options must be completely removed. SC Upgrade deletes these old options while the upgrade process is running.

**Important:** To ensure that customized display options are not lost, they must have a **User Condition** specified. If the User Condition field is blank in any displayoption records (no condition is defined), those records are deleted and replaced with the SC51 set of options.

After the display component upgrade is complete, run the **Results of displayoptions upgrade** (apm.upgrade.display.opts*)* report to view the status of the user customized display options.

### Displayevent records

The displayevent table defines the events that a screen handles. After the display component upgrade is complete, run the **Results of displayevent upgrade** (apm.upgrade.display.event*)* report to view the status of the display event conversion.

### Displayscreen records

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

**Important:** Triggers are attached to this file. Changes to the records in the displayscreen file impact their associated display options and events.

When making changes to these records:

- Any changes to the new version of the displayscreen file must be incorporated into the existing version of the file.
- Any changes from the NEWSC51 version of the record must be manually reflected in the original displayscreen record.

## Step 3. Conflict Resolution of the Database Dictionaries

When SC Upgrade upgrades your database dictionaries, it often encounters situations where your version of a database dictionary does not match any previous ServiceCenter versions. For example, this conflict occurs if you added or deleted fields to the problem database dictionary or the contact database dictionary.

When SC Upgrade discovers these conflicts, it merges your database dictionary with the database dictionary included with the upgrade. The resulting database dictionary contains all the fields in your original database dictionary, as well as any fields from the upgrade's database dictionary that were not in yours.

SC Upgrade records these combined database dictionaries with a status of MERGED. The MERGED status does not indicate an error, it indicates that changes may have been made to that database dictionary. Database dictionaries that have been mapped to SQL always appear as MERGED, as the mapping data within each table is unique and cannot be compared to base ServiceCenter data. The MERGED status does not indicate an error, but simply states that changes may have been made to that database dictionary.

When database dictionaries are merged and the new version of a field is a different type than the old version, the old field type is kept. Peregrine recommends that you modify the new fields to the expected field type.

The field types of both the old and new database dictionaries in question can be found in the process.log file. If you have any questions or problems, call Peregrine Systems Customer Support.

### Post Upgrade Reports for Database Dictionary Conversion

SC Upgrade ships with three reports that are designed to provide you with feedback of its activities in the Database Dictionary and Data phases of the upgrade.

The report for the Database Dictionary and Data phases include:

- apm.upgrade.results.exceptions (The **Service Pack Exception Report**)
- apm.upgrade.results.full (The **Service Pack Full Upgrade Report**)
- apm.upgrade.job.log (The **Service Pack Job Log**)

For information on how to run the reports, see *Step 1: Running Post Upgrade Reports* on page 102.

### Resolving Database Dictionary conflicts

When processing an apm.upgrade.results file, the system sets the status of each item. The following table lists the possible states and what you should do about them.

Table 5-5: : Upgrade Results found in the apm.upgrade.results file

| Upgrade result | Definition | Action |
|---|---|---|
| **Merged** | The database dictionary from SC Upgrade was merged with the database dictionary in your system. | No action required. |
| **Added** | The SC Upgrade object was added to your system because you did not have an existing version. | No action required. |
| **Error** | The ServiceCenter object could not be upgraded. | Figure out why it had an error and then fix the problem. If you have trouble doing this, contact customer support. |

#### Data Policy Changes

If you modified a data policy, SC Upgrade usually cannot upgrade the associated data policy record. In these situations, the current version of the database dictionary should be kept, as it is automatically updated to reflect the current state of the database dictionary. The *NEW* version of the database dictionary should be investigated for any changes to data policy that you may want to move to the current database dictionary record.

## Step 4: Conflict Resolution of the Data

The SC Upgrade utility upgrades any objects in your file system that it recognizes as being unmodified Peregrine Systems source code. When SC Upgrade encounters code that has been modified, the upgrade either copies the new version of the application as NEW*<release><object name>* (for example, *NEWSC51apm.first*) or renames your version out of the way as OLDSC51*<object name>* (for example, *OLDSC51pm.main*). The method used depends upon the choice you made when starting the upgrade process.

**Note:** This naming convention only indicates that your version of the object was marked during the upgrade to SC51 and does not reflect the version from which you are upgrading.

As the upgrade administrator, you must examine these conflicts and make a decision to either retain your existing version of the object in question, or accept the new object in lieu of what exists on the current system. Most objects, such as forms, Format Control records, or validity records, are relatively easy to check. Examine the old object first, and then the new one. Applications are more complicated, and require more than a glance to compare.

When evaluating conflicts in RAD applications where the name of the application starts with apm.upgrade, the existing application must be kept. In this case, delete the NEWSC51 versions of the applications.

**Warning:** RAD applications named after the pattern apm.upgrade.* are Peregrine-reserved applications. Changing them may result in serious problems.

Another RAD application to avoid changing is macro.build.field.away. If it has been modified, revert to the Peregrine version.

## Post Upgrade Reports for Data Conversion

SC Upgrade ships with three reports that are designed to provide you with feedback of its activities in the Database Dictionary and Data phases of the upgrade.

The report for the Database Dictionary and Data phases include:

- apm.upgrade.results.exceptions (The **Service Pack Exception Report**)
- apm.upgrade.results.full (The **Service Pack Full Upgrade Report**)
- apm.upgrade.job.log (The **Service Pack Job Log**)

For information on how to run the reports, see *Step 1: Running Post Upgrade Reports* on page 102.

## Resolving Data Conflicts

When processing an apm.upgrade.results file, the system sets the status of each item. The following table lists the possible states and what you should do about them.

**Table 5-6: : Upgrade Results found in the apm.upgrade.results file**

| Upgrade result | Definition | Action |
|---|---|---|
| **Renamed** | The object from SC Upgrade was renamed to *NEWSC51<object.name>* and added to your system. | Do one of the following actions:<br>■ **Keep the old version:** Do nothing<br>■ **Keep the new version:** Rename the old version and give the new version the original name of the old version.<br>■ **Merge new and old versions:** Add the new features to the old version.<br>**Note:** Always make your changes in the "old" record 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. |
| **Forced** | Your object was renamed *OLDSC51<object.name>* and the SC Upgrade object was added as *<object.name>*. | Do one of the following actions:<br>■ **Keep the old version:** Do nothing<br>■ **Keep the new version:** Rename the old version and give the new version the original name of the old version.<br>■ **Merge new and old versions:** Add the new features to the old version.<br>**Note:** Always make your changes in the "old" record 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. |
| **Added** | The SC Upgrade object was added to your system because you did not have an existing version. | No action required. |
| **Error** | The ServiceCenter object could not be upgraded. | Figure out why it had an error. Fix the problem. If you have trouble doing this, contact customer support. |

### Application Conflicts

Application conflicts usually occur only when there have been changes to the RAD code in the old version. This can occur when Peregrine sends you a patch, or when your system is altered by new RAD programming.

To resolve application conflicts:

- If your company does not have a RAD license, always select the new version.
- If your company has a RAD license, determine what change was made to the old system and what the new version provides and do one of the following:
  - Keep the old version.
  - Keep the new version.
  - Merge new and old versions.

### Application dependencies

Peregrine Systems RAD language is often used by one application to call another application. For example, the Change Management `cm3r.main` application calls a second application, `cm3r.update`, to actually manage the updating of change records. Consequently, the successful functioning of `cm3r.main` depends on certain expected behaviors in `cm3r.update`.

From an upgrade standpoint this is important, because it is possible to upgrade `cm3r.main` but not `cm3r.update` (or the other way around). Upgrading one application and not the other has the potential to cause two different types of problems: the number of *Parameter conflicts in a sub-application* on page 114 and *Logical dependency conflicts* on page 115.

#### Parameter conflicts in a sub-application

The most common type of dependency problem arises when the number of parameters in a sub-application changes. For example, `cm3r.update` changes from having four parameters to having five parameters. The parent application was compiled expecting to pass four parameters to `cm3r.update`, but the child application now expects five. Because of this discrepancy, the parent application cannot function properly.

To help you examine the differences between the new and old applications, Peregrine Systems has bundled the RAD Comparison Utility with SC Upgrade. This utility examines new and old versions of an application and displays a list of differing elements (in other words, panels and lines that do not match). For detailed instructions on how to use this utility, see *RAD Comparison Utility* on page 153. See also *Application Changes for this Release* on page 116.

- To resolve parameter conflicts, contact Peregrine Systems' Customer Support.

**Logical dependency conflicts**

The second type of application dependency is less common but more difficult to correct. *Logical* dependency usually occurs when a new feature, required by a parent application to run properly, is added to a child application. If the parent application is upgraded, but the child application is not, the parent application cannot perform correctly, since its child application is not performing as expected.

- Resolving logical dependencies usually requires either reverting the parent application to the previous version or upgrading the child application to the latest version.

## Special instructions

### Globallists

Changes to the **Regen Every** field causes the globallist in question not to upgrade.

- If comparison to the NEW version of the globallist shows no other changes, the current version should be kept.

### Altered validity table entries:

ServiceCenter uses a binary sort order to process validity table entries, based on the sequence number of the entries. A validity entry with a sequence number of 1 is processed before an entry with a sequence number of 2. Validity entries with a sequence number of NULL are processed before either sequence number 1 or 2, since NULL is the first entry in a binary sort.

Most users who implement multiple validity entries tend to 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 ServiceCenter's default validity entries have sequence numbers of NULL.

When SC Upgrade updates your validity entries, it adds ServiceCenter's default validity lookup entries, if you have not created 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, SC Upgrade adds the new entry to your file system.

Although SC Upgrade has not removed any of your code, it has added a validity entry which gets processed before yours. As a result, the first validity lookup you see is a default validity entry, rather than one of your customized entries.

For more information on validity, refer to the ServiceCenter *System Tailoring Guide*.

**To remove the invalid validity table entries that SC Upgrade added:**

1. Type validity on a command line.
2. Press **Enter** to display the Validity Table Specifications form.
3. Enter the name of the file or format in question in the **Files/Formats** field.
4. Click **Search** (or press **F6**).
5. Select a **Field Name** from the QBE list displayed.
6. Look for an entry with a sequence number of NULL. Verify that this entry was added by SC Upgrade and is not an important part of your validity processing.
7. To remove the invalid validity record, click **Delete**.

   The next time validity is processed, it starts with your lowest sequence numbered validity entry.

# Application Changes for this Release

## Changes in Change Management

ServiceCenter 5.1 contains improvements to streamline the set-up and administration of Change Management. Most of the updates are applied in the background during the upgrade process, and do not require intervention to implement.

**Note:** Alerts, approvals, phases, and categories are updated by the upgrade. However, the operation of these functions has not changed. The only areas of Change Management where you may be required to do manual modifications are in the display screens and the display options that have been customized for your system.

This section provides an overview of the updates to Change Management and what changes, if any, the end user sees.

For details on using the Change Management application, see the ServiceCenter *User's Guide*.

The following records are updated in Change Management:

- Change category records
- Change phase records
- Task category records
- Task phase records
- Message group records
- Profile records
- Profile group records

## Alerts

Two alert files were moved to new files in ServiceCenter 5.0. These alert files function the same as in previous releases, but they are simply contained in different files.

Definitions are contained in the AlertDef file. In previous versions, the definitions were in the cm3ralerts and cm3talerts file.

Alert logs are located in the Alertlog file. In previous versions, the logs were not stored.

### To access alert definitions:

1. Access Change Management.
2. Go to the **Maintenance** tab.
3. Select the **Alerts** option.

   The names of these new records are prefixed either with *Change* - or *Task* -, depending on the file from which they originated.

## Approvals

Approval definitions have been added in ServiceCenter 5.0. The definitions are stored in the ApprovalDef file. In previous versions, the definitions were part of the phase records.

The new records for the approval definitions are based on the existing approvals already in the system. The upgrade process automatically moves the old definitions to the new file. The same approval rules are simply stored in a different file.

- Approval definitions are based on the same phase name. This applies to changes and tasks.
- *ApprovalDef* records are created for each phase and message group.
- The upgrade updates the phase definitions to reflect the approval definitions.
- The *(CS)* approval requirement is replaced with *Change Sponsor Approval* in each change SLA record.

The approvals log is now located in the **Approvallog** file. In previous versions, the logs were not stored.

### To access approvals:

1. Access Change Management.
2. Go to the **Maintenance** tab.
3. Select the **Approvals** option.

   The Approvals option was added in ServiceCenter 5.0.

## Phases

ServiceCenter 5.0 adds the following changes to phases:

- The Approvals array now holds the name of the phase.
- References to *$filer*, *$filet* and any *$cm3** variables are replaced with local variables.
- Fields are no longer referenced by the structure name (such as *header* or *middle*) in formats and Format Control.
- Each element in the Alerts array is prefixed with either *Change* - or *Task* -.

### To access phase definitions:

1. Access Change Management.
2. Go to the **Changes** tab.
3. Select the **Change Phases** option.

### Variables

In ServiceCenter 5.0, the upgrade of Change Management changed references to the following variables to *$L.file* — *$filer* and *$filet* and *$cm3**.

### Display screens

You may need to manually update Change Management display screens that were customized.

For details on using the Display application, see the ServiceCenter *System Tailoring Guide*.

### Changes/Tasks

The unique ID (number field) is converted to a character string. This change allows prefixes and suffixes to be added to the ID.

Alerts are scheduled and any applicable Approval and ApprovalLog records are created.

### Profiles

ServiceCenter 5.0 adds two default category fields: *Default Change Category* and *Default Task Category*. Depending on which file it came from, the former *Default Category* is stored in either field.

## Changes in Incident Management

The following records are updated in Incident Management:

- Category records
- Assignment groups

The following updates were made to Incident Management in ServiceCenter 5.0:

- The probsummary file is viewed and updated instead of the problem file.
- The upgrade automatically examines the problem file and adds the necessary fields to probsummary.
- The upgrade updates the build.problem.summary link record with these fields.

You may want to check your database dictionaries to ensure that the fields have been updated.

- Additionally, the dependence on structure names (such as *header* and *middle*) is no longer needed by formats and Format Control records.

No manual steps are required for this portion of the upgrade.

## Changes in Request Management

**Note:** If you are upgrading from ServiceCenter 4, you can ignore this section on Request Management. If you are upgrading from a version prior to ServiceCenter 4, continue with this section.

ServiceCenter 4.0 contained improvements to streamline the set-up and administration of Request Management. Most of the changes are applied in the background during the upgrade process, and do not require intervention to implement. A stockroom feature was also added for catalogs.

**Note:** The catalog, alerts, approvals, phases, and categories are updated by the upgrade. However, the operation of these functions has not changed. The only areas of Request Management where you may be required to do manual modifications are in the display screens and the display options that have been customized for your system.

This section provides an overview of the updates to Request Management and what changes, if any, the end user sees.

For details on using the Request Management application, see the ServiceCenter *Request Management Guide*.

### Catalog and Model Supporting Files

Starting with ServiceCenter 4.0, the components for a part are now displayed. In previous ServiceCenter versions, only a part's parent devices were listed in the Request Management component definition records and the model records. Now all the components associated with the part are listed.

**To access the Catalog and Model records:**

1. Access Request Management.
2. Go to the **Maintenance** tab.
3. Select the **Supporting Files** option.

4 Go to the **Catalog** tab.

5 Select either the **Catalog** or **Model** option.

6 Access a record.

- Under Catalog, the components are listed on the Components tab and the Dependencies are listed on the Dependencies tab.
- Under Model, go to the Catalog tab. The Components and Dependencies tabs are there.

The catalog operates the same as in ServiceCenter 3, but the presentation now contains more information.

The upgrade utility looks at a component's parent, level, and sequence when it displays a record.

The components are grouped. In the group field of the Components tab, the name of the group to which the part is linked is listed in a format of level.sequence. For example, group name 2.1 defines the first part in the second group. The Dependencies tab forms a grouping order that creates dependencies, according to how groups are set up, and sets up dependency types.

### Stockrooms

In ServiceCenter 4, stockrooms were created to allow each part in the catalog to be stored in a separate stockroom. Stockrooms are created for each location listed in ServiceCenter.

The ServiceCenter 4 installation or the ServiceCenter Upgrade (for pre-SC4 systems) creates a stockroom for each location. Multiple locations can be manually associated with one stockroom. See *Creating Stockrooms for Multiple Locations (for pre-SC4 systems only)* on page 81.

### Alerts

Two alert files were moved to new files in ServiceCenter 4.0. These alert files function the same as in previous releases, but are simply contained in different files. Definitions are contained in the AlertDef file. In previous versions, the definitions were in the ocmalertpool file.

The alerts log is located in the Alertlog file. In previous versions, the logs were located in the ocmalertlog file.

### To access the alert files:

1 Access Request Management.

2 Go to the **Maintenance** tab.

3 Select the **Supporting Files** option.

4 Go to the **Support** tab.

5 Under the **Alerts** structure, select either the **Definitions** or **Alert Logs** option.

A Current Alerts option is also included.

## Approvals

Two approval files were moved to new files in ServiceCenter 4.

Definitions are now contained in the ApprovalDef file. In previous versions, the definitions were in the ocmapprpool file.

The new records for the approval definitions are based on the existing approvals already in the system. The upgrade process automatically moves the old definitions to the new file. The same approval rules are simply stored in a different file.

- Approval definitions are based on the same phase name. This applies to quotes, line items, and orders.
- Definitions are created for any catalog items that require specific approvals.
- The upgrade updates the phase and catalog definitions to reflect the approval definitions.

The approvals log is now located in the Approvallog file. In previous versions, the logs were located in the ocmapprlog file.

### To access the approval files:

1 Access Request Management.

2 Go to the **Maintenance** tab.

3 Select the **Supporting Files** option.

4 Go to the **Support** tab.

5 Under the **Approvals** structure, select either the **Definitions** or **Approval Logs** option.

A Current Approvals option was also added in ServiceCenter 4.

## Quotes

A new field has been added to the quote category record. Multiple selections allows you to select multiple components before a request is opened. If you prefer to prevent users from selecting multiple components, enter false in the field.

## Phases - Quotes and Orders

Phases can be further defined as of ServiceCenter 4.

Starting with ServiceCenter 4, quote and order phases can be set to use line item and model approvals. The **Use Line Item/Model Approvals** field is located under Approval Controls in the Approvals tabs for quote and order phases. Set the field to **false** if you do not want the phase to have line item or model approvals.

The Scripts/Views tab in phases for quotes also includes two open options for running scripts.

- **Pre Catalog Open** runs the selected script *before* an item is selected from the catalog.
- **Post Catalog Open** runs the selected script *after* an item is selected from the catalog.

## Variables

In ServiceCenter 4, the upgrade of Request Management changed references to the following variables to *$L.file* — *$fileq*, *$fileo,* and *$filel.*

## Display Screens

You may need to manually update display screens that were customized. The naming convention for these screens changed with ServiceCenter 4. The prefix of *ocm* was replaced by *rm*. For example, *ocm.main.display* became *rm.main.display* in ServiceCenter 4.

Any display options added to *ocm* screens in previous versions need to be manually checked in the comparable *rm* screens. If these options were not moved to the new screen, you will need to manually move them.

For details on using the Display application, see the ServiceCenter *System Tailoring* guide.

## Updating Database Dictionary Fields for Request Management

A key in the Database Dictionary must be manually updated for Request Management. A key in the ocml record must be modified.

For more information on using the Database Dictionary, see the ServiceCenter *System Tailoring* guide.

### To update a key:

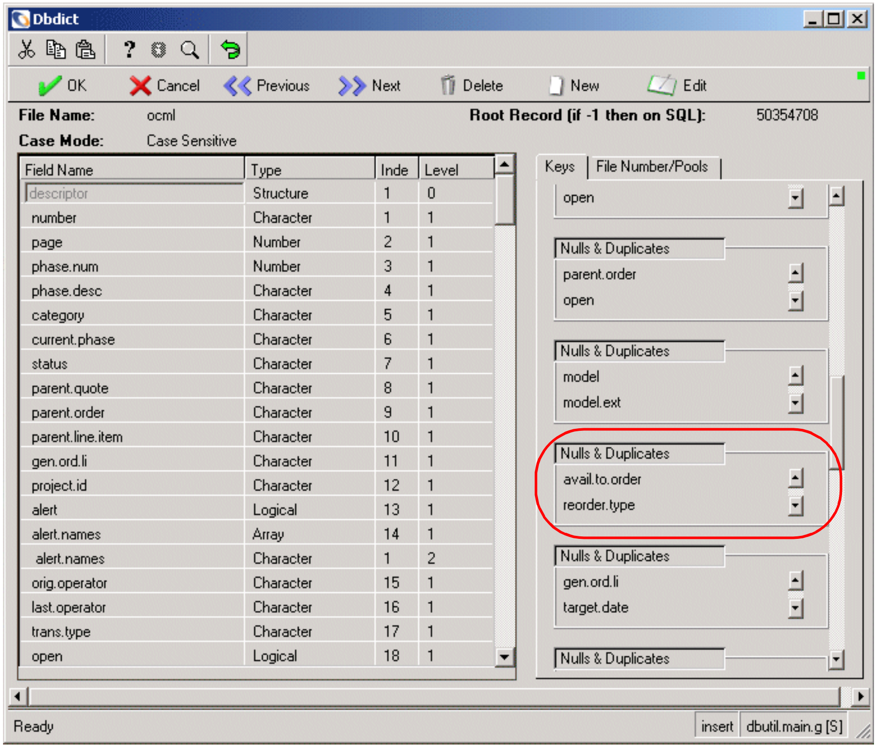
1 Click the **Database Dictionary** button in the **Toolkit** tab of the ServiceCenter Home menu, or type dbdict on a command line and press Enter.

The Database Dictionary prompt form is displayed.

2 Type ocml in the **File Name** field to search for all database dictionaries beginning with **ocml**.

3 Click **Search** or press **Enter**.

The ocml records are displayed. See Figure 5-20 on page 124.



**File Name:** ocml

**Root Record (if -1 then on SQL):** 50354708

**Case Mode:** Case Sensitive

| Field Name | Type | Inde | Level |
|---|---|---|---|
| descriptor | Structure | 1 | 0 |
| number | Character | 1 | 1 |
| page | Number | 2 | 1 |
| phase.num | Number | 3 | 1 |
| phase.desc | Character | 4 | 1 |
| category | Character | 5 | 1 |
| current.phase | Character | 6 | 1 |
| status | Character | 7 | 1 |
| parent.quote | Character | 8 | 1 |
| parent.order | Character | 9 | 1 |
| parent.line.item | Character | 10 | 1 |
| gen.ord.li | Character | 11 | 1 |
| project.id | Character | 12 | 1 |
| alert | Logical | 13 | 1 |
| alert.names | Array | 14 | 1 |
| alert.names | Character | 1 | 2 |
| orig.operator | Character | 15 | 1 |
| last.operator | Character | 16 | 1 |
| trans.type | Character | 17 | 1 |
| open | Logical | 18 | 1 |

open

Nulls & Duplicates
parent.order
open

Nulls & Duplicates
model
model.ext

Nulls & Duplicates
avail.to.order
reorder.type

Nulls & Duplicates
gen.ord.li
target.date

Nulls & Duplicates

Ready | insert | dbutil.main.g [S]

**Figure 5-20: OCML Record**

4. Select the ocml record in the QBE list.
5. Locate the key that begins with the field **avail.to.order**.
6. Position the cursor on the key type *Nulls&Duplicates* above the key field of avail.to.order.
7. Click **Edit**.
8. In the fields list, ensure that *only* the following fields are included:

   avail.to.order,<br>
   open<br>
   target.order<br>
   reorder.type<br>
   quantity.balance

9. Delete any other field names.
10. Click **OK** (the check button) to save the changes.
11. Click **OK** in Database Dictionary record.
12. A prompt is displayed that states: **You are about to Regen your file. Do you wish to continue?**
    - Click OK to run the regen.
    - Click the **Schedule** button to set a different time to run the regen.
13. Click **OK.**

    When the regen successfully completes, you are returned to the Database Dictionary prompt. A message is displayed in the status bar that states the regen is finished.

    If the regen does not successfully complete, contact Peregrine Customer Support.

# 6 CHAPTER Building a Custom Upgrade

This section describes how to build a custom upgrade, and apply it to your *Production* system.

If you have followed all the steps to this point, you have already run SC Upgrade against your *Development* system and have performed conflict resolution on that system. You have also tested that system to verify that everything is working properly, and are now ready to create the custom upgrade that will be used to upgrade your *Development* system.

This chapter includes:

- *Preparing to Build the Custom Upgrade* on page 128 — discusses how to allocate disk space and lists the upgrade files, including their sizes, for Windows, OS/390, and Unix.
- *Building the Custom Upgrade* on page 131 — describes how to build the custom upgrade and tells you what additional files need to be copied to your system.
- *Testing Your Custom Upgrade* on page 137 — discusses how you should test the custom upgrade.
- *Upgrading Your Production System* on page 137 — discusses training users and applying the custom upgrade to your *Production* system.
- *Purging Upgrade Files* on page 139 — discusses how to purge the upgrade files from your *Development* system.

# Preparing to Build the Custom Upgrade

## Allocating Disk Space

SC Upgrade automatically exports the new upgrade for you; however, you must have a location ready to where the upgrade can be placed. You need sufficient free hard drive space to build the upgrade on the server. When deciding how much space to allocate to the upgrade, consider the amount of customization your system has undergone.

Use one of the following formulas to compute the amount of hard drive space you will need to allocate for the upgrade.dta file:

- For the system as shipped:

  **10 KB per application + 2 KB per form (format) + 40 MB**

  There are 2,000 applications in ServiceCenter, using 20 MB of space. The approximately 27,000 forms make up another 45 MB.

  This formula should be adequate unless your system has large, complex forms and/or applications, or unless you have added substantially to any of the files listed in the patch record (for example, formatctrl, knowledge, menu, or link). If that is the case, use the option for customized systems.

- For customized systems:
  - 150 MB of free drive space for moderate customization.
  - 200 MB of free drive space for heavy customization.

## Allocating Disk Space on an OS/390 System

SC Upgrade attempts to allocate space for its new upgrade automatically by using the Database Create/MVS config record in your file system. Most of these config records do not allocate sufficient space for all of the SC Upgrade files. Peregrine Systems recommends that you use the upgalloc job provided in the cntl library to automatically allocate the datasets for the upgrade process.

When the cntl dataset is unloaded from the ServiceCenter OS/390 Upgrade cartridge, PDS member upgalloc needs to be modified and executed to allocate space for all SC upgrade files.

### Make the following modifications in the JCL:

1 Modify the **PREFIX** to match your dataset high level qualifier for your upgrade process.

2 Modify the **DVOLSER** to match a desired DASD location where your upgrade datasets will be allocated.

```
//      REGION=4096K
//*****************************************************************
**
//*
//* THIS IS THE JCL REQUIRED TO ALLOCATE THE DATASETS FOR THE
//* UPGRADE PROCESS.
//*
//* BEFORE RUNNING THIS JOB YOU MUST:
//*  1) MODIFY THE JOB CARD TO MEET YOUR SITES SPECIFICATIONS
//*  2) MODIFY THE PREFIX VARIABLE ON THE PROC STATEMENT
//*    TO SPECIFY THE CORRECT HIGH LEVEL QUALIFIER (NOTE1)
//*  3) MODIFY THE APPLLEV VARIABLE ON THE PROC STATEMENT
//*    TO SPECIFY THE CORRECT APPLICATION LEVEL  (NOTE2)
//*  4) MODIFY THE DVOLSER VARIABLE ON THE PROC STATEMENT
//*    TO SPECIFY THE VOLSER OF THE TARGET DISK (NOTE3)
//*
//****************************************************************
//*
//ALLOC  PROC PREFIX='XXXX',<===NOTE1
//      APPLLEV=A9802,<===NOTE2
//      DVOLSER=XXXXXX<===NOTE3
//*
//****************************************************************
//*
```

If you plan to use external logging, you must also preallocate upgrade.log and process.log. You need a minimum of two (2) MB for process.log, and ten (10) MB for upgrade.log.

If you decide to log messages internally, ensure there is sufficient space in both the scdb.db1 and scdb.asc files. Peregrine Systems recommends expanding these files by 10 MB and 2 MB respectively.

## Upgrade Files for Windows, Unix, and OS/390

Refer to the following list of file names and their approximate sizes:

| File | Size |
|---|---|
| detail.log | 2 MB |
| preupg.bin | 200 KB |
| sql.upgrade.unl | 100 KB |
| transfer.bin | 2.5 MB |
| upgdbdct.dta | 500 KB |
| upgdisp1.dta | 5 KB |
| upgdisp2.dta | 500 KB |
| upglang.unl | 300 KB |
| upgnew.dta | 5 KB |
| upgrade.dta | 100 MB |
| upgrade.inf | 20 MB |
| upgrade.log | 3 MB |
| upgrade.mak | 10 KB |
| upgrade.str | 1 MB |
| upgrade.ver | 1 KB |

To provide a guideline for allocation requirements, refer to the file size estimates above, then increase those allocations as follows:

| Customization Level | Allocation Increase |
|---|---|
| lightly customized systems | Increase the allocations for **upgrade.dta**, **upgrade.str**, and **upgrade.inf** by 25 percent. |
| moderately customized systems | Increase the allocations for **upgrade.dta**, **upgrade.str**, and **upgrade.inf** by 50 percent. |
| heavily customized systems | Increase the allocations for **upgrade.dta**, **upgrade.str**, and **upgrade.inf** by at least 100 percent. |

As a precaution, designate a sufficient secondary memory allocation for your files of at least 50 percent of your primary allocation. This allocation allows disk space growth if necessary.

**Note:** After you have *successfully* upgraded your *Production* system, the above files can be deleted.

## Modify the User ID

Be sure that the User ID to run ServiceCenter has read and write access to your intended destination directory.

# Building the Custom Upgrade

This section provides instructions for building the custom upgrade for your system. It also provides information about the additional files that are needed and how to purge the upgrade files from your system when you have completed the process.

## How to Build the Custom Upgrade

Use the ServiceCenter Upgrade Builder wizard to create a custom upgrade.

### To build the custom upgrade:

1. If you are not logged on, log on to your *Development* system with an express client.
2. If you have not already done so, create a new directory for the custom upgrade files. (Call this directory *CustomUpgrade*).

   If you create a new, custom upgrade, it does not delete existing files in the target directory, but appends to them.
3. Give ServiceCenter the rights to create files and write to files in the destination directory.
4. Open the Upgrade menu:
   - In GUI mode, enter `SC51upgrade` on a command line.
   - In text mode, enter `SC51upgradetext`.
5. Press **Enter**.

   The Upgrade menu is displayed.

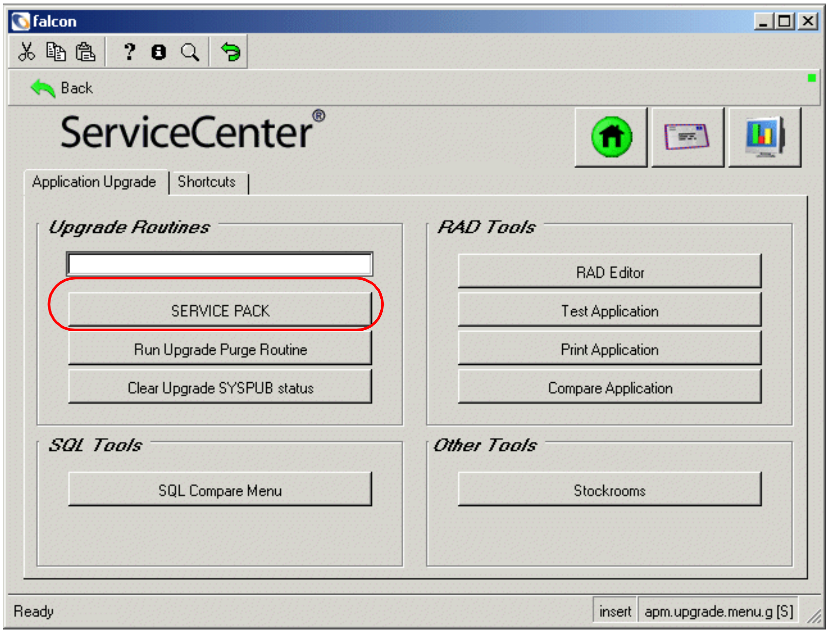
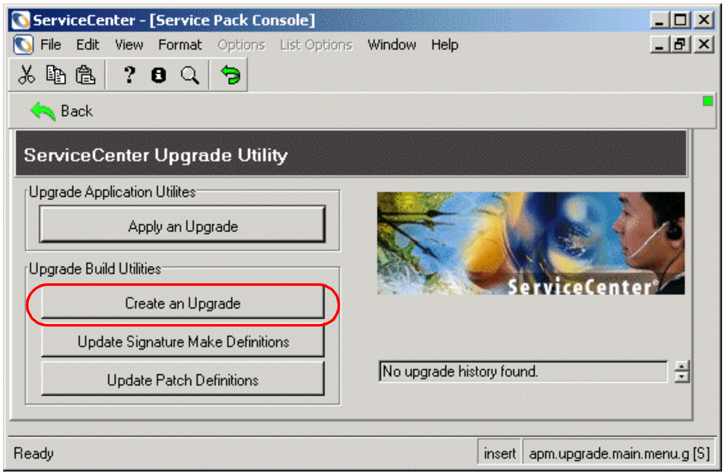


Figure 6-1: SC Upgrade Menu

6 Click the **SERVICE PACK** button in GUI mode, or select **Service Pack Console** (or press **F6**) in text mode.

The ServiceCenter Upgrade Utility console is displayed.

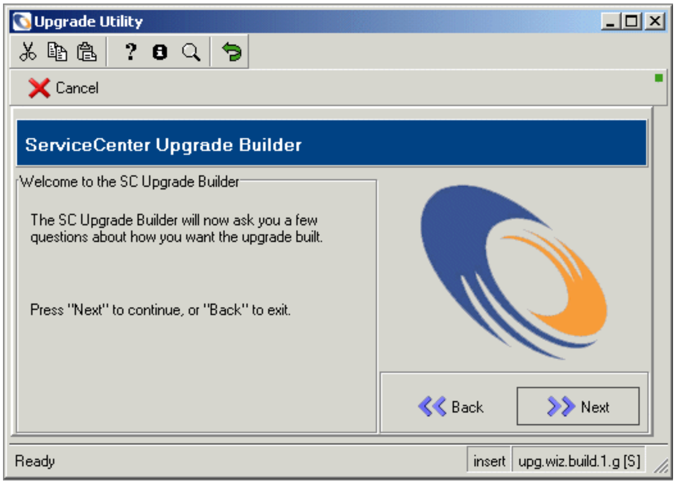

ServiceCenter - [Service Pack Console]
File
Edit
View
Format
Options
List Options
Window
Help
Back
ServiceCenter Upgrade Utility
Upgrade Application Utilites
Apply an Upgrade
Upgrade Build Utilities
Create an Upgrade
Update Signature Make Definitions
Update Patch Definitions
ServiceCenter
Ready
insert
apm.upgrade.main.menu.g [S]

No upgrade history found.

Figure 6-2: The Upgrade Console

7 Click the **Create an Upgrade** button, or press **F4** in text mode.

The Peregrine Upgrade Builder is launched.



Welcome to the SC Upgrade Builder

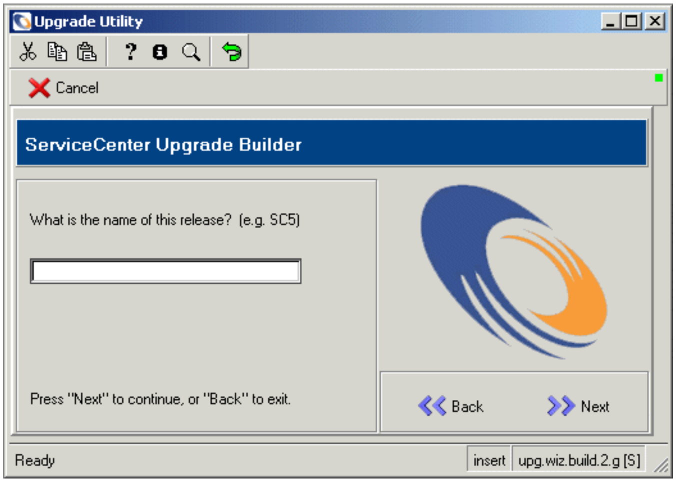
The SC Upgrade Builder will now ask you a few questions about how you want the upgrade built.

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

**Figure 6-3: Peregrine Upgrade Builder**

8 Click **Next**.

You are prompted for the name of the release.



What is the name of this release? (e.g. SC5)

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

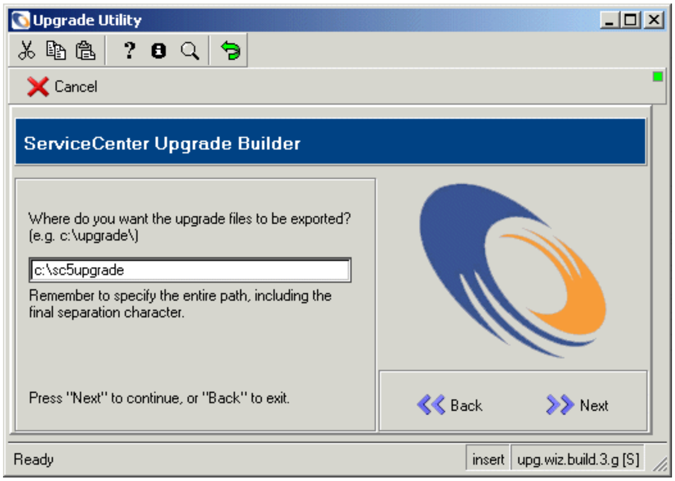
**Figure 6-4: Name the Release**

9 Choose a name to describe your current release level. For example, <*mycompany*> NEW is appropriate, such as, *Peregrine NEW*.

10 Enter this name into the field labeled **What is the name of this release?**

11 Click **Next.**

A prompt asks for a location to export the upgrade files.



Where do you want the upgrade files to be exported? (e.g. c:\upgrade\)

c:\sc5upgrade

Remember to specify the entire path, including the final separation character.

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

Ready | insert | upg.wiz.build.3.g [S]

**Figure 6-5: Locate a Path**

**Warning:** Do not place the custom upgrade files in the same directory as the SC Upgrade Utility.

12 In the field labeled **Where do you want to put the upgrade files to be exported?**, enter the fully qualified path to the directory where SC Upgrade can create and output its export files (See step 2 on page 131).

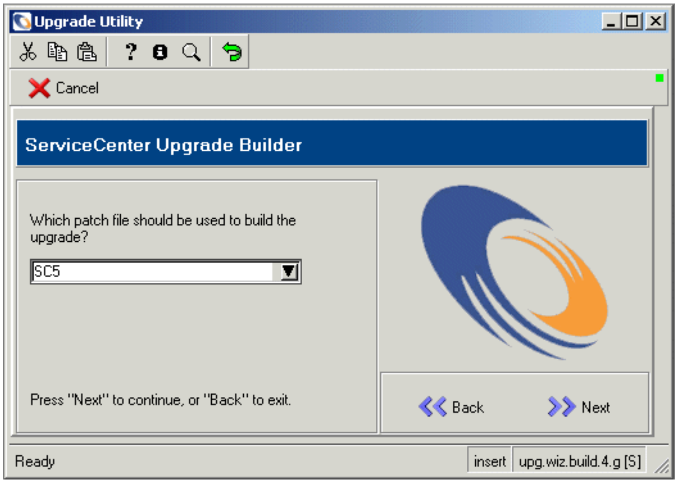
This path must contain a final directory qualifier:

- Unix — forward slash: /
- Windows — back slash: \
- OS/390 users must enter in the high level qualifier (.) when allocating the dataset described above (for example, <.`UPGRADE`.>.

**Note:** The directory must already exist, as ServiceCenter will not create a new directory.

13 Write down the path you have established for export files. Click **Next.**

A prompt provides build options for building the upgrade.



Which patch file should be used to build the upgrade?

SC5

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

Figure 6-6: Select a Patch File

14 Select a patch file from the drop down menu.

Since this is the ServiceCenter 5.1 upgrade, select SC51.

15 Click **Next**.

You are prompted to select the next step for the process.

16 In the structure labeled *Take Which Action?*:

- Select **Complete Upgrade Build** in GUI mode.
- In text mode, type the all command.

**Note:** The other four options correspond to the specific actions that occur when creating the upgrade. Although these steps may be run one at a time in order, it is easier to run the complete upgrade build using the all command. The other options may be used if it becomes necessary to work with Peregrine Systems Customer Support in diagnosing a specific problem with the upgrade build.

17 Click **Next**.

You are asked if you would like SC Builder to use internal logging.

18 Select the type of logging:

- Click **Yes** to activate internal logging.
- Click **No** to deactivate internal logging.

19 Click **Next**.

ServiceCenter Upgrade Builder informs you that it is ready to build the upgrade.

20 Click **Next**.

A warning message is displayed stating: **This process will destroy any existing upgrade definitions on file. Proceed?**

- Click **Yes** to proceed. This destroys any upgrade definitions currently on file and creates the custom upgrade for this system.
- Click **No** to cancel.

**Note:** This process can take anywhere from 30 minutes to several hours, depending upon the speed and load of your server.

The ServiceCenter Upgrade Builder monitors the signaturing process of the build.

21 To further monitor the status of the upgrade, start a ServiceInfo client by following the instructions in *Monitoring the Progress of the Application Upgrade* on page 148.

The time required to build the custom upgrade is dependent on the size and customization of your system.

When the signaturing is complete, ServiceCenter Upgrade Utility main menu is displayed with a message that states: **Finished creating transfer files for the upgrade.**

## Additional Files Needed

Complete the following procedure after the system builds the custom upgrade:

- Copy the upglang.unl file (from the original upgrade) to the new directory defined in step 12 on page 134.

**Note:** If your original system is version A9902, you do not need to copy the upglang.unl file, as it will not be used in the upgrade.

# Testing Your Custom Upgrade

Before you can apply the custom upgrade to your *Production* system, you must apply it to a *Test* system. You will repeat procedures that you have done earlier in the upgrade process, but this time using the custom upgrade instead of the one Peregrine provided.

### To test the custom upgrade:

1. Create the *Test* system from a copy of a fresh backup of the *Production* system. See *Preparing the Development System* on page 55.
2. Apply the custom upgrade that you created in *Building the Custom Upgrade* on page 131 to your production file system.
3. This process is identical to the one you followed in upgrading your *Development* system. Go back to *Upgrading Your System* on page 69, and using the procedures described there, and the upgrade files that you created when you built the custom upgrade, apply the custom upgrade to the *Test* system.

**Important:** Since you have already completed your conflict resolution and other steps, this custom upgrade should apply smoothly with few, if any, exceptions. If this is *NOT* the case, further Conflict Resolution is necessary.

When you have a Custom Upgrade that works smoothly, then apply it to the *Production* system.

# Upgrading Your Production System

This section discusses the final stage of the upgrade process — implementing the upgrade on your *Production* system.

To upgrade your *Production* system, you must consider two items:

- Training your users on new features.
- Applying the upgrade to the *Production* system.

## Train Your Users on Updated Applications

Before implementing the upgraded system into production, users need to be trained on any new features that they might be using. To learn about the new features, refer to the ServiceCenter Release Notes. The release notes list the new features and identify where the new features are described in the ServiceCenter guides.

## Apply the Upgrade to Your Production System

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

**Important:** Do not apply any upgrade to your productions system that has not first been thoroughly tested.

When Upgrading the *Production* system, be aware that:

- The *Production* system is not available to the users while the custom upgrade is applied.
- Make sure 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, ensure that the FTP is set to *binary* mode.

### To apply the upgrade:

1. Check the disk space availability on the production server.
2. Plan the shutdown of the *Production* system.
3. Advise the users of the shutdown.
4. Shut down the *Production* system.
5. Make a backup of the *Production* system. This step is a safety precaution.
6. If you have not already done so, upgrade the RTE using the SC51 installation media. See *Upgrading the Run-time Environment* on page 150 for instructions.
7. If you are upgrading to a different release (for example ServiceCenter 4.0 and ServiceCenter 5.0), do an IR regen on the new server.

**Note:** This step is not necessary if you are upgrading to different versions of the same release (for example 4.0 and 4.3).

8 Restart the server. Do not let the users log back on.

9 Log on as a system administrator, using an express client.

10 Apply the custom upgrade to the *Production* system, using the custom upgrade files you created while building your custom upgrade.

Use the procedures described in *Upgrading Your System* on page 69.

**Important:** You must complete all the upgrade steps, including the preparation steps, on your *Production* system, using the new upgrade files you created from your *Development* system.

11 Once the upgrade has been successfully applied to the *Production* system, advise the users the system is available.

# Purging Upgrade Files

After you have 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.

**Note:** Do not purge the upgrade files if the upgrade fails. If the upgrade fails, you simply pick up where it left off.

The *apm.upgrade.purge* utility can be accessed from the SC Upgrade main menu. The purge utility is discussed in more detail in *Purging Upgrade Files* on page 139.

**Important:** Do not purge the upgrade files until the upgrade to the *Production* system is complete. It may be necessary to re-create a Custom Upgrade. A Custom Upgrade can only be built again, if the upgrade files have not been purged.

### To run apm.upgrade.purge:

1 Access the upgrade menu.

2 Click the **Run Upgrade Purge Routine** button on the Upgrade main menu.

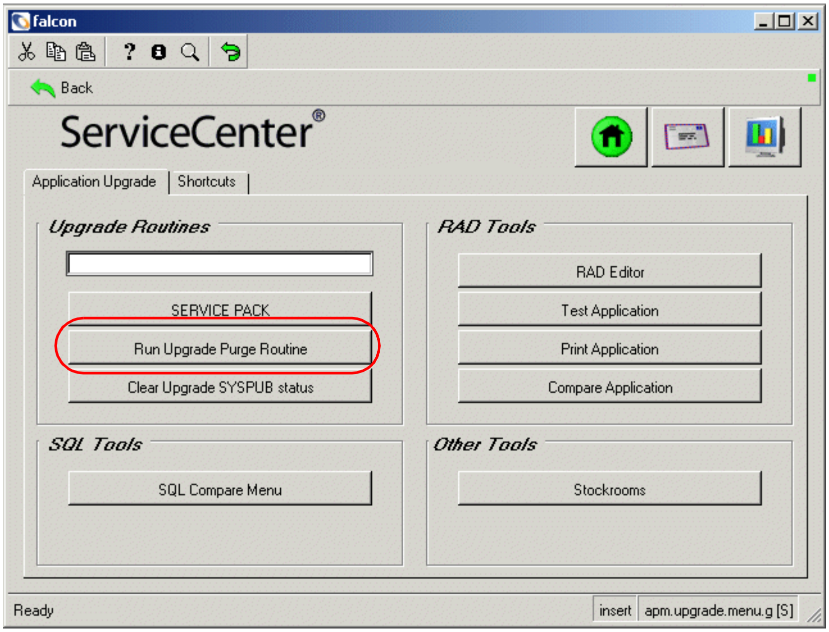
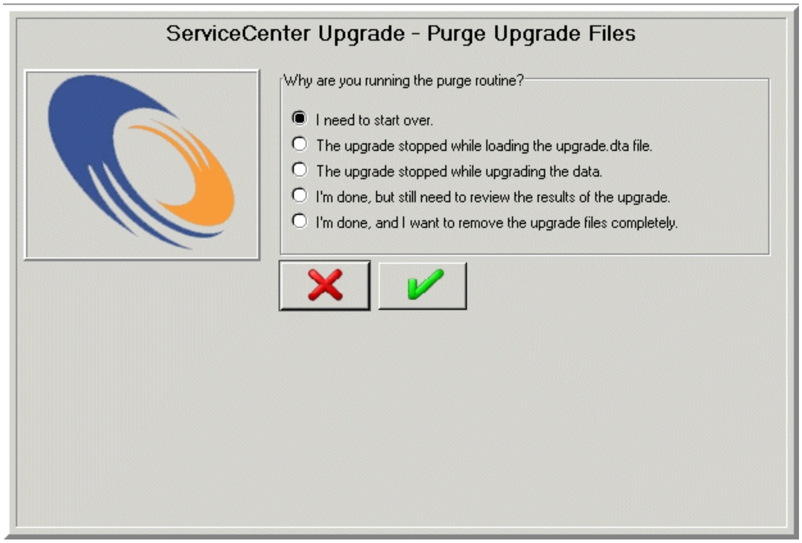


Figure 6-7: Upgrade Main Menu

— or —

Enter *aapm.upgrade.purge on the command line.

3 In the form displayed, select:
**I'm done, and I want to remove the upgrade files completely.**



ServiceCenter Upgrade - Purge Upgrade Files

Why are you running the purge routine?

- ● I need to start over.
- ○ The upgrade stopped while loading the upgrade.dta file.
- ○ The upgrade stopped while upgrading the data.
- ○ I'm done, but still need to review the results of the upgrade.
- ○ I'm done, and I want to remove the upgrade files completely.

Figure 6-8: Purge Upgrade Files Selections

4 Click **OK**.

5 Run SCDBUTIL LFMAP, Option 4.

Although the upgrade data is purged, free space is not reclaimed until you run LFMAP, Option 4.

For instructions on running LFMAP, see the *P4 File System Utility (SCDBUTIL)* section in the *Database Management and Administration Guide*.

# A APPENDIX Procedures You Need to Know to Run the Upgrade

This section contains procedures that you use while running SC Upgrade. These procedures are not performed at this time, but are a precursor to certain steps in the remaining chapters in the book. This section includes steps for loading a file into ServiceCenter and stopping ServiceCenter schedulers.

Topics in this appendix include:

# Loading a File into ServiceCenter

The following instructions explain how to load an external ServiceCenter unload file into your system.

As an example, the following procedure uses the `preupg.bin` file that is included in the upgrade. The `preupg.bin` file is loaded later in the upgrade process.

**Important:** *You will not load this file at this time.* These steps are provided for illustration purposes only.

You can use these steps to load other files, as needed in the process. Simply substitute the name of the file to be loaded for `preupg.bin`.

**Example**

### To load a file into ServiceCenter:

1. Log on to ServiceCenter with an express client that has access to the directory where you loaded the SC Upgrade files. Log on as a system administrator.

   If you loaded the SC Upgrade files onto your ServiceCenter server (recommended installation), one of the following connection methods is recommended:

   - From OS/390 (MVS), connect normally from a terminal.
   - From Unix, run `scenter` or `scenter -G` from the ServiceCenter directory.
   - From Windows, open an *express* connection to your ServiceCenter server using the syntax:

     scenter -express:*<SC server host>* .*<your express port ID>*.

2. Click the **Toolkit** tab in the Home menu.
3. Click the **Database Manager** button, or type `db` in a command line and press **Enter**.
4. Select **import/load** from the options menu, or press **F8** in text mode.

The File Load/Import Utility is opened.

5 Type the fully qualified path to the SC Upgrade Utilities, followed by the file name.

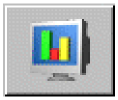
- GUI mode — Type the path in the **File Name** field. For example, if you copied the SC Upgrade files to a `\tmp` directory, the path could be `c:\tmp\upgrade\preupg.bin`.
- Text mode — Type the path in the **External File Name** field.
- OS/390 (MVS) mode — Use your qualifier in lieu of a path (for example, if SC Upgrade is loaded in `USR`, you would enter `USR.PREUPG.BIN`).
- Unix mode — If you loaded the SC Upgrade tape to `/tmp/upgrade`, type `/tmp/upgrade/preupg.bin`.

6 Click the **load fg** button (or press **F1**) to start loading the file.

Depending on the speed of your server, current activity, and the speed of your disk subsystem, this load should take from one to five minutes. When the operation is complete, the system switches back to the main Database Manager form.

## Shutting Down ServiceCenter Schedulers

One of the preliminary steps to running the upgrade process is to shut down the ServiceCenter schedulers.



### To shut down ServiceCenter schedulers:

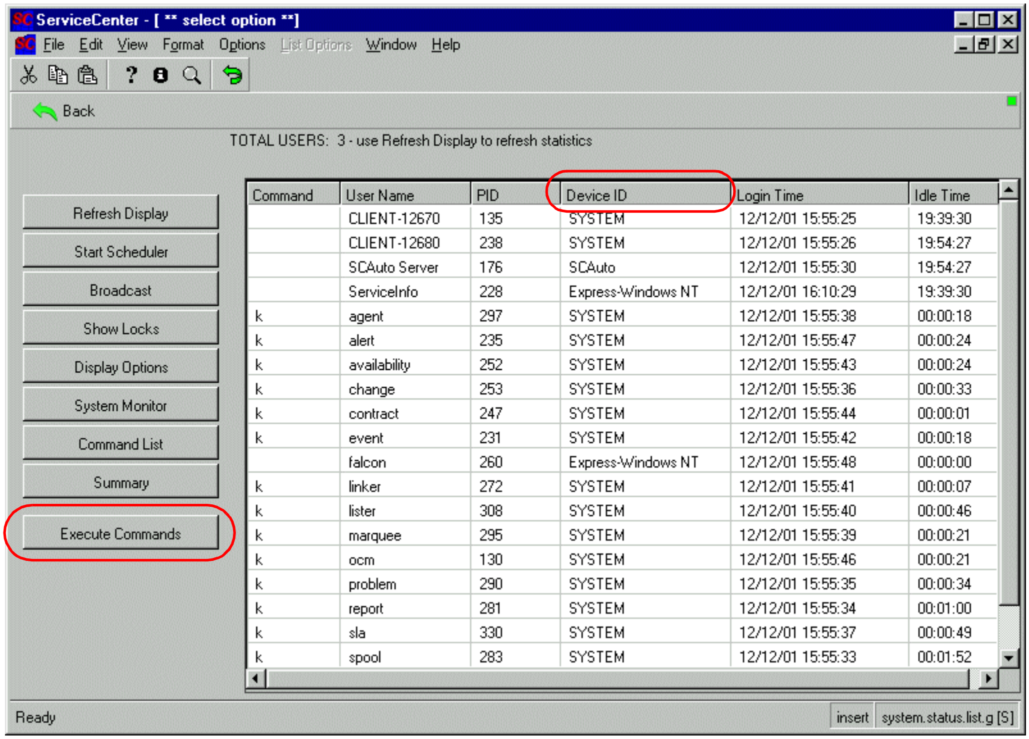
1 Click the **System Status** button in the Home menu.

2 Enter the letter `k` in the **Command** column on every line with a Device ID of *SYSTEM*.

For example, *agent* and *alert* have a Device ID of SYSTEM.

**Note:** You do not need to kill the Express Listener process and client session, identified in the figure as CLIENT-12680 and Falcon, respectively. If you do kill the Express Listener process, you must shut down and restart the server before you can log back on after running the upgrade.

3 Click **Execute Commands**.



TOTAL USERS: 3 - use Refresh Display to refresh statistics

| Command | User Name | PID | Device ID | Login Time | Idle Time |
|---|---|---|---|---|---|
| | CLIENT-12670 | 135 | SYSTEM | 12/12/01 15:55:25 | 19:39:30 |
| | CLIENT-12680 | 238 | SYSTEM | 12/12/01 15:55:26 | 19:54:27 |
| | SCAuto Server | 176 | SCAuto | 12/12/01 15:55:30 | 19:54:27 |
| | ServiceInfo | 228 | Express-Windows NT | 12/12/01 16:10:29 | 19:39:30 |
| k | agent | 297 | SYSTEM | 12/12/01 15:55:38 | 00:00:18 |
| k | alert | 235 | SYSTEM | 12/12/01 15:55:47 | 00:00:24 |
| k | availability | 252 | SYSTEM | 12/12/01 15:55:43 | 00:00:24 |
| k | change | 253 | SYSTEM | 12/12/01 15:55:36 | 00:00:33 |
| k | contract | 247 | SYSTEM | 12/12/01 15:55:44 | 00:00:01 |
| k | event | 231 | SYSTEM | 12/12/01 15:55:42 | 00:00:18 |
| | falcon | 260 | Express-Windows NT | 12/12/01 15:55:48 | 00:00:00 |
| k | linker | 272 | SYSTEM | 12/12/01 15:55:41 | 00:00:07 |
| k | lister | 308 | SYSTEM | 12/12/01 15:55:40 | 00:00:46 |
| k | marquee | 295 | SYSTEM | 12/12/01 15:55:39 | 00:00:21 |
| k | ocm | 130 | SYSTEM | 12/12/01 15:55:46 | 00:00:21 |
| k | problem | 290 | SYSTEM | 12/12/01 15:55:35 | 00:00:34 |
| k | report | 281 | SYSTEM | 12/12/01 15:55:34 | 00:01:00 |
| k | sla | 330 | SYSTEM | 12/12/01 15:55:37 | 00:00:49 |
| k | spool | 283 | SYSTEM | 12/12/01 15:55:33 | 00:01:52 |

**Figure A-1: System Status Screen**

## Starting the ServiceCenter Schedulers

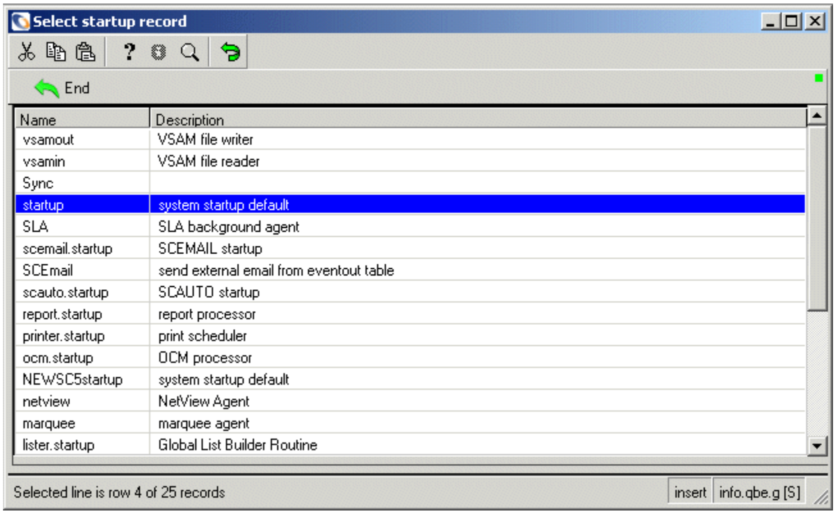
### To return the system to its normal operating environment:

1 Return to the ServiceCenter home menu.

2 Click the **System Status** button.

3 Click the **Start Scheduler** button.

A QBE list of schedulers is displayed.



| Name | Description |
|---|---|
| vsamout | VSAM file writer |
| vsamin | VSAM file reader |
| Sync | |
| startup | system startup default |
| SLA | SLA background agent |
| scemail.startup | SCEMAIL startup |
| SCEmail | send external email from eventout table |
| scauto.startup | SCAUTO startup |
| report.startup | report processor |
| printer.startup | print scheduler |
| ocm.startup | OCM processor |
| NEWSC5startup | system startup default |
| netview | NetView Agent |
| marquee | marquee agent |
| lister.startup | Global List Builder Routine |

Selected line is row 4 of 25 records | insert | info.qbe.g [S]

**Figure A-2: System Schedulers**

4 Select the **startup** scheduler.

5 Press **Enter**.

The startup scheduler activates the schedulers that are normally launched when the ServiceCenter server is started.

After the schedulers are activated, you are returned to the System Status screen.

6 Log out of your current client session.

**Note:** You do not need to shut down and restart the ServiceCenter server unless you stopped the Express Listener when you were stopping the schedulers.

7 Log back on as an administrator using the express client.

**Note:** If you cannot log back on, restart the server.

# Monitoring the Progress of the Application Upgrade

This section provides instructions for monitoring the progress of your upgrade in GUI mode and text mode. Check the log files periodically during the upgrade process to monitor SC Upgrade's progress. See *Tracking the Upgrade Process* on page 150. If you suspect SC Upgrade is hung in OS/390, access the job log to see if there is any change in activity.

**Note:** During the upgrade process, the Windows Task Manager indicates that ServiceCenter is *Not Responding*. This is normal and does NOT indicate a problem with the upgrade.

## Monitoring the progress of the upgrade in GUI mode

### To monitor the progress of the upgrade from a GUI client:

1 Log on to a ServiceCenter express client, if you are not currently logged on.

Select **View** > **Active Notes** to enable Active Notes.

2 Start a ServiceInfo client. This client is a standard ServiceCenter express client with the parameter -si added to the start-up command.

The examples in this step show the procedure if you are *applying* an upgrade, either to a *Development* system or a *Production* system.

Use the same procedure if you are building a custom upgrade, replacing the word *status* with *build* in all of the commands.

- If you are *applying* an upgrade, connect the client to a form called apm.upgrade.*status*.display.
- If you are *building* an upgrade, connect the client to a form called apm.upgrade.*build*.display.

3 Add the command, -si:apm.upgrade.status.display to an express client connection command line. For example, if your original command line read:
\sc\scguiw32.exe -express:myhost.myserv,
your final command line would read as shown for the following platforms:

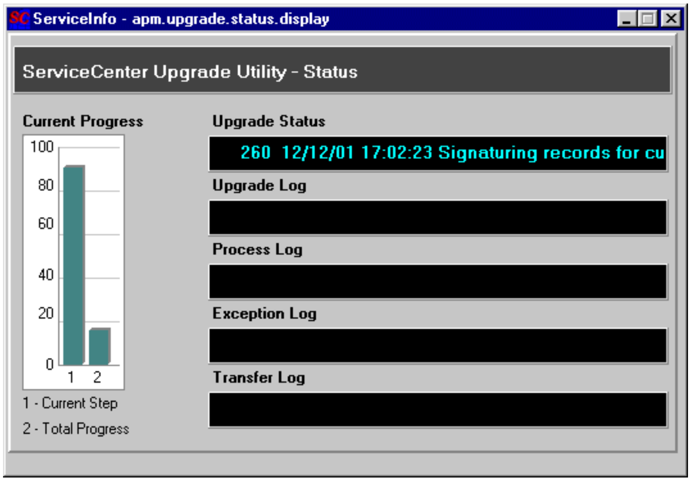
- *Microsoft Windows 32 Bit*
  :\sc\scguiw32.exe -express:myhost.myserv *-si:apm.upgrade.status.display*
- Unix (if your executables are in /user/sc/):

  *Unix Direct Connect*
  /user/sc/scenter -G *-si:apm.upgrade.status.display*

*Unix Client/Server*

/user/sc/scclient -G -express:myhost.myserv *-si:apm.upgrade.status.display*

The ServiceInfo client relates the upgrade's current status and any current messages in the marquee fields. The messages are stored in the upgrade logs. No data is displayed until the upgrade begins.



ServiceInfo - apm.upgrade.status.display

ServiceCenter Upgrade Utility - Status

Current Progress

Upgrade Status

260 12/12/01 17:02:23 Signaturing records for cu

Upgrade Log

Process Log

Exception Log

Transfer Log

1 - Current Step

2 - Total Progress

Figure A-3: ServiceInfo Status Display

## Monitoring the Progress of the Upgrade in Text Mode

To monitor progress in text mode, read SC Upgrade's messages from its internal job log.

### To read the internal job log:

1. Launch a new text connection.
2. Open the Database Manager by typing db in a command line (or press **F7**).
3. Type upgradepseudolog in the **File** field.
4. Press **Enter**.
5. Select **apm.upgrade.log.reader** from the QBE list.
6. Press **Enter**.
7. From the blank message format, press **Enter** to query the message log.

8 To display the details of the process, select a message.
The first log message on the list reflects the most recent job output.

9 Press **Enter**.

## Tracking the Upgrade Process

A set of log files is created during the upgrade process. These files reside in the directory with the upgrade files.

| Log File | Contents |
|---|---|
| upgrade.log | Contains the information about where the upgrade is at any point. This file only contains the main steps of the upgrade. |
| detail.log | Contains specific information about the upgrade, such as which files are being signatured at any time. |
| process.log | Contains specific information about specific records being processed by the upgrade, such as Database Dictionary and data records. |
| except.log | Contains information on any exceptions reported by the upgrade. |

Additionally, you will have the option to store a log internally within ServiceCenter. You can access the logs through the upgrade menu.

# Upgrading the Run-time Environment

If you have not previously upgraded your RTE to SC 5.1, you must do so before starting the upgrade process. Upgrading the RTE is done from the ServiceCenter installation media and does not require any conflict resolution or customization procedures.

For instructions on how to perform and RTE upgrade, refer to the ServiceCenter *Client/Server Installation Guide* for your platform.

**Important:** Shut down your ServiceCenter system before upgrading the RTE. Ensure that the ServiceCenter Console dialog box is closed, or the upgrade will fail.

If the RTE upgrade works properly and passes all tests, copy the production data to this system and use it as your *Production* system.

### To use the RTE upgraded system as your Production system:

1. Bring down the ServiceCenter *Production* system.
2. Make a *full* backup of the ServiceCenter system in your *Production* system. See *Making a Backup of the Production System* on page 58.
3. Copy `scdb`.* and `ir`.* files from the Data folder in the *Development* environment, overwriting the files in the *Production* system.

   You can also create a separate folder for the OUT OF BOX data and move the `scdb`.* files to that folder. If you overwrite them and need to refer to them during the upgrade, they can be copied from the WIN\DATA folder on the SC 5.1 Upgrade CD. Be sure to remove the **Read Only** property.
4. Restart the *Production* system.

**Note:** Having the RTE upgraded and running while you develop the custom application upgrade lessens the time your system will have to be down for the application upgrade. It also allows time for any RTE issues to come up, and ensures that you are only changing one thing at a time.

# B RAD Comparison Utility

APPENDIX

The RAD Comparison Utility is an on-line software management tool that allows for one version of a ServiceCenter RAD application to be compared to a different version of the same application.

The RAD Comparison Utility allows you to quickly and accurately determine what changes have been made to a RAD application. Those users who have made customized changes to ServiceCenter RAD applications will find this utility useful during the ServiceCenter upgrade process.

Topics in this appendix include:

# The RAD Comparison Utility

The RAD Comparison Utility can be accessed from two locations:

- From the command line
- Through the RAD Editor and the PeregrineFour Application Encyclopedia

The utility can be run either in GUI or text mode. GUI mode is used as an example in this chapter.

## Accessing the RAD Comparison Utility from a Command Line

### To access the RAD Comparison Utility from a command line:

▶ Enter `agcompare` on the command line.

The RAD Comparison Utility is displayed.

## Accessing the RAD Comparison Utility through the RAD Editor

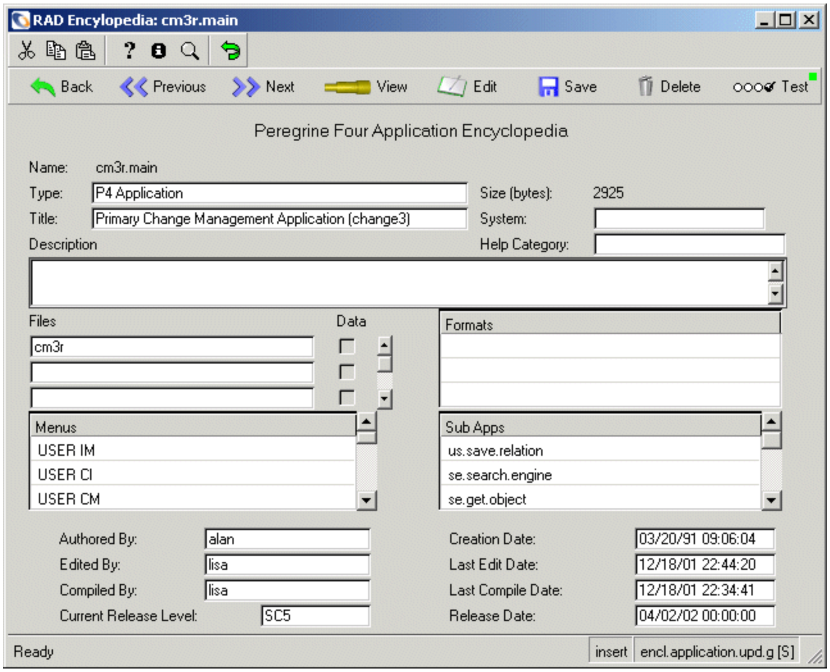
### To access the RAD Comparison Utility through the RAD Editor:

1. Click the **Toolkit** tab in the Home menu.
2. Click the **RAD Editor** button.

   The RAD Editor is displayed.
3. Type the name of the application you want to compare in the RAD Editor form.

   For example, you could compare the `cm3r.main` application from Change Management.
4. Click the **Search** button or press **Enter**.

The Application Encyclopedia form for the application is displayed.



Peregrine Four Application Encyclopedia

Name: cm3r.main

Type: P4 Application

Size (bytes): 2925

Title: Primary Change Management Application (change3)

System:

Description

Help Category:

Files: cm3r

Data: □ □ □

Formats

| Menus | Sub Apps |
|---|---|
| USER IM | us.save.relation |
| USER CI | se.search.engine |
| USER CM | se.get.object |

Authored By: alan

Edited By: lisa

Compiled By: lisa

Current Release Level: SC5

Creation Date: 03/20/91 09:06:04

Last Edit Date: 12/18/01 22:44:20

Last Compile Date: 12/18/01 22:34:41

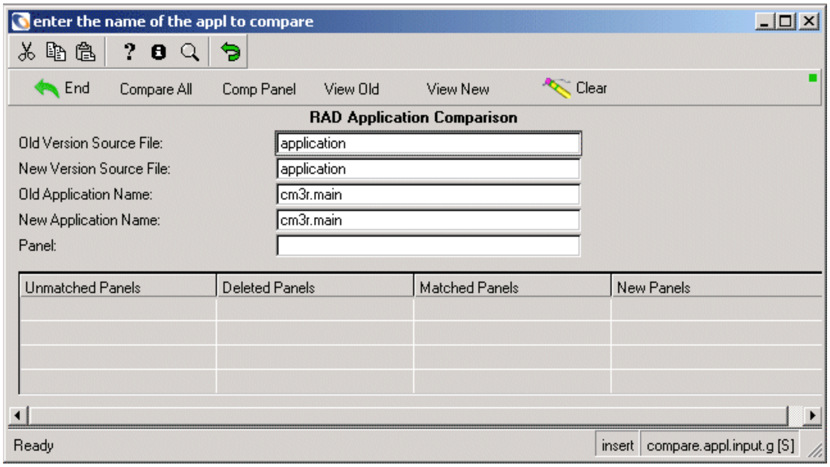
Release Date: 04/02/02 00:00:00

Ready | insert | encl.application.upd.g [S]

**Figure B-1: Application Encyclopedia Containing *cm3r.main***

5 Select **Options** > **Compare Applications** from the menu bar.

A form is displayed from which you can perform the comparisons.



RAD Application Comparison

Old Version Source File: application

New Version Source File: application

Old Application Name: cm3r.main

New Application Name: cm3r.main

Panel:

Figure B-2: RAD Application Comparison Form

When you select the Comparison Utility from the Options menu, the **Old Application Name** and **New Application Name** fields default to the name of the application as it appeared in the Application Encyclopedia Record.

The following sections describe the functions of the system tray buttons, define the data fields, and give instructions for using the comparison options.

Table B-1: System Tray Buttons

| 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. |
| 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 impact the old or new RAD applications. |

Table B-2: Data Fields

| Field | Definition |
|---|---|
| Old Version Source File | Defines the logical file containing the application records (RAD panels) of the application specified in the **Old Application Name** field. This field defaults to *application*. |
| New Version Source File | Defines the logical file containing the application records (RAD panels) of the application specified in the **New Application Name** field. This field defaults to *application*. |
| Old Application Name | Defines the name of the application that resides in the **Old Version Source File** field. The Old Version Source file is used as a base upon which the application specified in the **New Application Name** field is modeled.<br>If the Comparison Utility is called from an Encyclopedia Record, the **Old Application Name** field defaults to the application name displayed in the Encyclopedia Record. |
| New Application Name | Defines the name of the application residing in the **New Version Source File** field that is to be compared against the application specified in the **Old Application Name** field.<br>If the Comparison Utility is called from an Encyclopedia Record, the **New Application Name** field defaults to the application name displayed in the Encyclopedia Record. |
| Panel | Names a panel to be used for comparison checking. The **Panel** field allows for viewing of a specified panel in either the old or new versions of the application. |
| Unmatched panels * | Contains a list (array) of panel names in which differences were detected between the old and new versions of the RAD application. |
| Deleted panels * | Contains a list (array) of panel names present in the old version of the application that are not present in the new version. |
| Matched panels * | Contains a list (array) of panel names in which no differences were detected between the old and new versions of the RAD application. In text mode, the form must be scrolled to the right (**F15**) to view this field. |
| New panels * | Contains a list (array) of all panel names present in the new version of the application that are not present in the old version. |

* The list is not completed until the **Compare All** button is clicked. The field is set to allow for modifications and easy tabbing.

**Important:** Manual changes should not be made to the contents of the fields marked above with an asterisk (*).

Detail Level results are available for all panels where differences have been detected between the old and new versions. The comparison results may be viewed online and/or printed on the user's default printer. The old and new versions of each panel may also be viewed online.

## Defining or Modifying Source File Definitions

The Old Version Source File and New Version Source File fields define the name of the logical file, which contains the RAD panels for the applications to be compared. Although both fields default to `application`, you can override the defaults.

**Note:** By default, ServiceCenter contains one Application Library. If you find it necessary to maintain two or more Application Libraries, you are responsible for allocating and controlling the functionality of these files and for defining those routines necessary for the exchange of data between files.

### To define or modify source file definitions:

1. Access the **RAD Application Comparison** form.
2. Tab to the **Old Version Source File** field.
3. Enter the name of the logical file that contains the panels of the old application.

Follow the same procedure for specifying a different **New Version Source File**, if necessary.

# Defining Application Names

### To define an application name:

1. In the RAD Application Comparison form, enter the name of the old version of the application in the **Old Application Name** field.
2. Enter the name of the new version of the application in the **New Application Name** field.

Both of these definitions must be in place before attempting to compare either the entire application or a single application panel.

## Comparing Entire Applications

### To compare entire RAD applications:

1. Access the RAD Application Comparison form.
2. Enter the proper data in the **Old Version Source File**, **New Version Source File**, **Old Application Name** and **New Application Name** fields.
3. Click the **Compare All** button.

ServiceCenter 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 using the command buttons in the Service Tray.

Following the execution of a **Compare All** command, another button appears in the Service Tray.

Prints all the panel records currently displayed.

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.

Refer to the following sections for more information on these options.

**Note:** If you decide to compare a different application, you MUST click the **Clear** button before proceeding. This step ensures that all controls are reset before processing begins.

## Printing a Report

To get a printed report of the Comparison Results, click the **Print All** button. The print job is routed to the user's default printer.

**Table B-3: Comparison Results Report Contents**

| Reported Item | Definition |
|---|---|
| **Matched Panels** | a summary page listing the names of all panels that matched. |
| **Unmatched Panels** | a summary page listing the names 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** | a summary page listing the names of all deleted panels and a printout of each deleted panel. |
| **New Panels** | a summary page listing the names of all new panels and a printout of each panel. |

# Comparing Single Panels

Compare single panels either before or after the entire application has been compared. Be certain you have entered valid data on the new and old versions of the application in all the source file and application name fields.

### To view a detailed comparison of a single panel after the entire application has been compared:

1. Select the desired panel in the **Unmatched Panels** array.
2. Click the **comp panel** button.

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

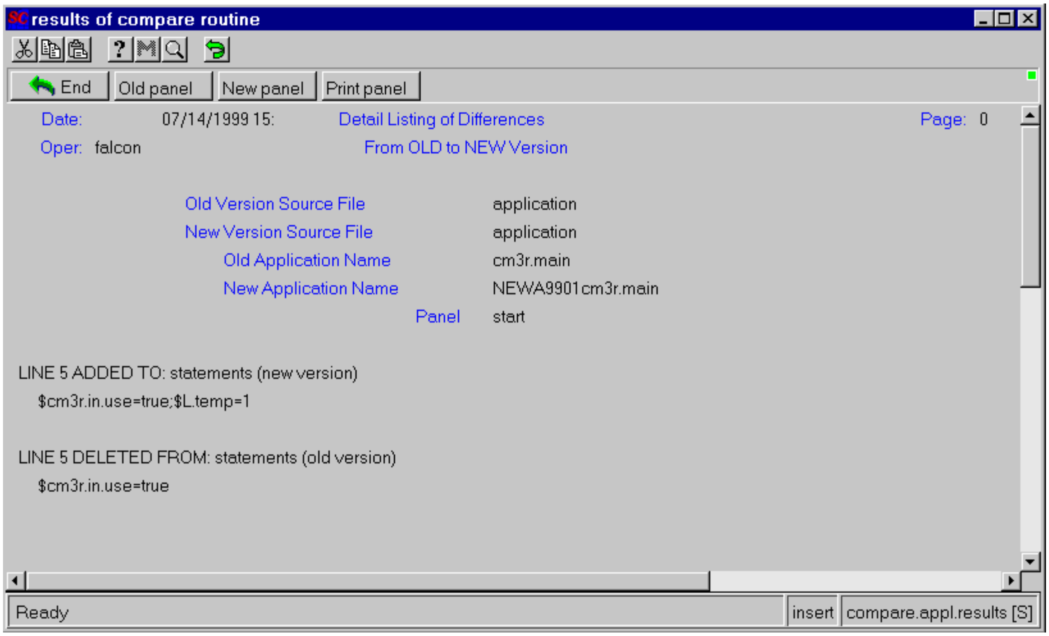
1. Enter the name of the panel in the **Panel** field.
2. Click the **comp panel** button.

**Note:** You must select the **Panel** field in order for ServiceCenter to locate the specified panel. A red border appears around a selected field.

The **Detail Listing of Differences** form displays the exact differences between the old and new versions of the panel. The results are presented in the same manner for both online viewing and in printed form. Every page shows the name of the Old and New Version Source File names and the names of the Old and New Application Name fields.

**Table B-4: Buttons on the Detail Listing of Differences Form**

| Button | Definition |
|---|---|
| old panel | Displays the old version of the application panel being compared. |
| new panel | Displays the new version of the application panel being compared. |
| print panel | Prints the *Detail Listing of Differences* form. |



```
Date:           07/14/1999 15:        Detail Listing of Differences                Page:  0
Oper:   falcon                          From OLD to NEW Version

                    Old Version Source File            application
                    New Version Source File            application
                       Old Application Name            cm3r.main
                       New Application Name            NEWA9901cm3r.main
                                      Panel            start

LINE 5 ADDED TO: statements (new version)
   $cm3r.in.use=true;$L.temp=1

LINE 5 DELETED FROM: statements (old version)
   $cm3r.in.use=true
```

**Figure B-3: Detail Listing of Differences**

# Array and Scalar Field Differences

When differences are noted between the old and new versions of an *array* or *scalar* field, the information is presented in the following sequence:

```
Line XX:Action Prompt
```

**Table B-5: Differences Definitions**

| Item | Definition |
|---|---|
| **Line** | constant. |
| XX | the element number of the array that has been deleted or added. For example, *LINE* 5 in Figure B-3 on page 161. |
| **Action** | changes to arrayed or scalar field elements are always accompanied by the phrases *ADDED TO* and *DELETED FROM*. If a change was made 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*. For example, *$cm3r.in.use=true,$L.temp*1* in Figure B-3 on page 161. |
| **Prompt** | the name of the prompt on the RAD panel that corresponds to the array or scalar field, which has been changed. |

## Viewing Versions of a Panel

You can view old and new versions of a panel from the following forms:

### From the RAD Compare Application form:

- After the entire application has been compared, type the name of the panel in the **Panel** field or select the panel from one of the lists. Click the **View Old** button, or click the **View New** button.

### From the Detail Listing of Differences form

- Click the **Old Panel** button or the **New Panel** button to open a version of the current panel.

## Printing a Detail Listing of Differences

### To print the Detail Listing you have displayed:

Click the **Print panel** button.

If **Active Notes** is enabled, a dialog box opens indicating the report has been spooled and scheduled for printing on the user’s default printer.

## Continuation Lines

If the contents of a scalar field or an element of an arrayed field have a length of greater than 72 bytes, the comparison results for those lines are displayed in their entirety with the use of *continuation lines*. All continuation lines start with three asterisks (***).

Continuation lines are employed for both online and printed comparison results.

The last two characters of a Detail Line appear as the first two characters of the next continuation line.

APPENDIX C

# Adding ITIL Functionality

ServiceCenter 4.0 incorporated workflows and best practices, based on the Information Technology Infrastructure Library (ITIL). For complete information on the best practices and workflows of ITIL, see the ServiceCenter *Introduction and Best Practices* guide.

If you are upgrading from a release prior to ServiceCenter 4.0, this functionality is not installed automatically with the upgrade, but can be added through a series of unload files, as listed in this appendix.

The module known in previous ServiceCenter releases as Problem Management was renamed Incident Management to more clearly draw its correspondence to ITIL workflows.

To minimize the impact when upgrading, the names of the underlying database dictionary structure, probsummary and problem files, remain unchanged. You can still customize the forms to be called Problem Management, instead of Incident Management.

The out-of-the-box naming convention for incident tickets is in the format *IMnnn* where *IM* is the prefix and *nnn* is a sequential number.

# Unload Files

The following files contain category and group information for Incident and Change Management. You may load these files onto your system to take advantage of ITIL-based functionality.

- *IMCAT.unl* on page 166.
- *IMGRP.unl* on page 167.
- *CMCAT.unl* on page 167.
- *CMGRP.unl:* on page 168.

Use the import/load procedures described in *Loading a File into ServiceCenter* on page 144.

**Warning:** *Loading these files causes any category or group record with corresponding names to be overwritten.* Rename the existing files to prevent this occurrence.

# Unload Contents

The contents of the unload files are listed in the following tables.

## IMCAT.unl

### Incident Management category records in IMCAT.unl

| | |
|---|---|
| DEFAULT | business applications |
| change | client system |
| enquiry | example |
| getservice | network |
| other | printing |
| security | shared infrastructure |
| tbd | telecoms |

## IMGRP.unl

### Incident Management assignment groups (assignment) in IMGRP.unl

| | |
|---|---|
| AUTO | CLIENT SECURITY |
| CRSP | DEFAULT |
| DODSPG2DUSD1 | DUTYMANAGER |
| ENF01OPS | ENF01TS1 |
| ENF01TSG | FACILITIES |
| FEL01U1 | FIELD ENG. |
| FIRSTLINE | GLO06DB |
| GLO6TS | HELPDESK |
| LAN SUPPORT | M/F SUPPORT |
| MAN23TSI | ONSITE SUPPORT |
| PEREGRINE | PROCUREMENT |
| REPLACEMENT | SECONDLINE |
| SERVICE MANAGEMENT | SMSDESKSXXME |
| SOFTWARE | STE04U1 |
| SYSTEMS ADMIN | SYSTEMS SUPPORT |
| TELECOMS | TRAINING |
| WAN SUPPORT | WOR01TS |

## CMCAT.unl

### Change records (cm3rcategory) in CMCAT.unl

| | |
|---|---|
| RFC | RFC - ADVANCED |

### Change phase records (cm3rcatphase) in CMCAT.unl

| | |
|---|---|
| Assessment | Building |
| RFC Testing | RFC Implementation |
| 1.assess | 2.plan |
| 3.build | 4.implement |
| 5.accept | |

### Task category records (cm3tcategory) in CMCAT.unl

| | |
|---|---|
| Hardware | Software |
| client.mgt | implement.task |
| plan.1/2.task | procurement |
| resource.mgt | third.party.mgt |

### Task phase records (cm3tcatphase) in CMCAT.unl

| | |
|---|---|
| Hardware | Software |
| client.mgt | implement.task |
| plan.1/2.task | procurement |
| resource.mgt | third.party.mgt |

## CMGRP.unl:

### Change Management message group, profile, and profile group records (cm3groups, cm3profile, cm3profilegrp) in CMGRP.unl

| | |
|---|---|
| ASSET MANAGEMENT | CA |
| CI | CM |
| CO | CS |
| EMERGENCY GROUP | FACILITIES |
| HELPDESK | LAN SUPPORT |
| M/F SUPPORT | ONSITE SUPPORT |
| PROCUREMENT | SERVICE MANAGEMENT |
| SOFTWARE | SYSTEMS ADMIN |
| SYSTEMS SUPPORT | TELECOMS |
| TRAINING | WAN SUPPORT |

# D APPENDIX Using the SQL Compare Utility

ServiceCenter contains a set of applications to assist the administrator of an SQL-mapped system in determining the database changes necessary to support an upgrade without converting their files back to P4.

These applications determine which databases that have been mapped to SQL must be modified prior to the upgrade procedure. By correctly applying the changes specified by these applications, the ServiceCenter system may be upgraded while mapped to SQL. This comparison is done on the *Development* system. The speed of the SQL Compare utility was enhanced in SC51 to make it run faster.

Topics in this appendix include:

- *Upgrading SQL Databases* on page 170
- *Loading the Compare Applications* on page 170
- *Running the Compare Applications* on page 171
- *Analyzing the Results* on page 172
- *Adding New Fields* on page 173

# Upgrading SQL Databases

**To prepare the system for the application upgrade:**

**Step 1** Install the Compare Application.
See *Loading the Compare Applications* on page 170

**Step 2** Run the Compare Application.
See *Running the Compare Applications* on page 171.

**Step 3** Determine any new fields that must be added to SQL mapped files. The Compare utility determines these fields for you.
See *Analyzing the Results* on page 172.

**Step 4** Add the new fields to the SQL database.

**Step 5** Add the new fields to the P4 database dictionary.

**Step 6** Update the SQL mapping within ServiceCenter for each affected file.

For instructions on adding new fields, see *Adding New Fields* on page 173.

The first step is accomplished by running the SQL Compare application suite. These applications compare any database dictionary currently mapped to SQL with the newest version needed for the upgrade. The applications also report any fields that need to be added prior to applying the upgrade.

# Loading the Compare Applications

The SQL Compare applications are located on the SC Upgrade media.

Two separate files are used:

- sqlupgrade.unl
- upgdbdct.dta

▶ Load the sqlupgrade.unl file into your ServiceCenter system.

For instructions, see *Loading a File into ServiceCenter* on page 144.

# Running the Compare Applications

Once the applications are started, they begin to analyze your system. When the routine is finished, the system returns a message stating:

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

An additional message is returned for every P4 database dictionary mapped to SQL that contains new fields. These database dictionaries must be updated to contain the fields specified by the SQL Compare applications before the application upgrade can be applied.

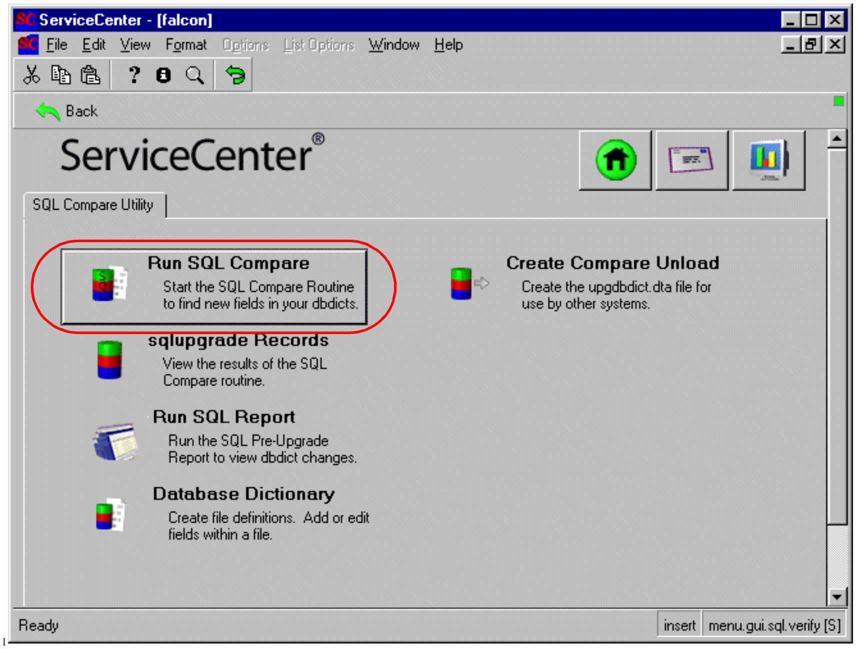
## GUI Mode

### To run the compare applications in GUI mode:

1 From a command line, type: *`mSQL COMPARE`

The **SQL Compare Utility** menu is displayed.

2 Click the **Run SQL Compare Routine** button.



**Run SQL Compare**
Start the SQL Compare Routine to find new fields in your dbdicts.

**Create Compare Unload**
Create the upgdbdict.dta file for use by other systems.

**sqlupgrade Records**
View the results of the SQL Compare routine.

**Run SQL Report**
Run the SQL Pre-Upgrade Report to view dbdict changes.

**Database Dictionary**
Create file definitions. Add or edit fields within a file.

**Figure D-1: . SQL Compare Utility Menu**

## Text Mode

### To run the compare applications in text mode:

1. From a command line, type: *aapm.upgrade.verify.sql
2. The system prompts you for the path to the upgdbdct.dta file. Enter the full path (for example, /usr/upgrade/) or the high level qualifier.
3. Click **OK**.

# Analyzing the Results

The results of the SQL Compare process are stored in the sqlupgrade file.

**Note:** This file is reset every time the SQL Compare process is run.

### To view the sqlupgrade file in GUI mode:

1. Access the **SQL Compare Utility** menu.
2. Click the **sqlupgrade Records** button.

### To view the sqlupgrade file in text mode:

Text mode users can access the sqlupgrade records via the Database Manager utility.

A Report Writer report has been created to print all sqlupgrade records. The report is titled sql.pre.upgrade.rpt, and may be accessed from the SQL Compare menu by clicking the **Run SQL Pre-Upgrade Report** button.

## Reviewing the sqlupgrade Records

Each file that requires changes is stored as a separate record in the sqlupgrade database. This record also details the new fields that must be added to the database.

In the sqlupgrade record, the following information is given for each field that needs to be added:

| Field | Description |
|---|---|
| **Field Name** | The exact field name to add to the P4 database dictionary specified in the file name. |
| **Type** | The data type of the field. |
| **Level** | The level at which this field resides. |
| **Structure/Array** | The structure and/or array name to which this field should be added. |

For more information on adding fields, see *Adding New Fields* on page 173.

# Adding New Fields

In order for the new fields to perform correctly, they must be added to both the P4 database dictionary and the SQL database. In addition, the existing SQL Mapping inside of ServiceCenter must be updated manually.

When updating a table that is defined as a system table (in the sqlsystemtables file), fields should only be added through the ServiceCenter Database Dictionary utility. Modifying the SQL mapping damages the file structure of the table.

For more information on adding fields to an RDBMS-mapped system, refer to the Database Conversion section of the *Database Management and Administration* guide.

After you have finished adding the new fields, proceed to *Upgrading Your System* on page 69.

# Determining the Correct Structure/Array

In most instances, the new field is added to the descriptor structure.

In three instances, the Structure/Array field contains something other than the word **descriptor**.

- The field resides in another structure.

- The field is an array.
- The field is part of an array of structures.

If the Structure/Array field does not read **descriptor**, and the field is not an array field (see the next paragraph), then the field must be added to the structure listed in the Structure/Array field. For example, if the Structure/Array field reads **middle**, the field should be added to the middle structure of the dbdict.

If the field is an array, the field name is listed twice in the new field list. The first entry is of type **array**, and the second is the data type of the array (**character**, **logical**, etc.). The first entry is used to determine the structure to which the array should be added, using the rules outlined above. The Structure/Array 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.

If the Structure/Array field lists multiple fields exclusive of an array name, the field must be added 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.

**Important:** When adding fields to a structured array, it is imperative that the fields are added in the same order as they are listed in the `sqlupgrade` record.

## Creating Subtables from an Array of Structures

ServiceCenter enables a dbdict administrator to manage data more effectively by creating subtables of unique and non-unique attributes within an array of structures. You can use this feature to:

- Improve mapping to external SQL database tables.
- Implement a more cost-effective solution for managing attribute information.
- Simplify queries.

The dbdict administrator can identify two subtable names for each array of structures in the dbdict. One table contains the names of unique attributes; the second table names non-unique attributes. A pop-up utility dialog box enables you to identify which attributes are unique.

The subtable feature helps you create queries that can return detailed information. This type of available detail can improve business and management decisions. You can create subtables for an array of structures in any dbdict. ServiceCenter ships with subtables already created for all arrays of structures in the inventory dbdicts.

# E Troubleshooting

APPENDIX

This appendix includes the following sections:

- *Missing Function Keys* on page 178 — describes how to fix missing database option buttons in the system tray.
- *Contacting Customer Support* on page 180 — lists telephone numbers, email addresses and mailing addresses for Peregrine Systems Customer Support around the world.

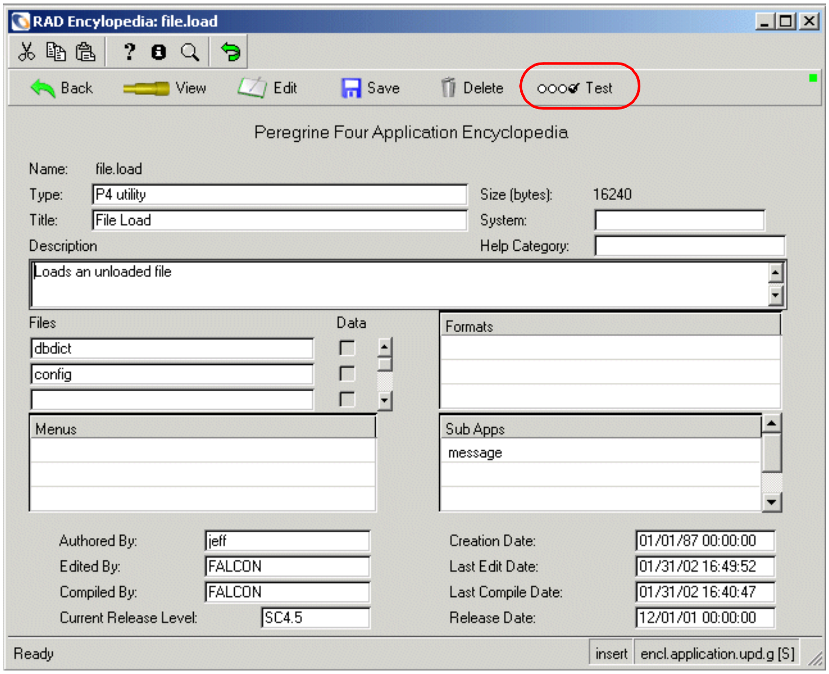
# Missing Function Keys

If you have completed the upgrade and the database option buttons in the system tray have disappeared, the procedure for upgrading the displayoption and displayevent files has failed, and your system cannot access the options defined in these files. Since the database options are unavailable, you cannot use the database Import/Load Utility. You must load the missing datafiles from RAD.

### To reload the display files:

1. Click the **Toolkit** tab in the **Home** menu.
2. Click the **RAD Editor** button.
3. Type file.load in the **Application** field of the RAD Editor prompt.
4. Click **Search.**
5. Select **file.load** from the QBE list.

   The Application Encyclopedia for **file.load** is displayed.



Peregrine Four Application Encyclopedia

| | | | |
|---|---|---|---|
| Name: | file.load | | |
| Type: | P4 utility | Size (bytes): | 16240 |
| Title: | File Load | System: | |
| Description | | Help Category: | |

Loads an unloaded file

| Files | Data |
|---|---|
| dbdict | □ |
| config | □ |
| | □ |

Formats

Menus

Sub Apps: message

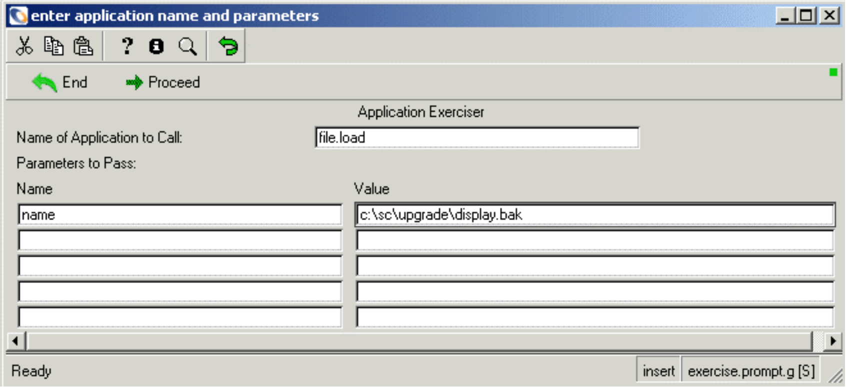
| | | | |
|---|---|---|---|
| Authored By: | jeff | Creation Date: | 01/01/87 00:00:00 |
| Edited By: | FALCON | Last Edit Date: | 01/31/02 16:49:52 |
| Compiled By: | FALCON | Last Compile Date: | 01/31/02 16:40:47 |
| Current Release Level: | SC4.5 | Release Date: | 12/01/01 00:00:00 |

Ready | insert | encl.application.upd.g [S]

**Figure E-1: Application Encyclopedia**

6 Click **Test**.

The Application Exerciser is displayed.



Name of Application to Call: file.load

Parameters to Pass:

| Name | Value |
|---|---|
| name | c:\sc\upgrade\display.bak |

**Figure E-2: RAD Application Exerciser**

7 Enter the following parameter values:

| Name | Value |
|---|---|
| name | *<path and filename>* |

where the *path and filename* point to one of two files:

- The display.bak file created by the upgrade.

**Important:** If you choose this path, you MUST run the application apm.upgrade.disp by typing *aapm.upgrade.disp on a command line.

- The path to the upgdisp2.dta file from the SC Upgrade Utility CD.

**Note:** If you choose this path, you will lose any customized display option features.

8 Click **Proceed** to load the datafile and restore your displayoption and displayevent files.

9 You must log out of your ServiceCenter client and then log back on again before your options are displayed.

# Contacting Customer Support

For more information and assistance with this new release or with ServiceCenter in general, contact Peregrine Systems' Customer Support. Current details of local support offices are available through these main contacts.

## Peregrine's CenterPoint Web site

Current details of local support offices are available through the following main contacts or through Peregrine's CenterPoint Web site at:

http://support.peregrine.com/

After logging in with your login and password:

- Select **Go** for **CenterPoint**.
- Then select **Whom Do I Call?** from **Contents** on the left side of the page to display the **Peregrine Worldwide Contact Information.**

## Corporate Headquarters

| | |
|---|---|
| Address: | Peregrine Systems, Inc.<br>Attn: Customer Support<br>3611 Valley Centre Drive<br>San Diego, CA 92130 |
| Telephone: | +(1) (858) 794-7428 |
| Fax: | +(1) (858) 480-3928 |

## North America and South America

| | |
|---|---|
| Telephone: | (1) (800) 960-9998 (US and Canada only, toll free)<br>+(1) (858) 794-7428 (Mexico, Central, and South America) |
| Fax: | +(1) (858) 480-3928 |
| E-mail: | support@peregrine.com |

## Europe, Asia/Pacific, Africa

For details of local offices, see Peregrine's CenterPoint Web site, as explained under *Peregrine's CenterPoint Web site* on page 180.

You can also contact the Corporate Headquarters, using the information provided above.

# Searching for Details of SCRs

You can search for details of corrections and enhancements that were implemented within the release via Peregrine's CenterPoint Web site: http://support.peregrine.com/

**After logging in with your login and password,**

1 Select **Go** for **CenterPoint**.

2 Select **ServiceCenter** from **My Products** at the top of the page.

3 Then, from **Contents** on the left, select E**nhancements and Corrections** (**SCR**).

4 Follow the search **Instructions** for entering criterion (for example, SCR number, Incident ticket number, or a keyword), then click **Search**. A report of SCRs matching the search criteria is displayed.

# Index

Peregrine
SYSTEMS®

May 28, 2003