Caché Installation Guide

Version 5.1
15 June 2006

InterSystems Corporation 1 Memorial Drive Cambridge MA 02142 www.intersystems.com

Caché Installation Guide
Caché Version 5.1 15 June 2006

This book was assembled and formatted in Adobe Page Description Format (PDF) using tools and information from the following sources: Sun Microsystems, RenderX, Inc., Adobe Systems, and the World Wide Web Consortium at www.w3c.org. The primary document development tools were special-purpose XML-processing applications built by InterSystems using Caché and Java.

For Support questions about any InterSystems products, contact:

InterSystems Worldwide Customer Support

Tel:	+1 617 621-0700
Fax:	+1 617 374-9391
Email:	support@InterSystems.com

Table of Contents

List of Tables

Introduction

Caché runs on several different platforms. Please check the Caché Supported Platforms document to verify if Caché runs on your particular version of the supported operating systems.

The *Caché Installation Guide* contains the following chapters and related appendixes.

If you are upgrading from Caché version 4.1 or later, read the following chapter for a list of pre-installation upgrade tasks:

- Upgrading Caché

Install Caché following the instructions in the appropriate platform-specific installation chapter:

- Installing Caché on Microsoft Windows
- Installing Caché on OpenVMS
- Installing Caché on UNIX and Linux
- Installing Caché on Macintosh

Before installing, review the material in the appropriate platform-specific appendix for setting operating system parameters:

- Calculating Caché System Parameters for OpenVMS
- Calculating Caché System Parameters for UNIX and Linux

You may need to read the following appendixes before you install Caché if they apply to your environment:

- Preparing for Caché Advanced Security — If you are using the Kerberos authentication method, read this appendix before installing Caché.
- Upgrading Caché from Versions Prior to 4.1 — If you are upgrading from a Caché version prior to Caché 4.1, read this appendix before installing Caché.

1

Upgrading Caché

This chapter is intended for customers who are upgrading from Caché release 4.1 and above. Topics in this document include:

- Supported Upgrade Paths
- Upgrade Tasks

Important: Before upgrading, review the Caché 5.1 Conversion Guide for issues that may apply to your site.

1.1 Supported Upgrade Paths

Direct Upgrades

The following *direct upgrade* paths to Caché 5.1 are supported:

- Caché 5.1.x
- Caché 5.0.x
- Caché 4.1.x

You may upgrade directly to Caché 5.1 from version 4.1 or above without converting existing databases.

For upgrading from earlier versions, see the "Upgrading Caché from Versions Prior to 4.1" appendix.

1.2 Upgrade Tasks

The following upgrade tasks are necessary regardless of the platform on which you are upgrading and running Caché. Perform these tasks before you run the Caché installation procedures:

1. *Obtain an updated license key* — the key structure is new in Caché 5.1; upgrades to Caché 5.1 from Caché 5.0 or earlier require an updated key.
2. *Backup system* — before upgrading, InterSystems recommends that you run a complete backup of your system. Use your customary full operating system backup procedures.
3. *Check system integrity* — run a system integrity check on existing directories to ensure there is no corruption in any of the databases.
4. *Save custom routines and globals* — to prevent your own routines and globals in the %SYS namespace from being affected by the upgrade installation, ensure that they have names that begin with "Z" , "z" , "%Z" , or "%z" . All .int and .obj routines (except for Z*, z*, %Z*, and %z*) are deleted from the %SYS namespace when upgrading.

   On an upgrade, the CACHELIB, CACHETEMP, DOCBOOK, and SAMPLES databases are completely replaced.

   Any .mac or .inc routines are not affected during the upgrade.
5. *Save user files* — additional files and directories are also deleted or replaced during an upgrade. It is safest to have all your user files in the \Devuser directory or any subdirectories named \User under the Caché installation directory. For a list of which files are deleted or replaced, see the Files Deleted or Replaced on Upgrade section of the "Upgrading Caché from Versions Prior to 4.1" appendix.
6. *Remove PIJ file (OpenVMS clusters only)* — If you are upgrading a member of a Caché OpenVMS cluster system to Caché 5.1, cleanly shut down all members of the Caché cluster and remove the CACHE.PIJ file. If you do not remove this file, the installation is not upgraded and error messages are written in the cconsole.log for startup:

```
Cache (2100036c) Tue Aug 1 14:28:59 2005
Activating Namespaces
Cache (21000404) Tue Aug 1 14:28:59 2005 Cluster image journal
is incompatible with this version
Cache (21000404) Tue Aug 1 14:28:59 2005 Unable to join the cluster
Cache (21000404) Tue Aug 1 14:29:00 2005
ENQdaemon exited due to VMS error code (decimal) 0
```

Important: *Upgrade CSP Gateway* — If your CSP Gateway is on a separate machine from the Caché server you are upgrading, you must also upgrade the CSP Gateway on that separate machine. You accomplish this by performing a custom Caché install on your Web server machine and choosing only to install the CSP Gateway. See the Using Caché Server Pages with a Remote Web Server section of the "Connecting to Remote Servers" chapter of the *Caché System Administration Guide* for details.

2

Installing Caché on Microsoft Windows

This chapter describes how to install Caché 5.1 on a Microsoft Windows system. It assumes you are familiar with Windows directory structures, utilities, and commands. This chapter contains the following major sections:

- Installation Requirements
- Caché Installation
- Post-Installation Tasks
- Special Considerations

2.1 Installation Requirements

This section describes the hardware and software requirements for new and upgrade installations of Caché 5.1.

2.1.1 Disk Space Requirements

A standard Caché installation that includes support for Caché Server Pages (CSP) uses approximately 450 MB (megabytes) of disk storage (not including disk space for user data). You must have 10 MB additional disk space free on your Windows system disk for installation.

Your system must have access to a CD-ROM drive for installation, either attached to your computer or available across a network.

Any system that can effectively support Windows should be sufficiently powerful to run Caché. Caché performance greatly improves with increased processor and disk speed.

2.1.2 Supported Platforms and Web Servers

The latest version of Caché is supported on the following versions of the Microsoft Windows operating system:

- Windows Server 2003 (with SP1) — 32- and 64-bit versions
- Windows XP Pro (with SP2)
- Windows 2000 (with SP4)

The CSP technology is supported on the Microsoft IIS Web server and on the Apache Web server, versions 1.3 and 2.0.

If you are using CSP, you must install the Web server before installing Caché. Its support on each operating system is dependent on the operating system vendor and is subject to change. See the Web Server Configuration section of the "CSP Configuration" chapter of the *Using Caché Server Pages* guide for more information.

With each instance, Caché installs a private Apache Web server and a private CSP Gateway to serve CSP pages to assure proper operation of the System Management Portal and the Caché Online Documentation. Its Windows service name is "Web Server for <instance>" where *<instance>* is the instance name you enter when you install Caché. Caché installs the Web server into the <CacheSys>\httpd directory, where *<CacheSys>* is the Caché installation directory. It is uninstalled when you uninstall the corresponding Caché instance.

If you are installing Caché from a network, first copy the entire contents of the CD to the network drive. Map the network drive to a particular drive letter, for example, "Q:" , by clicking **Map Network Drive** from the **Tools** menu in Windows Explorer.

2.1.3 Supported Upgrade Paths and Procedures

If you are performing an upgrade, please first read and perform all necessary procedures described in the "Upgrading Caché" chapter of this guide.

When upgrading, back up your old Caché installation after completing all the pre-installation upgrade tasks and before installing Caché.

2.1.4 Installation Directory Restrictions

You cannot install Caché into a destination directory that has any of the following characteristics:

- It has a caret (^) in the pathname.
- It has a character that is not in the US ASCII character set.
- It is at the root level of a drive (such as C:\).
- It has more than 37 characters in the pathname.

2.2 Caché Installation

The steps for installing each type of a Caché 5.1 configuration are fundamentally the same, but diverge slightly in the steps to select the type of installation. The differences are detailed in subsections after the standard installation description.

2.2.1 Caché Standard Installation

The standard installation procedure installs both Caché server and client components on the computer. To perform a standard installation follow these steps:

1. If you are upgrading, stop any running Caché server on the computer. Also, close all other Windows applications and shut down the Web server if it is installed on the same computer. For a CD-ROM installation, load the software CD into your CD drive. For a network installation, ensure that you have access to the files.
2. Click **Start** and click **Run** from the menu. In the **Open** text box of the **Run** dialog box, enter:

   ```
   [drive]:\nt\setup.exe
   ```

   The drive value is the letter of your CD-ROM drive. Click **OK** to start the Caché InstallShield Wizard.
3. The Caché setup begins. Within setup, reply to the prompts as they appear:
   - Click **Next** to continue to the next dialog box.
   - Click **Back** to go back to a previous dialog box and change what you have entered.

- Click **Cancel** to stop the installation.

4. **License Agreement** — If there are no instances of Caché on this machine, setup displays the Caché License Agreement. Click **Yes** to confirm that you accept this agreement.
5. Depending on whether other Caché instances exist on this machine, one or both of the following dialog boxes displays:
   a. **Select Caché Instance** — Displays a list of existing installation directories if there are other instances of Caché. You can select one of these directories to update, or choose to install a new instance into a new directory.
   b. **Define Caché Instance Name** — Prompts you to enter a name for a new Caché instance. The default name is CACHE, or if other instances exist *CACHEn*, where *n* is the number of Caché instances including this new one. Accept the default or enter another name. Subsequent updates to this instance maintain the instance name you enter here.
6. **Setup Type** — Select the type of installation you prefer:
   - **Standard** — Installs the server, client, and ODBC driver components. Select this option if you plan to use this computer as a Caché database server.
   - **Client** — Installs only the Caché client components. Select this option if you plan to use the Caché system and configuration utilities on this computer as a client to a Caché database server on another computer. See Caché Client Installation for details.
   - **Custom** — Allows you to select which components you want to install. See Caché Custom Installation for details.

   This dialog box also allows you to select a destination directory for the Caché software for a new instance; the default location is C:\CacheSys (or *Cachen* when multiple instances exist). You can enter another directory by clicking **Browse**. If you enter a directory that does not exist, setup asks if you want to create it, click **Yes** to create it and continue.

   Note: If you are upgrading, you do not have the option of changing the destination directory.

   The following three steps only appear when installing a new instance; upgrade installations keep the settings of the existing instance:
7. **Install Unicode Support** — Select either 8-bit or Unicode support for your installation (the default depends on your operating system locale):
   - **8-bit** — The software handles characters in an 8-bit format.

- **Unicode** — The software handles characters in the Unicode (16-bit) format. Select Unicode if your application uses languages that store data in a Unicode format, such as Japanese.

  InterSystems recommends 8-bit character support for locales based upon the Latin-1 character set, ISO 8859–1. Use Unicode if the base character set for your locale is not Latin-1, or if you plan to have data from locales based upon a different character set. If you use an 8-bit version of Caché, your data is not portable to 8-bit locales based on a different character set.

  CAUTION: If you choose a Unicode installation, you cannot revert to an 8-bit version without potential data loss. This is because an 8-bit version of Caché cannot retrieve 16-bit character data from a database.

8. **Initial Security Settings** — Decide how restrictive you want the initial Caché security settings to be. If you choose **Minimal**, the installation continues with the next step. If you select **Normal** or **Locked Down**, it performs these additional steps:

   a. **User Credentials** — Choose an existing Windows user account under which to run Caché. You can choose the default system account, which runs Caché as the Windows Local System account, or enter a defined Windows user account. When you click **Next**, the installation verifies the following if you enter a defined user account:

      - The account exists on the domain.
      - You have supplied the correct password.
      - The account has local administrative privileges on the server machine.

      Important: If you are using Kerberos, you must enter a defined account that you have set up to run the Caché service. InterSystems recommends you use a separate account specifically set up for this purpose.

   b. **Caché Users Configuration** — The installation creates a Caché account with the %All role for the user who is installing Caché to grant that user access to services necessary to administer Caché. Enter and confirm the password for this account. The password must meet the criteria corresponding to the security setting.

Important: If you select Minimal for your initial security setting, but Caché requires network access to shared drives and printers, you must manually change the Windows user account under which to run the Caché service, choosing an existing or creating a new account the has local administrative privileges on the server machine.

For a detailed explanation of these settings, see the Initial Caché Security Settings section of the "Preparing for Caché Advanced Security" appendix of this guide.

9. **Enter License** — If you do not have a Caché license already on your computer (no cache.key file in the system manager's directory), the **Enter License** dialog box explains that a key was not detected and asks if you want to enter a license during the installation.

   Click **No** to continue with the remaining installation steps. You can update your license information at another time by using the System Management Portal from the **[Home] > [Licensing] > [License Key]** page.

   Click **Yes** to open the Caché License Wizard.

   See the Maintaining License Key section of the "Managing Caché Licensing" chapter of the *Caché System Administration Guide* for details about entering InterSystems Caché licensing information.

10. **Install Summary** — Review the selected installation options and destination directory for the software files. Click **Next** to continue. Setup installs Caché in the selected directory.

    A standard installation sets the following port numbers for your Caché instance:

    - SuperServer port number — `1972` or the first available subsequent number
    - Web server port number — `8972` or the first available subsequent number
    - Telnet port number — `23`

    You may change the SuperServer port value after installation from the **[Home] > [Configuration] > [Memory and Startup]** page of the System Management Portal.

    You may change the Web server and Telnet port values after installation from the **[Home] > [Configuration] > [Advanced Settings]** page of the System Management Portal.

11. If you are installing the Web Server Gateway (CSP Gateway) and a Web server is running, a dialog box appears asking if you want to stop the Web server. If you click **Yes**, the installation procedure stops the server, installs the CSP Gateway, and restarts the server. If you click **No**, the procedure does not configure the CSP Gateway for the Web server;

it only installs the private CSP gateway for the private Apache Web server included with Caché.

12. **InstallShield Wizard Complete** — Indicates the installation has completed successfully. Choose whether you want to see the *Getting Started* page and click **Finish** to start Caché.

    The installation procedure notifies you if a restart is necessary, and prompts you to choose to restart now or later.

After Caché is installed, the **Caché Cube** icon appears in the system tray area of the Windows tool bar. Click the cube to bring up the Caché menu. In addition, there is a **Caché** item on the Windows **Programs** menu.

For Windows servers, the installation names the Caché Windows service "Caché Controller for *instname*," using the name from the **Define Caché Instance Name** dialog box. The service is set up to start automatically as a Windows service when you start your server.

You may change this setting, **Start Caché on System Boot**, from the **[Home] > [Configuration] > [Memory and Startup]** page of the System Management Portal (available from the **Caché Cube**. If this instance is part of a Windows cluster, you must clear the check box to prevent automatic startup, allowing the cluster manager to start Caché.

2.2.2 Caché Client Installation

If you wish, you can install only those parts of Caché that are required on a client machine. These include the Caché system and programmer utilities. The client installation follows a set of dialog boxes similar to a Caché standard installation.

This software is included when you install Caché and does not require a separate installation medium.

To perform a client installation:

1. Follow the first five steps of the standard installation.
2. At the **Setup Type** dialog box, select **Client**.

   The dialog box allows you to select a destination directory for the Caché software; the default location is C:\CacheSys. You can enter another directory by clicking **Browse**.
3. If the directory you enter does not exist, confirm you want it created by clicking **Yes**.
4. **Install Summary** — Review the installation name, type, and destination directory for the software files. Click **Next** to continue. Setup installs Caché in the selected directory.

5. **InstallShield Wizard Complete** — Indicates the installation has completed successfully. Click **Finish**.
6. When the installation is complete, define a preferred server for this client. This procedure is described in the Define a Remote Server Connection section of the "Connecting to Remote Servers" chapter of the *Caché System Administration Guide*.

After Caché is installed on a client, the **Caché Cube** icon appears in the system tray area of the Windows tool bar; though it appears dimmed because there is no Caché server running.

2.2.3 Caché Custom Installation

The Caché installation program allows you to select certain Caché components to be installed on the computer. For example, you may want to install only the Caché Tools and Utilities. Keep in mind that some selections require that you also install other components.

To perform a Caché custom installation:

1. Follow the first five steps of the standard installation.
2. At the **Setup Type** dialog box, select **Custom**.

   The dialog box allows you to select a destination directory for the Caché software; the default location is C:\CacheSys. You can enter another directory by clicking **Browse**.
3. **Select Components** — Select those components you want to install as described in the Caché Custom Components table.
4. **Select SuperServer and Web Server Ports** — You can choose to let Caché assign the port numbers as described in the standard installation procedure or you can enter custom port numbers.
5. Continue with the **Install Unicode Support** step of the standard installation.

The following table lists the components you can select.

Caché Custom Components

Component	Description	Other Required Components
Caché Application Development	Installs development components for building your own applications (C, C++, Java) that use Caché databases.	

Component	Description	Other Required Components
Web Server Gateway (CSP)[1]	Configures the Web server gateway for Caché Server Pages (CSP).	
WebLink	Configures the Web server to use WebLink to connect to Caché databases.	
ActiveX Connectivity	Installs ActiveX to provide connectivity between client applications and Caché.	Caché Direct Connectivity
Documentation	Installs the Caché documentation.	Caché Engine
SQL Tools[2]	Installs ODBC and JDBC drivers.	
Caché Tools and Utilities	Installs development tools (such as the Caché Studio) and system management utilities (such as the System Management Portal).	Caché Direct Connectivity and ActiveX Connectivity
Caché Engine	Installs the base Caché database engine.	Caché Tools and Utilities, Caché Direct Connectivity, ActiveX Connectivity
Caché Direct Connectivity	Installs Caché Direct runtime client components, which are needed for client connectivity.	
Manager Utility Source Code	Installs the utility source code into the %SYS Namespace.	Caché Engine
Caché Engine Link Libraries	Installs C header files and object files which are necessary for relinking Caché on certain Windows platforms.	

[1] The installation detects three types of Web servers and presents options to automatically configure them for the CSP Gateway, see CSP Gateway Install Options for detailed information.

[2] Contains two components: **ODBC Connectivity** and **Java Database Connectivity**. You may choose which of these subcomponents to install by selecting them.

Note: The default selections on the custom menu are **Caché Application Development**, **Web Server Gateway**, **ActiveX Connectivity**, **Documentation**, **SQL Tools** (both subcomponents), **Caché Tools and Utilities**, **Caché Engine**, and **Caché Direct Connectivity**; these are the same components installed in a standard installation.

2.2.3.1 CSP Gateway Install Options

The installation detects three types of Web servers and presents options to automatically configure them for the CSP Gateway. If detected, a check box for each of the following is presented:

- CSP IIS
- CSP for Apache 1.x
- CSP for Apache 2.x

If you select any of these options, Caché installs the main CSP Gateway and configures the corresponding Web server to work with the CSP Gateway. If you select the CSP Gateway component, but clear all these options, it installs all the files required for the CSP Gateway, but does not change the Web server configurations. You can change the configurations manually after installation from the CSP Gateway application.

If you select all available check boxes, Caché installs the main CSP Gateway and configures each Web server.

Caché always installs the private Apache Web server with a private CSP gateway on the selected Web server port for serving the System Management Portal and the online documentation.

Important: Installing the CSP Gateway overlays any changes you previously made in the CSP Gateway configuration.

2.3 Post-Installation Tasks

You can manage your Caché instance using the System Management Portal, which is accessible from the Caché Cube. For more information on this management tool, see the "Using the System Management Portal" chapter of the *Caché System Administration Guide*.

- If you plan to connect remotely to other instances of Caché, follow the procedure described in the Define a Remote Server Connection section of the "Connecting to Remote Servers" chapter of the *Caché System Administration Guide*
- If you are upgrading from a prior version of Caché, you must recompile any Caché Objects applications after installation by running the following:

  ```
  Do $system.OBJ.UpgradeAll("c")
  ```

  This upgrades and compiles the class dictionaries in every namespace.
- If appropriate for your installation, perform any additional tasks described in the Special Considerations section.

2.4 Special Considerations

The following topics describe particular issues or tasks associated with licensing, specific platforms, or kinds of installations:

- Multiple Caché Installation Issues
- Change the Caché Language
- Uninstall Caché
- InterSystems Caché Packet Drivers

2.4.1 Multiple Caché Installation Issues

You can install and simultaneously run multiple instances of Caché 4.0 and later on a single Windows machine. Install Caché as for a single installation, giving each instance a unique name, a unique installation directory, and a unique port number.

Please reference the Multiple Caché Instances section of the *Caché System Administration Guide* for more detailed information.

Installing multiple Caché instances is limited by components where only one exists on a system. For example, typically there is only one Web server on a system; and as such, CSP is configured for the last installation. Caché client components stored in the registry encounter the same issue. The Caché ODBC driver and ActiveX components are stored in the registry using one name for each. Currently the last installation updates these components to point to the last instance installed. If you are adding Caché 5.1 to your machine and keeping an older version running also, run RegFiles.bat in 5.1 to register all the 5.1 components.

Caché makes an effort to move common components to a common directory that can be shared across Caché instances. Unfortunately, because of backward compatibility issues, not all Caché 5.1 components support Caché 5.0 and 4.1 instances, and are even less likely to support Caché 4.0 instances on the same machine.

As a work-around, you can take advantage of a feature Microsoft introduced with Windows 2000 and later. You can force your executable to ignore the registry paths to an executable by creating an empty file of the same name with .local appended to the executable name.

For example: CObjArch.exe would need an empty file called CObjArch.exe.local to force the Architect program to look in the current directory for registered ActiveX components, before using the registry path. By creating these empty .local files, you enable a previous Caché instance to use the compatible local files, rather than a newly installed Caché 5.1 set of registered executables.

To create .local files for all the executables in a directory type the following at a DOS prompt:

```
for %c in (*.exe) do set tempvariable= >%c.local
```

For more information on .local files see the Microsoft article entitled: *The End of DLL Hell*.

2.4.2 Change the Caché Language

When you install Caché, all supported language-specific utility DLLs are installed in the CacheSys\Bin directory. Each DLL contains localized strings and messages.

The format of the name of the DLL is UTILxxx.DLL, where *xxx* is a 3-letter code that signifies the following languages:

Code	Language
DEU	German (Standard)
ENU	English (United States)
ESP	Spanish (Spain)
FRA	French
ITA	Italian (Standard)
JPN	Japanese
KOR	Korean
NLD	Dutch (Standard)
PTB	Portuguese (Brazilian)
RUS	Russian

If you wish to change to the locale of a Caché installation, run the Caché National Language Support application cnls.exe from your <cache-install-dir>/Bin directory and update the information on the **Locale** tab.

2.4.3 Uninstall Caché

To uninstall Caché, first stop Caché and then click **Exit** on the Caché Cube to remove the cube from the system tray.

Use only the Caché-supplied uninstall program, accessible on Windows systems using the **Add or Remove Programs** utility from the Windows **Control Panel**.

Important: Other uninstall programs are not supported and using them may cause unexpected results.

2.4.4 InterSystems Caché Packet Drivers

Install Caché Packet Driver for Windows 2000, XP, and 2003 Server

To use Raw Ethernet, DDP, or LAT with Caché on Windows systems, you must install the appropriate packet driver as described below. First load the CD-ROM or verify the network location of the appropriate driver file.

For Windows 2000, XP, and 2003 Server systems, install the InterSystems Packet Protocol Driver as follows:

1. Right-click **My Network Places** on the desktop and click **Properties**.

   You can also click **Start**, point to **Settings** and click **Network and Dial-up Connections**.

2. Right-click **Local Area Connection** and click **Properties**.
3. Click **Install**.
4. Click **Protocol** in the **Select Network Component Type** dialog box and then click **Add**.
5. Click **Have Disk** in the **Select Network Protocol** dialog box.
6. Enter the path to the packet driver kit and click **OK**.

   You can also enter the appropriate drive letter and click **Browse** to search for the correct path,\drivers\win2k, that contains the file ispkt2k.inf. Click **Open** and then click **OK**.

7. Select **InterSystems Packet Driver for Windows 2000, XP and 2003 Server** and click **OK**.
8. After the driver is installed, click **Close**.

After you restart Windows, Caché is fully available to you.

3

Installing Caché on OpenVMS

This chapter describes how to install Caché 5.1 on an OpenVMS system. It assumes that you are familiar with OpenVMS directory structures, utilities, and commands. This chapter contains the following major sections:

- Installation Requirements
- Caché Installation
- Post-Installation Tasks
- Special Considerations

You may wish to consult the appendix on Calculating System Parameters to verify your parameter settings before you begin the installation, and use the online OpenVMS Parameter Calculator to obtain parameter values that InterSystems recommends.

3.1 Installation Requirements

This section describes the hardware and software requirements for new and upgrade installations of Caché 5.1.

3.1.1 Disk Space Requirements

A standard Caché installation needs approximately 340 MB (megabytes) of disk space depending on the type of installation you choose. This is the equivalent of 696320 blocks of disk space.

3.1.2 Supported Platforms and Web Servers

The latest version of Caché is supported on HP Alpha computers running OpenVMS v7.3-2 and v8.2 and HP Itanium processors running OpenVMS v8.2-1. If you are installing Caché on a Non-Uniform Memory Access (NUMA) machine, such as the GS160, contact the InterSystems Worldwide Response Center (WRC) for current configuration recommendations.

Important: Caché does not support any Web servers on the OpenVMS platform.

3.1.3 Supported Upgrade Paths and Procedures

If you are performing an upgrade, please first read and perform all necessary procedures described in the "Upgrading Caché" chapter.

When upgrading, back up your Caché installation after completing all the pre-installation upgrade tasks and before installing Caché.

3.2 Caché Installation

To install Caché 5.1, log into OpenVMS as the system manager or with CMKRNL, WORLD, BYPASS, SYSLCK, ALTPRI, and OPER privileges. If your UIC is not [1,4] when you run the installation script, the command procedure, **CINSTALL**, automatically changes your UIC to [1,4].

Note: If the logical name *CACHE$MGRUIC* is defined during a new installation, the installation sets the owner UIC of the target directory to the value of that logical. Subdirectories and files inherit that UIC.

Once you are logged into OpenVMS:

1. Transfer files from the distribution media.
2. Run the installation script.
3. Edit the startup file.

3.2.1 Transfer Files from the Distribution Media

Transfer the Caché installation files from the distribution media either to a work directory (recommended) or to the target directory. Use the OpenVMS backup command to copy the distribution files to the work directory. Examples follow:

- For CD distribution:

```
$ MOUNT/OVERRIDE=IDENTIFICATION cd_dev:
$ BACKUP cd_dev:[CACHEDIST]C51Bnnn.BCK/SAVE/SELECT=[REL.ALPHAVMS.DIST...]
         dest_dev:[CACHEKIT...]
$ DISMOUNT cd_dev:
```

  Where cd_dev is the device name of your CD-ROM drive, [CACHEDIST]C51Bnnn.BCK is an example name of the compressed backup file on the CD, dest_dev is the device name for your local hard disk, and [CACHEKIT] is a directory that you create on your local disk to temporarily uncompress and store the installation kit distribution files. Two levels down in the uncompressed kit is CINSTALL.COM.

- For magnetic tape distribution:

```
$ MOUNT/FOREIGN magtape_dev:
$ BACKUP magtape_dev:C51Bnnn.BCK/REWIND/SAVE/SELECT=[REL.ALPHAVMS.DIST...]
         dest_dev:[CACHEKIT...]
$ DISMOUNT magtape_dev:
```

  Where magtape_dev is the device name of your magnetic tape drive, C51Bnnn.BCK is an example name of the compressed backup file on the tape, dest_dev is the device name for your local hard disk, and [CACHEKIT] is a directory that you create on your local disk to temporarily uncompress and store the installation kit distribution files. Two levels down in the uncompressed kit is CINSTALL.COM.

- For backup save set distribution:

```
$ CREATE/DIR dest_dev:[CACHEKIT...]
$ BACKUP C51Bnnn.BCK/SAVE dest_dev:[CACHEKIT...]
```

  Where dest_dev is the device name for your local hard disk, [CACHEKIT] is a directory that you create on your local disk, and C51Bnnn.BCK is an example name of the compressed backup file on the ftp server. This is required to correctly recreate the directory structure on your local disk.

3.2.2 Run the Installation Script

The installation script, CINSTALL, automatically does the following:

- Installs the Caché system manager databases.
- Starts Caché in installation mode.
- Installs Caché system manager globals and routines.
- Shuts down Caché and restarts using the default configuration file. Upgrade installations restart using their original configuration files, updated as necessary.

To perform the installation:

1. Start the installation procedure by running the CINSTALL script, located at the top level of the installation files:

   ```
   @CINSTALL
   ```

   Or, if you are not in the installation directory, use a full path to invoke the script, such as:

   ```
   @DKA0:[CACHEKIT.5-1-0-nnn-0]CINSTALL
   ```

   Where, in this case, DKA0:[CACHEKIT.5-1-0-nnn-0] specifies the directory path where you uncompressed and stored the installation kit distribution files.

2. The installation script displays a list of any existing Caché instances on this machine. At the prompt, enter an instance name. If an instance with this name already exists, the program asks if you wish to upgrade it. If no such instance exists, it asks if you wish to create it and asks you to specify its location on disk. If the directory you specify does not exist, it asks if you want to create it. The default answers are `Yes`; press **Enter** to continue with the installation.

3. You next are asked if you want to install Caché with 8-bit or Unicode character support.

   InterSystems recommends 8-bit character support for locales based upon the Latin-1 character set, ISO 8859–1. Use Unicode if the base character set for your locale is not Latin-1, or if you plan to have data from locales based upon a different character set. If you use an 8-bit version of Caché, your data is not portable to 8-bit locales based on a different character set.

   CAUTION: If you choose a Unicode installation, you cannot revert to an 8-bit version without potential data loss. This is because an 8-bit version of Caché cannot retrieve 16-bit character data from a database.

4. The script then asks if you wish to load the source code for the various system management utilities and load the Caché engine link libraries, which are used for building custom

callin and callout modules. The default options, <YES>, are appropriate for these prompts in most cases.

5. You next decide how restrictive you want the initial Caché security settings to be. Choose from Minimal (1), Normal (2), and Locked Down (3). The default is Minimal; if you choose this, the installation continues with the next step.

   If you enter 2 or 3, the script asks for additional information:

   a. Owner of the instance — Enter a user name under which to run Caché. Caché creates an account for this user.

   b. Password for the entered user — Enter the Caché password for the user name twice to confirm it.

   For a detailed explanation of these settings, see the Initial Caché Security Settings section of the "Preparing for Caché Advanced Security" appendix of this guide.

6. If the installation does not detect a cache.key file in the MGR subdirectory, it asks if you want to enter the license key information; the default is **No**.

   If you choose **Yes**, Caché installs a key as part of the installation process. See the Maintaining License Key section of the "Managing Caché Licensing" chapter of the *Caché System Administration Guide* for details about entering InterSystems Caché licensing information.

3.2.3 Edit the Startup File

For Caché to function properly, you may need to edit the CACHE$STARTUP.COM file to add commands that set up Caché when you restart your machine. The CACHE$STARTUP.COM script should only contain commands that affect all Caché instances, such as hardware-dependent definitions. This also includes defining logicals for magnetic tape devices and starting certain Caché instances automatically. To set up the file:

1. Copy CACHE$STARTUP.COM from your installation subdirectory [.BIN] to SYS$LIBRARY.
2. Make changes in that copy.
3. Edit the OpenVMS system startup command file, SYS$MANAGER:SYSTARTUP_VMS.COM, to invoke the CACHE$STARTUP.COM file.

For further details, read the comments contained within the CACHE$STARTUP.COM file.

3.3 Post-Installation Tasks

Once you have completed running the installation procedure, perform the following tasks:

- Install Caché on a Web Server.
- Start Caché.
- Set OpenVMS Time for Use with Caché.
- If you are upgrading, Perform Post-installation Upgrade Tasks.
- If appropriate for your installation, perform any additional tasks described in the Special Considerations section.

3.3.1 Install Caché on a Web Server

Caché does not install a private Web server on a OpenVMS instance; therefore to use the System Management Portal to manage a Caché server on OpenVMS, you must install the CSP Gateway on a supported Web server machine and configure it to remotely control the Caché instance on OpenVMS.

The version of Caché on the Web server, in most cases, must be the same or a later version of the Caché system it manages.

See the Accessing the System Management Portal section of the "Using Caché on OpenVMS" chapter of the *Caché System Administration Guide* for instructions on installing the CSP Gateway component of Caché on a separate Web server to manage the OpenVMS Caché instance you just installed.

3.3.2 Start Caché

When Caché is installed it is left running. However, if you need to start Caché, first log into your operating system, then start Caché using the **ccontrol** command:

```
ccontrol start <instname>
```

Where *instname* is the instance name that you chose during the installation.

Use the **ccontrol** command to start and stop Caché. It is described in greater detail in the Controlling Caché Instances section of the *Caché System Administration Guide*.

Once Caché is started, initiate a Caché session using the **csession** command:

```
csession <instname>
```

Where *instname* is the instance name that you chose during the installation.

For more information, see the "Using Caché on OpenVMS" chapter of the *Caché System Administration Guide*.

3.3.3 Set OpenVMS Time for Use with Caché

Beginning with OpenVMS version 7.0, C programs that use UTC time functions (including Caché) require the proper setup for the DEC C Runtime Library interface; they may otherwise obtain incorrect time values. This procedure is fully described in the OpenVMS documentation. In brief:

1. Check if times in Caché and OpenVMS are synchronized, using the following commands at the Caché prompt:

   ```
   > Write $ZTIME($P($H,",",2))
   > !SHOW TIME
   ```

   The first displays the Caché time and the second displays the OpenVMS system time. The two values should be approximately the same.

2. Run the UTC$TIME_SETUP procedure to set the system time.

3. If this procedure exits prematurely, enter the following command at the OpenVMS prompt:

   ```
   $ @SYS$MANAGER:UTC$TIME_SETUP.COM "" BOTH
   ```

   This command forces an update of both the time zone and Time Differential Factor (TDF) data. Make sure these values are consistent. For example, if the selected main time zone is US (33), Eastern zone (6) or EST, the appropriate TDF is -5:00 (or -4:00, if Daylight Saving Time is in effect).

   You can configure OpenVMS to automatically adjust for Daylight Saving Time as described in the OpenVMS documentation.

3.3.4 Perform Post-installation Upgrade Tasks

If you are upgrading, there are additional upgrade tasks to perform after the installation is complete:

- *Recompile Objects* — you must recompile any Caché Objects applications after installation if you are upgrading from a prior version of Caché by running the following:

  ```
  Do $system.OBJ.UpgradeAll("c")
  ```

This upgrades and compiles the class dictionaries in every namespace.

- *Update Cluster Configurations (OpenVMS clusters only)* — prior to Caché 5.0 when configuring a Caché cluster it was necessary to define the network type for the cluster (UDP or Ethernet) and define DCP connections between the cluster members. This is no longer necessary. ECP is the default for new installations.

  After the upgrade to 5.1 from 4.1, a Caché cluster configuration needs to be changed manually. The changes are in two places of the System Management Portal:

  - Enable the ECP service from the **[Home] > [Security Management] > [Services]** page. Click **`%Service_ECP`**, select the **Service enabled** check box, and click **Save**.
  - From the **[Home] > [Configuration] > [Legacy Network Connections]** page, delete any DCP connections from the network table that you do not need; they were only there to support clusters.

  The upgrade does not automatically make these changes because it cannot detect which DCP connections support clusters and which might be for communicating with machines that are still running a prior release of Caché. With ECP, networking the cluster automatically configures the network tables as needed; it is not necessary to define any ECP connections between the cluster members to support Caché clusters. However, you cannot access the ECP connection created automatically. If the configuration requires ECP, to gain read-only access to privately mounted databases on another cluster member, you must define those connections.

3.4 Special Considerations

The following sections describe particular issues or tasks associated with licensing, specific platforms, or kinds of installations:

- Multiple Caché Instances
- Patch for Client Access Using Kerberos on Itanium Platform
- SYSTEM-F-SECTBLFUL Error

3.4.1 Multiple Caché Instances

You can install and simultaneously run multiple instances of Caché 4.0 and later on a single OpenVMS machine. Install Caché as for a single installation, giving each instance a unique name, a unique installation directory, and unique port numbers.

Please reference the Multiple Caché Instances section of the *Caché System Administration Guide* for further details.

3.4.2 Patch for Client Access Using Kerberos on Itanium Platform

Applications attempting to access OpenVMS servers that use Kerberos authentication must install the patch, `HP-I64VMS-TCPIP-V0505-11ECO1-1`. The ECO is for TCP/IP, not the actual operating system. Without this patch, the server often transmits erroneous response packets back to clients using the C++ binding, ODBC, JDBC, or Studio.

Note: This ECO applies only to OpenVMS on Itanium hardware. It is not needed for OpenVMS on Alpha.

3.4.3 SYSTEM-F-SECTBLFUL Error

Contact your system administrator or the WRC if you encounter an error similar to the following when installing Caché:

```
%INSTALL-E-FAIL, failed to CREATE entry for STRIPE1$:<CACHESYS.BIN>CACHE.EXE
-SYSTEM-F-SECTBLFUL, process or global section table is full
```

Also see the Calculating OpenVMS Parameters section of the "Calculating System Parameters for OpenVMS" appendix of this guide for setting values for the *GBLSECTIONS* parameter.

4

Installing Caché on UNIX and Linux

This chapter describes how to install Caché 5.1 on a UNIX or Linux system. It assumes that you are familiar with UNIX and Linux directory structures, utilities, and commands. This chapter contains the following major sections:

- Installation Requirements
- Caché Installation
- Post-Installation Tasks
- Special Considerations

You may wish to consult the appendix on Calculating System Parameters to verify and adjust your parameter settings before you begin the installation.

4.1 Installation Requirements

The following sections describe the hardware and software requirements for new and upgrade installations of Caché 5.1.

4.1.1 Disk Space Requirements

A standard Caché installation that includes support for Caché Server Pages (CSP), needs 370 – 450 MB (megabytes) of disk space depending on the type of installation you choose.

4.1.2 Supported Operating Systems and Features

The latest version of Caché is supported on the following operating systems. For each operating system, the table indicates the availability of selected Caché features, including Unicode, Web servers, and client platforms. For the particular versions and platforms supported for each operating system, see the Supported Operating Systems table of the *Caché Supported Platforms* guide.

Caché 5.1 Supported Platforms and Web Servers

Operating System[1]	Unicode Support	Apache Web Server Support	ODBC Client Support	C++, Perl, and Python Client Support
Tru64 UNIX	Yes	Yes	Yes	No
HP-UX	Yes	Yes	Yes	No
AIX	Yes	Yes	Yes	No
Red Hat Enterprise Linux AS	Yes	Yes	Yes	No
Sun Solaris (64-bit only)	Yes	Yes[2]	Yes	No
SUSE Linux Enterprise Server	Yes	Yes	Yes	No

[1] See the following topics for each operating system for important platform-specific information:

- See the Tru64 UNIX Platform Notes section of the *Calculating System Parameters for UNIX and Linux* appendix for detailed configuration information.
- See the HP-UX topic of the *Special Considerations* section for required patches.
- See the IBM AIX topic of the *Special Considerations* section for system requirements and environment considerations.
- See the Red Hat Linux topic of the *Special Considerations* section for information regarding shared memory size and the MQ Series interface.
- See the Sun Solaris topic of the *Special Considerations* section for environment considerations.

- See the SUSE Linux topic of the *Special Considerations* section for information regarding shared memory size.

[2] Sun Solaris (SPARC) also supports the Sun Java System Web Server.

Caché installs a private Apache Web server with each instance to assure proper operation of the System Management Portal and the Caché Online Documentation.

If you are using a Web server for CSP other than the private one installed with Caché, you must install it before installing Caché. Its support on each operating system is dependent on the operating system vendor and is subject to change. See the Web Server Configuration section of the "CSP Configuration" chapter of the *Using Caché Server Pages* guide for more information.

4.1.3 Supported Upgrade Paths and Procedures

If you are performing an upgrade, please first read and perform all necessary procedures described in the "Upgrading Caché" chapter of this guide.

When upgrading, back up your Caché instance after completing all the pre-installation upgrade tasks and before installing Caché.

4.2 Caché Installation

To install Caché 5.1, log in as user ID `root`. It is acceptable to **su** (super user) to `root` while logged in from another account. Once you are logged into your operating system:

1. Transfer files from the distribution media.
2. Run the installation script.
3. Install Caché Client on Windows for Development.

Important: Do not use symbolic links for any Caché directory; unexpected results can occur.

4.2.1 Transfer Files from the Distribution Media

Caché for UNIX-based platforms is distributed either on CD-ROM or acquired from InterSystems in a compressed file with a tar.Z extension. There is also a single-user RPM (Redhat

Package Manager) Linux kit available for download from the InterSystems Web site. The # represents the UNIX prompt in the following examples.

Transfer the Caché installation files by choosing the one appropriate method from the following:

- Mount the distribution CD.
- Uncompress and untar the downloaded tar.Z file:

  ```
  # uncompress -c cache-x.x.x.xxx-platform.tar.Z | tar xvf -
  ```

  Where "x.x.x.xxx" is the Caché build identifier and "platform" is the identifier of the specific platform kit, for example, cache-5.1.0.825-linux.tar.Z.
- Unpack the RPM package file:

  ```
  # rpm -ivh cache-x.x.x-x.i386.rpm
  ```

  Where "x.x.x-x" is the Caché build identifier, for example, cache-5.1.0-1.i386.rpm.

  Run the installation from the /usr/cachekit/x.x.x directory where, "x.x.x" is the Caché release identifier, for example: /usr/cachekit/5.1.0.

Note: By default, the Caché RPM kit installs into /usr/cachekit/5.1.0. Your /usr directory may be mounted read-only or may contain little free space, so you may want to change the location.

4.2.2 Mount the Distribution CD

For the CD distribution, install directly from the CD, using the **mount** command. The device name and options for a CD **mount** command vary between systems. Typically, the **mount** command is:

```
# mount <device> <mountpoint>
```

Where *<device>* specifies the name of the CD-ROM device and *<mountpoint>* specifies the pathname of the directory to which the CD is mounted.

CD Mount Commands

Operating System	Sample Mount Command
Tru64UNIX	**mount /dev/disk/cdrom0a /cdrom**
HP-UX	**mount /dev/dsk/c1t2d0 /cdrom**
IBM AIX*	**mount /dev/cd0**
Red Hat Linux	**mount /dev/cdrom /mnt/cdrom -o map=o**
Sun Solaris	Mounted automatically (no command needed)
SUSE Linux	**mount /dev/cdrom /mnt/cdrom -o map=o**

* The IBM-AIX mount command accepts only the device or the mount point, not both.

For some operating systems, you must set up devices and mount points in the file system table (preferably through the system management utility). Note that devices and mount points vary by machine; those listed above are merely samples. Consult your operating system documentation for more information.

After running the installation script, unmount the CD with the **umount** command.

4.2.3 Run the Installation Script

The installation script, **cinstall**, automatically does the following:

- Installs the Caché system manager databases.
- Starts Caché in installation mode.
- Installs Caché system manager globals and routines.
- Shuts down Caché and restarts using the default configuration file (cache.cpf). Upgrade installations restart using their original configuration files, updated as necessary.

To perform the installation:

1. Start the installation procedure by running the cinstall script, located at the top level of the installation files:

   ```
   # /pathname/cinstall
   ```

   where pathname is the CD mount point or the directory where the downloaded tar or rpm file is unpacked.

2. The installation script identifies your system type and validates it against the installation type on the distribution media.

   If your system supports more than one type, for example, nonclustered and clustered, or if the install script cannot identify your system type, you are asked additional questions. If your desired system type does not match that on the distribution media, the installation stops. Contact the InterSystems Worldwide Response Center (WRC) for help in obtaining the correct distribution.

3. Next the script displays a list of any existing Caché instances on this machine. At the instance prompt, enter an instance name. If an instance with this name already exists, the program asks if you wish to upgrade it. If no such instance exists, it asks if you wish to create it and asks you to specify its location on disk. If the directory you specify does not exist, it asks if you want to create it. The default answers are `Yes`; press **Enter** to continue with the installation.

4. Next you select the setup type. The default is `Standard`, which installs the most commonly used Caché components. If you would like to choose which components to install, enter `2` for `Custom`. See Caché Custom Installation for details on selecting the custom components.

   The following prompts only appear when installing a new instance; upgrade installations keep the settings of the existing instance:

5. Install Caché with Unicode character support. The default is `No`, which installs the 8-bit Caché.

   InterSystems recommends 8-bit character support for locales based upon the Latin-1 character set, ISO 8859–1. Use Unicode if the base character set for your locale is not Latin-1, or if you plan to have data from locales based upon a different character set. If you use an 8-bit version of Caché, your data is not portable to 8-bit locales based on a different character set.

   CAUTION: If you choose a Unicode installation, you cannot revert to an 8-bit version without potential data loss. This is because an 8-bit version of Caché cannot retrieve 16-bit character data from a database.

6. If you have a supported Web server installed, you are asked if you want to install the CSP Gateway. Answer **`Yes`** to install the CSP Gateway after the Caché installation completes and configure it for your Web server.

7. You next decide how restrictive you want the initial Caché security settings to be. Choose from Minimal (`1`), Normal (`2`), and Locked Down (`3`). The default is Minimal; if you

choose this, the owner of the instance becomes `root` and the installation continues with the next step.

If you enter `2` or `3`, the script asks for additional information:

a. Owner of the instance — Enter a user name under which to run Caché. Caché creates an account for this user.

b. Password for the entered user — Enter the Caché password for the user name twice to confirm it.

For a detailed explanation of these settings, see the Initial Caché Security Settings section of the "Preparing for Caché Advanced Security" appendix of this guide.

8. At this point in the installation, you are asked which group should be allowed to start and stop Caché. Only one group can have these privileges and must be listed in the /etc/group file. The options are:

   - Enter the name or user ID number of an existing group; Caché verifies that the group exists before proceeding.
   - Enter `0` (zero) if you want the group to which the `root` user belongs to be the only group to start and stop Caché.

   Note: The permissions on the <cache-install-dir>/bin directory are modified at installation/upgrade time to remove write access by group and other. Since the owner is `root`, only the system administrator is able to modify files in this directory.

9. If the installation does not detect a cache.key file in the mgr subdirectory, it asks if you want to enter the license key information; the default is **`No`**.

   If you choose **`Yes`**, Caché installs a key as part of the installation process. See the Maintaining License Key section of the "Managing Caché Licensing" chapter of the *Caché System Administration Guide* for details about entering InterSystems Caché licensing information.

10. You are asked to review the installation options you entered and if you want to proceed. Press enter to continue; the installation copies files and displays various messages as it progresses.

    A standard installation sets the following port numbers for your Caché instance:

    - SuperServer port number — `1972` or the first available subsequent number
    - Web server port number — `8972` or the first available subsequent number

- Telnet port number — 23

You may change the SuperServer port value after installation from the **[Home] > [Configuration] > [Memory and Startup]** page of the System Management Portal.

You may change the Web server and Telnet port values after installation from the **[Home] > [Configuration] > [Advanced Settings]** page of the System Management Portal.

11. If you chose to install the CSP Gateway, the installation begins. See the "Web Servers for UNIX, LINUX, and Mac OS X" chapter of *Installing and Configuring the CSP Gateway* for more information.

When the installation completes, you are directed to the appropriate URL for the System Management Portal to manage your Caché system. See the "Using the System Management Portal" chapter of the *Caché System Administration Guide* for more information.

4.2.3.1 Caché Custom Installation

If you choose a custom installation, you must answer questions about installing the following individual components:

The following lists the components you can select:

- Manager Utility Source Code — Installs the utility source code into the %SYS namespace.
- Caché Engine Link Libraries — Installs C header files and object files used for building custom callin and callout modules.
- SQL Tools — Installs ODBC and the SQL Gateway.
- SuperServer and Web Server Ports — You can choose to let Caché assign the port numbers as described in the standard installation procedure or you can enter custom port numbers.

4.2.4 Install Caché Client on Windows for Development

Caché installs a private Apache Web server so that you can access the System Management Portal; therefore, a UNIX-based system does not require a Caché client on a Windows machine to perform system configuration and management tasks.

You do, however, require a Windows client to use the Studio development tool. The procedure is described in the "Installing Caché on Windows" chapter of this guide. You can install the client-only option. Once installed, from the Caché Cube of the Windows client:

- Point to **Preferred Server** and click **Add/Edit** to add a remote server connection to the Caché instance just installed. Make sure you specify the appropriate port numbers for this connection.
- Point to **Remote System Access**, point to **Terminal**, and then click the appropriate connection server name you entered in the previous step.

4.3 Post-Installation Tasks

Once you have completed running the installation procedure, perform the following tasks:

- Start Caché
- If you are upgrading from a prior version of Caché, you must recompile any Caché Objects applications after installation by running the following:

  ```
  Do $system.OBJ.UpgradeAll("c")
  ```

  This upgrades and compiles the class dictionaries in every namespace.
- If appropriate for your installation, perform any additional tasks described in the Special Considerations section.

4.3.1 Start Caché

When Caché is installed it is left running. However, if you need to start Caché, first log into your operating system, then start Caché using the **ccontrol** command:

```
ccontrol start <instname>
```

Where *instname* is the instance name that you chose during the installation.

Use the **ccontrol** command to start and stop Caché. It is described in greater detail in the Controlling Caché Instances section of the *Caché System Administration Guide*.

Once Caché is started, initiate a Caché session using the **csession** command:

```
csession <instname> [parameters]
```

Where *instname* is the instance name that you chose during the installation.

For more information, see the "Using Caché on UNIX, Linux, and Mac OS X" chapter of the *Caché System Administration Guide*.

4.4 Special Considerations

The following sections describe particular issues or tasks associated with specific platforms or kinds of installations:

- Multiple Caché Instances
- HP-UX
- IBM AIX
- Red Hat Linux
- Sun Solaris
- SUSE Linux

4.4.1 Multiple Caché Instances

You can install and simultaneously run multiple instances of Caché 4.0 and later on a single machine. Install Caché as for a single installation, giving each instance a unique name, a unique installation directory, and a unique port number.

Please reference the Configuring Multiple Caché Instances section of the *Caché System Administration Guide*.

4.4.2 HP-UX

HP-UX Parameters

See the HP-UX Platform Notes section of the *Calculating System Parameters for UNIX and Linux* appendix for detailed configuration information.

HP-UX Random Number Generator

Caché requires the optional HP-UX Strong Random Number Generator component for true entropy for its cryptographic random number generator. This component is available for free download at:

http://www.software.hp.com/portal/swdepot/displayProductInfo.do?productNumber=KRNG11I

4.4.3 IBM AIX

There are several issues with AIX:

- AIX Default Parameters
- PowerPC System Requirements
- Shared Library Environment Variable for Caché Engine Link Libraries
- Use of DDP and Raw Ethernet

4.4.3.1 AIX Default Parameters

The default settings of several AIX parameters can adversely affect performance. For detailed information on the settings and recommendations, see the AIX Platform Notes and AIX Default Parameters sections of the *Calculating System Parameters for UNIX and Linux* appendix.

4.4.3.2 PowerPC System Requirements

Current versions of Caché are only supported on PowerPC. PowerPC AIX 4.3 and later support a 64-bit runtime environment on 64-bit hardware. AIX 5.1 and later support a 64-bit kernel on POWER4, POWER5, or equivalent hardware.

The following table lists the allowable combinations of runtime, kernel, and hardware configurations:

AIX PowerPC Configuration Combinations

Combination	Runtime	Kernel	Hardware
1	32-bit	32-bit	32-bit
2	32-bit	32-bit	64-bit
3	64-bit	32-bit	64-bit
4	32-bit	64-bit	64-bit
5	64-bit	64-bit	64-bit

Note: Combinations 4 and 5 require AIX 5.1 or later.

64-bit Caché requires combination 5.

To determine whether the CPU is 64-bit, issue the following command:

```
# bootinfo -y
```

This returns `64` for POWER3, POWER4, or POWER5 and returns `32` on older PowerPC systems.

To determine whether the 64-bit runtime environment is installed, issue the following command:

```
# lslpp -l bos.64bit
```

To determine whether the 64-bit runtime environment is loaded at boot, issue the following command:

```
# grep load64bit /etc/inittab
```

To determine whether the system is running the 64-bit kernel, issue the following command:

```
# bootinfo -K
```

Alternatively, confirm that /unix points to unix_64, rather than unix_up or unix_mp:

```
# ls -l /unix
... /unix -> /usr/lib/boot/unix_64
# file unix
    unix:           symbolic link to /usr/lib/boot/unix_64
```

To summarize, 64-bit Caché for PowerPC AIX requires that the 64-bit runtime environment be loaded on the 64-bit kernel; this requires AIX 5.1, or later, on POWER4, POWER5, or equivalent hardware.

4.4.3.3 Shared Library Environment Variable for Caché Engine Link Libraries

The Caché Engine link libraries contain a batch file that references any installed C linker.

If you have either the standard UNIX C libraries or any proprietary C libraries defined in the *LIBPATH* environment variable, then your environment is ready.

If not, append the paths for the standard UNIX C libraries to *LIBPATH*; these paths are /usr/lib and /lib.

4.4.3.4 Use of DDP and Raw Ethernet

On IBM AIX machines, in order to use DDP and raw Ethernet, the machine must have the DLPI (Data Link Provider Interface) packages installed. If the machine does not have the DLPI packages, then obtain them from your IBM provider and create DLPI devices through the following procedure:

1. Log in as root.

2. In the PSE drivers section of the /etc/pse.conf file, uncomment the four lines that refer to the DLPI drivers.
3. Save the file.
4. Restart the computer.

If the DLPI devices are not installed, the Caché ObjectScript $ZU(114) function returns a null string rather than information about the Ethernet device.

4.4.4 Red Hat Linux

The default shared memory limit (*shmmax*) on Linux platforms is 32 MB. This value is too small for Caché, but it can be changed in the proc file system without a restart. See the Red Hat Linux Platform Notes section of *Calculating System Parameters for UNIX and Linux* for more information.

Red Hat Enterprise Linux V4 requires Websphere MQ version 6.0 to use the MQ interface.

4.4.5 Sun Solaris

The following is a list of Sun Solaris patches you may require to use Caché:

- SunOS 5.9 for Solaris SPARC Release 9 requires Patch ID `111712` or greater. You can download this patch from the Sun Update Connection - Patches and Updates page by entering this patch number in the **PatchFinder** section.
- Using Kerberos on Solaris SPARC Release 9 requires two Patch IDs `112907-06` and `112908-22`. You can download these patches from the Sun Update Connection - Patches and Updates page by entering each patch number in the **PatchFinder** section.
- Using Kerberos on Solaris SPARC Release 10 requires two Patch IDs `120469-03` and `121239-01`. You can download these patches from the Sun Update Connection - Patches and Updates page by entering each patch number in the **PatchFinder** section.

If the Ethernet adapters are protected against access by non-root users, the Caché ObjectScript $ZU(114) function invoked by a non-root user returns a null string rather than information about the Ethernet device.

See the Sun Solaris Platform Notes section of *Calculating System Parameters for UNIX and Linux* for more information.

4.4.6 SUSE Linux

The default shared memory limits (*shhmax* and *shmall*) on SUSE Linux platforms are too small for Caché, and can be changed in the proc file system without a restart.

See the SUSE Linux Platform Notes section of the *Calculating System Parameters for UNIX and Linux* appendix for detailed configuration information.

5

Installing Caché on Macintosh

This chapter describes how to install Caché 5.1 on a Apple Mac OS X system. It assumes that you are familiar with Mac directory structures, utilities, and commands. This chapter contains the following major sections:

- Installation Requirements
- Caché Installation

5.1 Installation Requirements

This section describes the hardware and software requirements for installations of Caché 5.1:

Disk Space Requirements

A standard Caché installation that includes support for Caché Server Pages (CSP), needs approximately 395 MB (megabytes) of disk space depending on the type of installation you choose.

Supported Platforms and Web Servers

The latest version of Caché is supported on version 10.3 and 10.4 of the Apple Mac OS X operating system. For Mac OS X, the Caché Server Pages (CSP) technology is supported on the Apache Web server (version 1.3 on Mac OS X 10.3 and version 2.0 on Mac OS X 10.4).

If using CSP, you must install the Web server before installing Caché for the installation to configure the Web server. Its support on each operating system is dependent on the operating system vendor and is subject to change. See the Web Server Configuration section of the "CSP Configuration" chapter of the *Using Caché Server Pages* guide for more information.

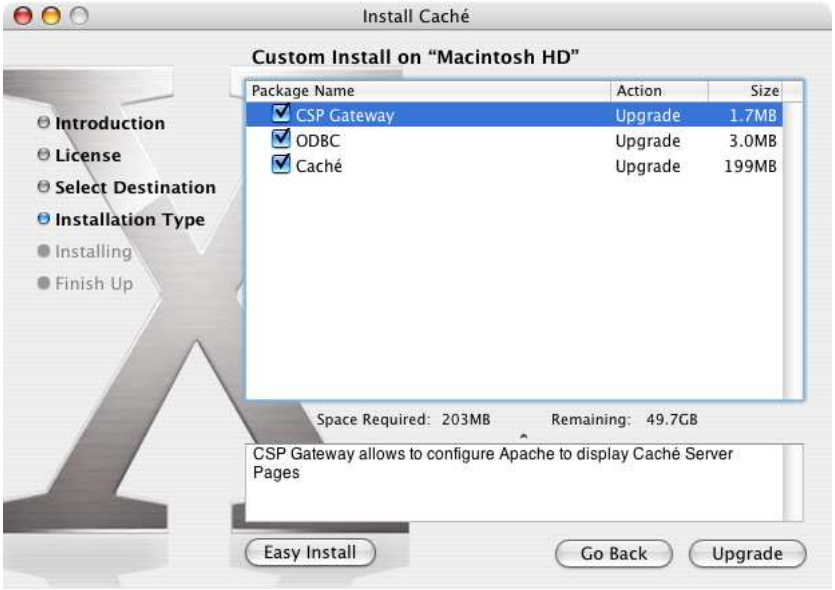
5.2 Caché Installation

In most cases, the Caché installation is much like installing other software products on the Macintosh OS, and does not require an archive file. For cases where you are installing multiple instances of Caché on one machine, see the Caché UNIX-based Installation section for detailed instructions. Otherwise, the procedure is straightforward:

1. Obtain the Caché disk image file (with a .dmg extension) from InterSystems.
2. If the source is on a CD, it automatically mounts and displays a window containing the image file, Cache_5.1.0.5700.0.2117.dmg, for example. Double-click the file to open a Finder window.
3. The new window displays two files: Cache.mpkg and a Packages directory. Double-click Cache.mpkg to start the installation.
4. The **Welcome to Caché Installer** window displays. The six steps involved in a Caché installation appear on the left-hand side of the window:

   Click **Continue** to begin the Caché installation.
5. The Caché Software License Agreement displays. You must click **Agree** to accept the license agreement before you can **Continue**.
6. Select the Macintosh HD volume as the destination and click **Continue**.

7. Next, choose the installation type. If there are no instances of Caché on this machine, this is a new install, otherwise it is an upgrade. You may choose an **Easy Install** or a **Custom Install**.



8. For a custom install, choose any or all of the three components:
   - CSP Gateway (required to view Caché online documentation)
   - ODBC
   - Caché Database Engine

   Clicking **Easy Install** installs all three components. If you plan to use this node only as a client, you may not need to install the Caché database engine.

9. After choosing the installation type, click **New** or **Upgrade** (only the applicable button appears). The install begins after asking you to authenticate that you have the correct privileges to install Caché. Enter your name and password and click **OK**.

10. As the installation completes, you see various messages with a progress bar and finally the "Software installed successfully" message. The installer places Caché in the /Applications/Cache folder and runs from port `1972` or the first available port number after that. It names the instance CACHE. To finish the installation, click **Restart**.

5.2.1 System Management Portal

You can manage your Caché instance using the System Management Portal, which is accessible from your Web browser; go directly to the System Management Portal url: http://<machine>:<port>/csp/sys/UtilHome.csp, where <*machine*> is the IP address of your system (such as `localhost`) and <*port*> is the port number of the Web server installed by Caché.

For example, the default location on the local server of a single installation of Caché is http://localhost:8972/csp/sys/UtilHome.csp.

For more information on this management tool, see the "Using the System Management Portal" chapter of the *Caché System Administration Guide*.

5.2.2 Uninstall Caché

To uninstall a Caché instance that was installed with the Mac OS X installer, perform the following from the terminal as *root*:

```
cd /Applications/Cache      ; ./cstop
rm -rf /Applications/Cache
rm -rf /Library/Receipts/Engine.pkg
rm -rf /Library/Receipts/ODBC.pkg
rm -rf /Library/Receipts/CSPGateway.pkg.
```

5.3 Caché UNIX-based Installation

The alternative installation of Caché on the Mac OS X is much like the installation on any UNIX-based platform. To install Caché 5.1, log in as userid *root*. It is acceptable to **su** (super user) to *root* while logged in from another account.

Once you are logged into your operating system, obtain the installation kit either on a CD-ROM which mounts automatically or from InterSystems in a compressed archive file. The Mac OS X tool, StuffIt Expander, automatically uncompresses the archive file and leaves the install files on the desktop.

Before installing Caché, see the Adjust Kernel Parameters section for information on shared memory adjustments.

After adjusting parameters, you can now follow the instructions in the Run the Installation Script section and subsequent sections of the "Installing Caché on UNIX and Linux" chapter of this guide.

5.3.1 Adjust Kernel Parameters

When using the UNIX-type installation, it may be necessary to adjust the shared memory kernel parameters of the operating system before installing Caché to avoid a shared memory problem illustrated by the following example:

```
Configuring minimum system...
Unable to allocate 1 MB global buffer space...

Unable to allocate shared memory
Cache: Invalid argument
Cache failed to start.
Check if shared memory requirements exceed system resources.
Call InterSystems Technical Support if you need assistance.

** Installation aborted **
```

Verify that the shared memory meet the minimum requirements. If it is necessary, increase the shared memory on the Mac OS X system in the /etc/sysctl.conf file to at least the following values:

```
kern.sysv.shmmax: 67108864
kern.sysv.shmall: 16384
kern.sysv.shmmin: 1
kern.sysv.shmseg: 4
kern.sysv.shmmni: 32
```

Important: Do not make these updates to the kernel parameters in the /etc/rc file; an operating system maintenance release could replace them.

A

Calculating System Parameters for OpenVMS

This document is part of "Installing Caché on OpenVMS" . It explains how you can calculate the best parameters for your system.

For optimal Caché performance, you need to calculate proper values for certain Caché system parameters. These values allow you to determine whether you need to adjust certain OpenVMS system parameters. The values you choose should minimize swapping and paging that require disk accesses, and thus improve system performance.

Review this section carefully and calculate the proper values for both your operating system and Caché before proceeding. Use the *OpenVMS Parameter Calculator* and the tables provided here to record the current and calculated values for your system parameters. You can then refer to these tables when you install Caché. After your system is running, you may need to adjust these values to gain optimal performance.

If you are not already familiar with the memory organization at your operating system level, consult the appropriate system documentation and read the overviews of process and physical memory allocation provided in the next few sections:

- OpenVMS Process Memory Organization
- OpenVMS Page Organization
- OpenVMS Physical Memory Allocation
- How Caché Uses OpenVMS Memory

Once you are familiar with OpenVMS memory organization, use the following sections to tune the parameters for Caché:

- Calculating OpenVMS Parameters
- Analyze the Calculation Results

A.1 OpenVMS Process Memory Organization

An OpenVMS process is any entity that can be scheduled. It may be interactive or run as a batch process. If the OpenVMS process creates subprocesses, that process and all its subprocesses together are called a job.

Many operating system tasks are OpenVMS processes. All processes, whether they are interactive, batch, or operating system-specific, share the structures that this section discusses. OpenVMS can run several processes concurrently. Depending on the size of your computer, OpenVMS could be running between 4 and 1,000 or more simultaneous processes.

Each process uses a share of physical memory, where it stores certain process-specific operating system information, any code the process is using, and any data that it uses or generates. For example, if a programmer edits a routine with an OpenVMS editor, OpenVMS loads the editor into memory, allocates space for the editor's buffers and variables and loads all or a portion of what is being edited into memory.

A.1.1 Shared Memory and Caché

Sometimes code or data is available to more than one process. The editor is one example of this type of code. Another example is the code for Caché and some of the data created with Caché.

Any piece of code or data that several processes can share is stored in a shared section of memory. Shared sections are created by the first user of a system, or during system initialization. They are then made available to subsequent users of that system rather than being duplicated and stored with each individual process.

Caché system code, global buffers, and routine buffers are stored in shared sections of memory.

A.1.2 How OpenVMS Measures Process Memory

OpenVMS measures the code and data area that a process requires in *pages*. Shared sections are also measured in pages, sometimes called shared or global pages.

On OpenVMS systems, page size is not fixed. Normally, it is 8192 bytes. Each page is divided into *pagelets* that are 512 bytes long.

A.2 OpenVMS Page Organization

Sometimes a process needs to access many more pages than can fit within the physical memory allocated to the process. When this situation arises, OpenVMS stores the extra pages on disk. The operating system keeps only the most active, or most recently used, pages in memory. The total of all the pages of a process, whether they are stored in memory or on disk, is called the *virtual memory set* of the process. The pages currently stored in memory are called the *physical working set* of the process, often called *working set*. The status of each page is kept in the process's working set list.

OpenVMS keeps track of free pages available to processes in a free page list. It also keeps a modified page list, which tracks pages that have been changed by a process and must be written to disk before being made available on the free page list.

Sometimes a process needs access to a page not currently in its physical working set. When this happens, OpenVMS searches two places for that page. First, OpenVMS looks in the modified and free page lists, because the page might still be available in physical memory. If the page does not appear on either of these lists, OpenVMS then retrieves the page from disk.

When OpenVMS must retrieve a page, the process has incurred a *page fault*. When the new page is in the modified or free page list, the page fault is called a *soft page fault*. When the page is only available on disk, the page fault is called a *hard page fault*. While OpenVMS retrieves the page from disk, the process becomes inactive. It remains in a page fault wait state until OpenVMS has successfully retrieved the necessary page. The length of time that this procedure requires varies according to the size of the disk, its rotation speed, and the load on the system.

When OpenVMS retrieves the new page, it places one of the current working set pages in the modified or free page list. OpenVMS chooses a page that the process has not used recently.

You can minimize soft page faults by setting the OpenVMS process parameters *WSDEFAULT*, *WSQUOTA*, and *WSEXTENT* appropriately for each process.

Note: The Caché **SETQUOTA** utility lets you modify the default values of quota parameters for jobbed processes, including the network daemons and job servers. **SETQUOTA** can be set to run automatically during multiuser startup, or interactively after startup.

A.3 OpenVMS Physical Memory Allocation

In OpenVMS, you use the **SYSGEN** utility to set system limits on process memory use. You can then set process-by-process (user-by-user) limits with the **AUTHORIZE** utility.

The table below describes the OpenVMS system parameters that affect the memory available to a process's working set.

OpenVMS Working Set Parameters

Parameter	Definition
WSDEFAULT	Initial size of the working set in 512-byte pagelets.
WSQUOTA	Maximum guaranteed amount of physical memory, in pagelets, that OpenVMS allocates to a given process. WSQUOTA is limited by the value of the SYSGEN parameter WSMAX. If WSQUOTA is larger than WSMAX, the user only has access to the amount of memory in WSMAX.
WSEXTENT	Indicates the maximum amount of physical memory which the process can own. The amount WSEXTENT less WSQUOTA is called "borrowed" memory, which is only allocated to the process if the system's current free list size exceeds the amount specified in the parameter BORROWLIM.
WSINC	The number of pagelets by which the working set size of a process is increased when it needs to grow.
WSDEC	The number of pagelets by which the working set size of a process is decreased when it needs to be reduced in size.

Parameter	Definition
BORROWLIM	Minimum number of pagelets that must be in the free page list before a process can borrow pages in WSINC amounts up to WSEXTENT pages.
PFRATH (Page Fault Rate High)	Indicates the number of page faults that must occur within the period specified by AWSTIME, before OpenVMS makes an "automatic working set adjustment" to increase the working set by WSINC.
PFRATL (Page Fault Rate Low)	If the number of page faults is less than this value, OpenVMS makes an "automatic working set adjustment" to decrease the working set by WSDEC, but not below the point set by AWSMIN.

Note: While parameters have units of pagelets, OpenVMS allocates memory in pages. For this reason, the parameter values you choose should be even multiples of the number of pagelets per page. OpenVMS automatically rounds these parameters if you do not follow this guideline.

A.3.1 Default Memory Allocation

The following steps summarize the algorithm OpenVMS uses to determine the amount of physical memory available to each process. Each capitalized term represents a system parameter that OpenVMS uses in allocating physical memory.

1. The process begins executing an image. OpenVMS allocates to it the number of pagelets specified by the quota parameter *WSDEFAULT*.

```
Page Allocation:
  0--->WSDEFAULT
```

2. If the process has a higher page fault rate than *PFRATH*, OpenVMS allocates more pagelets to it. OpenVMS allocates these pagelets in increments of *WSINC* until the process has *WSQUOTA* total pagelets.

```
Page Allocation:
  0--->WSDEFAULT-->WSINC-->WSINC-->WSQUOTA
```

3. If the process continues to have a high page fault rate and the free list size exceeds *BORROWLIM*, OpenVMS will continue to allocate pagelets in *WSINC* blocks until the free list size is inadequate or *WSEXTENT* is reached.

```
Page Allocation:
  0-->WSDEFAULT-->WSINC-->WSQUOTA-->WSINC-->WSEXTENT
```

4. If the process's page fault rate drops below *PFRATL*, OpenVMS removes pages from the working set in increments of *WSDEC* until the page fault rate exceeds *PFRATL*.
5. When the process exits the image, it loses these additional pages. For example, consider a monthly batch payroll job. When you run it in July, it begins execution at *WSDEFAULT*, and gains pages until it reaches *WSEXTENT*. When you run it in August, it again begins execution at the default value of *WSDEFAULT*.

See your OpenVMS documentation for more on the OpenVMS page allocation scheme.

A.3.2 Process Control of Memory Allocation

OpenVMS provides several facilities to alter working set parameters on a process-by-process or user-by-user basis. The table below summarizes these methods:

Working Set Control

Type of Process	Method of Working Set Control
Users	The **AUTHORIZE** utility lets you change the default working set on a user-by-user basis.
Interactive Processes	The DCL command **SET WORKING_SET** allows interactive processes to change working set parameters.
Batch Processes	The **SUBMIT** command qualifiers for working sets allow batch processes to alter physical memory allocations.
Batch Queues	You can use the DCL command **INITIALIZE/QUEUE** to alter working set parameters for batch queues.

A.3.3 Keeping Memory Free for New Processes

Memory is finite. Every time a new process begins under OpenVMS, it takes *WSDEFAULT* pages away from the total number of available pages. The remaining memory is available for the free page list and for *WSINC* additions to each process's allocation.

Heavily loaded systems can run so many concurrent processes that the number of pages remaining for use in the free page list becomes very small. OpenVMS includes a parameter called *FREELIM* that sets a lower limit for the number of pages in the free page list.

The size of the free and modified page lists directly affects the ratio of hard page faults to soft page faults. Hard page faults cause a process to experience a resource wait state, resulting in slow execution. For better performance, it is important to tune the system to minimize hard

page faults. While soft faults are less expensive, an excessive soft page fault rate can also lead to poor performance.

Whenever OpenVMS detects that the size of the free page list falls below *FREELIM*, OpenVMS starts a mechanism to bring the free list size up to *FREEGOAL*. To achieve this, if all other mechanisms are inadequate, OpenVMS may move the entire working set of an inactive process out to disk. This procedure is called *swapping*. The process placed on disk is swapped out. When the process is brought back into memory, it is swapped in. Swapping has a detrimental effect on overall system performance.

A.3.4 Allocation for Shared Sections

OpenVMS uses a slightly different approach for allocating memory for shared code and data. The *GBLSECTIONS* parameter sets the number of shared sections to be allocated when the system is started. The *GBLPAGES* parameter sets the number of global page table entries. Every group of 128 page entries requires 4 bytes of resident memory. In addition, OpenVMS uses the *GBLPAGFIL* parameter to set the maximum number of page file blocks available for global pages.

A.4 How Caché Uses OpenVMS Memory

Caché uses both shared memory and memory private to each process when running on OpenVMS.

A.4.1 Balance Memory Locking and Paging

All Caché code is shared, and can be physically locked in memory. The same is true for the global buffer pool.

Routine buffers work similarly, though with a slight difference: routine buffers are shared, but you can specify how much of each routine buffer is locked into memory. If a routine is larger than the locked portion of a routine buffer, only a portion of the routine in the routine buffer is locked in memory. The remainder of the routine is loaded into unlocked physical memory.

Locking shared data and routines in memory allows better response time since memory access is quicker than disk access. As a result, the more globals and routines that are kept in memory, the better. Memory, however, is a finite resource. The more global and routine buffers that

are allocated, the less memory is available for OpenVMS processes. When less memory is available, more OpenVMS paging occurs.

The goal, therefore, is to choose a number of global and routine buffers that lets you keep enough globals and routines in memory without negatively affecting OpenVMS paging. This document provides guidelines for selecting these values.

A.4.2 Use Process-Private Space to Reduce Paging

Every Caché process maps to the shared memory sections, but it also has access to a private area of memory called process-private space. This private area includes variables, arrays, stacks, and other data structures that belong to a particular process.

A portion of this private area of memory may be locked into the process's working set to reduce paging. None of the private area, however, is physically locked in memory.

A.5 Calculating OpenVMS Parameters

The accompanying OpenVMS Parameter Calculator offers good starting values for both Caché and OpenVMS parameters. Later, if your system is not functioning as well as you would like, you can adjust these parameters to achieve optimal performance. The following sections describe the process of determining OpenVMS parameters:

- Determine Parameter Calculator Input Values
- Record Current OpenVMS Parameter Values
- Use the Caché OpenVMS Parameter Calculator
- Analyze the Calculation Results

You may want to print out the tables that follow so that you can record the relevant values.

A.5.1 Determine Parameter Calculator Input Values

This section helps you find appropriate values for the fields that control the number of processes and the number of global and routine buffers used by Caché. You need to determine these values, so that you can use the Parameter Calculator to compute both Caché and OpenVMS parameters accurately.

During installation you can set these and other Caché parameters, or you can retain default values. If you find later that your system needs tuning, you can adjust the values at any time using the System Management Portal. You need to restart Caché for most parameter changes to take effect.

As you determine the values for these fields, also record them in the *Input to OpenVMS Parameter Calculator Table*:

- Determine Number of Processes
- Determine Size of Routine Cache
- Determine Number of Global Buffers Per Process

Important: A bug exists in the OpenVMS operating system when attempting to allocate resident global buffers if insufficient space is available. To prevent Caché from hanging at startup, it is best to calculate the specific memory needs of your system and manually enter these values on the **[Home] > [Configuration] > [Memory and Startup]** page of the System Management Portal or in the configuration (cache.cpf) file itself.

A.5.1.1 Determine Number of Processes

Before you can determine how many global and routine buffers Caché should allocate at startup, you must first determine the maximum number of Caché processes that run on your system at one time.

Every user in Caché constitutes one Caché process. Every Caché **Job** command creates a process. You should consider any process that appears on the %SS system status display, except the system processes **Garbage Collector** and **Write daemon**. As a rule of thumb, use the number of processes called for in your Caché license for this value; it cannot exceed your license limit.

Enter the value in the **Number of Processes** field in the parameter calculator.

A.5.1.2 Determine Size of Routine Cache

Each routine buffer holds one and only one routine at a time. An OpenVMS system can have up to 524 MB of routine cache.

Many processes can share a routine buffer. Thus, when a process switches from one Caché routine to another, it merely accesses the other routine in another buffer. When no process is currently accessing a buffer, the buffer is returned to the pool of free buffers, and a new routine

can be loaded into the empty buffer as it is required by another Caché process. If all buffers are occupied when Caché needs to load a routine, it chooses the least recently used (LRU) buffer, not the least frequently used (LFU) buffer. Allowing more buffers enhances performance.

The nature of the application influences the appropriate number of buffers. If many users access a small number of routines, a relatively small number of buffers will suffice for optimum response time. Conversely, an application with a large number of routines will benefit from a large number of buffers.

The parameter calculator starts with a default value for the number of routine buffers. This default is based upon a medium-sized site. You can enter a different value for this parameter in the **Routine Buffer Pool** field in the parameter calculator based on the specifics of your site.

Note: If you prefer to enter a value in this field with an MB unit of measure, this value is used for the **Memory for Routine Cache (MB)** field in the **Calculation Result** section and also in calculations involving routine buffers; it is not recalculated.

A.5.1.3 Determine Number of Global Buffers Per Process

The parameter calculator assumes the global buffers are 8 KB in size. The calculator starts with recommended "rule of thumb" values for clustered and nonclustered systems.

If your system is memory-poor, you can reduce this value. Do not, however, decrease it below a multiple of 32 KB per process.

Using more global buffers helps the performance of most sites. You can use the statistics produced by the **^GLOSTAT** utility to determine if adding more global buffers will reduce disk access and thereby improve performance.

A.5.2 Record Current OpenVMS Parameter Values

Every system has memory needs other than those for Caché. For example, there are memory requirements for FORTRAN and COBOL layered products as well as for OpenVMS itself. To analyze these requirements for your system, follow the procedures below to display the values of the indicated parameters prior to installing Caché.

Note: If you are performing an upgrade, shut down Caché to get accurate numbers.

1. Run the OpenVMS **SYSGEN** utility to display the current value of each parameter listed in the *OpenVMS System Parameter Values Table*.

```
$ SET DEF SYS$SYSTEM
$ RUN SYSGEN
SYSGEN> USE CURRENT
SYSGEN> SHOW parameter-name
```

2. Run the OpenVMS **SYSGEN** utility to display the current value of the process parameters listed in the *OpenVMS Process Parameter Values Table* below:

   ```
   $ SET DEF SYS$SYSTEM
   $ RUN SYSGEN
   SYSGEN> USE CURRENT
   SYSGEN> SHOW/PQL
   ```

   Alternatively, you may run the OpenVMS **AUTHORIZE** utility to display the current value of the user authorization file (UAF) parameters for each user account that uses Caché:

   ```
   $ SET DEF SYS$SYSTEM
   $ RUN AUTHORIZE
   UAF> show user-name
   ```

   You must then set the UAF records for all of the user accounts individually once the calculator returns the minimum recommendations for these values.

A.5.3 Use the Caché OpenVMS Parameter Calculator

This section describes how to use the Caché OpenVMS Parameter Calculator. This tool is an interactive HTML document that provides up-to-date calculations for the OpenVMS parameters required by Caché. The default values provide reasonable numbers for a medium sized installation. InterSystems experts are available to provide additional help configuring and tuning Caché for your site.

Perform the following steps to retrieve parameter values:

1. Choose whether you are calculating parameters for a clustered system. If you select **Yes**, you receive both the cluster master values and values for any additional nodes; otherwise, the tool calculates values for a single node.
2. As input to the OpenVMS Parameter Calculator use the values you determined in the previous section for the following fields:

Input to OpenVMS Parameter Calculator

Parameter	User-supplied value
Number of Processes	
Routine Buffer Pool	
Global Buffers per Process	

3. Access the Caché OpenVMS Parameter Calculator to calculate the necessary parameter values.
4. Fill in the values returned by the calculator in the appropriate tables in the following section.

A.6 Analyze the Calculation Results

The results from the calculator are suggested values; treat them as a guideline. You may need to adjust settings in order to balance and maximize system performance, depending on your site's hardware and software configuration.

A.6.1 Update Caché Parameters

The following values are for Caché parameters that you can update during the installation or by using the System Management Portal from a remote Web server after you install Caché.

Output Caché Parameters

Parameter	Calculated value
Number of Global Buffers	
Memory for Database Cache (MB)	
Memory for Routine Cache (MB)	

Update the memory cache settings using the **[Home] > [Configuration] > [Memory and Startup]** page of the portal. Click **Manually** for the **Configure Memory Settings** option, so that you can enter the values that are best for your system.

A.6.2 Update OpenVMS System Parameters

The system parameter values computed by the OpenVMS Parameter Calculator are the amounts you need to add to your current OpenVMS system in order to make it ready to run Caché. Caché will consume this much memory, so evaluate your system to make sure that there is the proper amount of memory available.

Use this table to help you combine the required amounts returned by the calculator with your current system values. Update these parameters by using the **SET** command of the OpenVMS **SYSGEN** utility.

OpenVMS System Parameter Values

Parameter	Current value	Required Caché amount	Resulting value
BALSETCNT			
MAXPROCESSCNT			
GBLPAGES			
GBLPAGFIL			
GBLSECTIONS			
LOCKIDTBL			
RESHASHTBL			
SYSMWCNT			

A.6.3 Update Special OpenVMS Parameters

The parameter calculator provides a few special parameters used to configure Caché on your OpenVMS system:

- The *Resident Memory* value is necessary if you want to create a named Resident Memory Section for Caché.
- The values for the special parameters *PRIORITY_OFFSET* and *PE1* are fixed values recommended by InterSystems; set these parameters to the recommended values.

Verify the values returned by the calculator match those of your OpenVMS system.

Special OpenVMS Parameter Values

Parameter	Returned value
Resident Memory (MB)	
PRIORITY_OFFSET	
PE1	

A.6.4 Update OpenVMS Process Parameters

InterSystems recommends minimum values for some PQL process quota parameters. This is because Caché maintains a table, *GJOBQ*, which holds default values for many of these parameters. You must have authorized these values in order for Caché processes to be able to use the values in the *GJOBQ* table.

You do not need to calculate values for these parameters. The minimum suggested values appear in the OpenVMS Parameter Calculator. Enter these values in the "Recommended minimum value" column in the table below.

OpenVMS Process Parameter Values

Parameter	Current value	Recommended minimum value
ASTLM		
BIOLM		
BYTLM		
DIOLM		
FILLM		
PGFLQUOTA		
TQELM		
WSQUOTA		
WSEXTENT		
ENQLM		

If your current value for any of these parameters is less than the recommended minimum value, you should adjust the values during installation using the **SET** command of the OpenVMS **SYSGEN** utility or the OpenVMS **AUTHORIZE** utility for each user account.

A.6.5 Resident Memory Section

The OpenVMS platform makes use of the Resident Memory Section facility. All OpenVMS Caché users are encouraged to make use of this facility.

Systems running OpenVMS v7.1 and later have a memory mechanism for allocating global sections (memory which can be shared between processes). The two features that Caché supports, memory resident global sections and shared page tables, are always used as a pair.

The advantages of using a memory-resident global section mapped via a shared page table for the global buffer pool are:

- Access to the pages in a memory-resident global section is not charged against the process's pagefile quota nor the working set quota. Also, when a process maps to a memory-resident global section and references the pages, it does not use the process's working set list, so process quotas may often be reduced.
- Shared page tables enable two or more processes to map to the same physical pages without each process incurring the overhead of page table construction, page file accounting, and working set quota accounting. Internally, shared page tables are treated as a special type of global section and are specifically used to map pages that are part of a memory-resident global section.
- There is only one copy of the global buffer pool page table on the system, which conserves physical memory and speeds up mapping the global section into a new process. Shared page tables dramatically reduce the database server startup time because server processes can map memory-resident global sections hundreds of times faster than traditional global sections. This increases overall system capacity and decreases response time to client requests.

The drawback to using these structures is that if you want to increase the size of your global buffer pool, you may have to restart your OpenVMS system to reconfigure the amount of space reserved for the memory-resident global section. You can reserve this space when the system starts (via **SYSMAN** and **AUTOGEN**) or you can allocate it dynamically. If you do not reserve it, when Caché requests it, there may be insufficient space available and the request fails. If Caché attempts to use a memory-resident global section and it cannot, it allocates the global buffer pool out of a non–memory-resident global section.

Create a Named Resident Memory Section

To use these structures you need to do the following:

1. Use the value returned by the OpenVMS Parameter Calculator for *Resident Memory* as the reserved memory size.

2. Use **SYSMAN** to reserve a section of memory. You must supply a name for the named section. You can use any name that consists of alphanumeric characters (and the underscore) and is not longer than 43 characters. The syntax for **SYSMAN** is:

```
MCR SYSMAN ! run sysman
SYSMAN> RESERVED_MEMORY ADD "Resident_Memory_Name" -
       /ALLOCATE/PAGE_TABLES/ZERO/SIZE=<size in MB>
SYSMAN> EXIT
```

3. Run **AUTOGEN** to process the Reserved Memory Registry data file which contains the information to establish your memory-resident global section. After the system is restarted, the **SYSMAN** command displays the shared memory section you reserved:

```
MCR SYSMAN ! run sysman
SYSMAN> RESERVED_MEMORY SHOW
```

4. Modify the Caché configuration files (config.def, cache.cpf, etc.) to use the reserved section. This is done by setting the reserved memory parameter in the [config] section so that it reads:

```
useresidentmem=<resident memory section name>
```

   The *<resident memory section name>* is the name of the section you specified in **SYSMAN** when you reserved it (“Resident_Memory_Name” in the example above). If you name this section, it is required in the configuration file.

   Alternatively, you can also modify the configuration files by using the System Management Portal that connects to your OpenVMS Caché instance from a remote Web server:

   a. Navigate to the **[Home] > [Configuration] > [Advanced Settings]** page and click **Memory** in the **Category** box or enter `resident` in the **Filter** box to shorten the list.

   b. In the **ResidentMemorySectionName** setting row, click **Edit** to display the **Configuration Setting** page.

   c. Enter the name you reserved in the **SYSMAN** utility (“Resident_Memory_Name” in the example above) in the **Value** box and click **OK**.

5. Restart Caché. If it cannot use the reserved memory section, Caché displays a message and stores an error code in the SYSLOG.

If you use resident memory to map your global buffer pool, you may be able to reduce some of the system parameters, in particular, *WSMAX*. Typically OpenVMS comes with a *WSMAX* default value much larger than is required for running Caché.

Resident Memory Section Name

On OpenVMS systems, when using a resident memory section for the global buffer pool, you have two options:

1. Indicate the desired name of the section in the *residentmem* parameter of the cache.cpf configuration file. This is useful to reserve physical memory using the **SYSMAN** utility, which now requires entry of the section name.
2. Let the system choose a name for the section. In previous versions of Caché, the system used the name *ISC_Shared_Memory*, but since this was a fixed name, it was not compatible with multiple configurations. Caché automatically creates a name based on the manager's directory to support multiple installations.

If you supply the section name in a certain configuration, this name is used instead of the system-generated one. InterSystems encourages the use of the **SYSMAN** utility to reserve a resident memory section in production environments to utilize shared page tables.

B

Calculating System Parameters for UNIX and Linux

This document is part of the *Caché Installation Guide for UNIX and Linux*. This document explains how you can calculate the best parameters for your system. It is divided into two sections:

- Determine Memory and Disk Requirements—calculate memory requirements, swap space, disk requirements, maximum buffers, maximum users, and maximum database size.
- Configure UNIX Kernel Parameters—set values for tunable UNIX parameters and other platform-specific memory management issues.

For optimal Caché performance, you need to calculate proper values for certain Caché system parameters. These values allow you to determine whether you need to adjust certain system level parameters. The values you choose should minimize swapping and paging that require disk accesses, and thus improve system performance.

Review this section carefully and calculate the proper values for both your operating system and Caché before proceeding. Use the tables provided here to record the current and calculated values for your system level parameters. You can then refer to these tables when you install Caché. After your system is running, you may need to adjust these values to gain optimal performance.

If you are not already familiar with the memory organization at your operating system level, consult the appropriate system documentation.

B.1 Determine Memory and Disk Requirements

This section outlines the basic system requirements for most systems. Because these requirements vary by platform, consult your platform documentation for additional information.

B.1.1 Calculate Memory Requirements

Use the breakdown of memory usage shown in the following table to calculate the memory your system needs for Caché.

UNIX Memory Requirements

Components	Memory Requirements
Operating system	1800 KB (operating system dependent)
Caché	842 KB
Global database cache	8 KB per buffer
Routine cache	32 KB per routine buffer
User overhead	1024 KB per process
Network (if present)	300 KB per port for each network system process (DMNNET, DCP, and RECEIVE). Caché ports have two DMNNET system processes per port. In addition, there is a network shared memory requirement, which depends on the number of ports and the number of remote hosts configured. For a basic system, this requirement is about 304 KB.

By default, Caché automatically allocates shared memory, including routine buffers and global buffers, to a total of one-eighth of the system available shared memory space. If you plan to run large applications or support large numbers of users, tune the system according to the following formula:

```
   (number of routine buffers)*32 KB
+    (number of global buffers)*8 KB
+                               4 MB
_____________________________________
=               Shared memory needed
```

B.1.2 Calculate Swap Space

The amount of swap space available on your system should never be less than the amount of real memory plus 256 KB.

With this minimum in mind, InterSystems recommends the following value as the minimum amount of swap space needed for Caché:

```
  ((# of processes + 4)† * (1024 KB)‡
+           total global buffer space
+          total routine buffer space
_____________________________________
=                   Minimum swap space
```

† You add 4 to the # of processes for the Caché Control Process, the Write daemon, the Garbage Collector, and the Journal daemon. You should also add 1 for each slave Write daemon. The # of processes must include all user and jobbed processes which might run concurrently. If you are running networking, add 1 for the RECEIVE system process plus the number of DMNNET daemons you have running (2 per port). If your system is a server, add the number of DCP processes that will be created when all potential clients access the server.

‡ The 1024 KB number is approximate. It is based on the current size of the Caché executable and grows with the partition size you allocate to each Caché process. On most systems, provide only as much swap space as necessary. However, some systems require you to provide swap space for the worst case. Under these conditions, you need to increase this number to as high as 1.5 MB, depending on the partition size you specify.

Be sure to confirm that your UNIX system permits the amount of swap space you require. For specific information about swap space on your system, consult your UNIX operating system manual.

B.1.2.1 Solaris Swap Space

To calculate swap space for the Solaris 9 platform:

```
swap -l
```

Example:

```
>swap -l
swapfile                  dev     swaplo blocks   free
/dev/dsk/c0t2d0s0       136,0       16    526304   526304
/dev/dsk/c0t2d0s1       136,1       16   2101184 2101184
```

B.1.2.2 Tru64 UNIX Swap Space

To display swap space for Tru64 UNIX:

```
/usr/sbin/swapon -s
```

Information for each swap partition is displayed similar to the following example:

```
Swap partition /dev/disk/dsk1b (default swap):
         Allocated space:        16384 pages (128MB)
         In-use space:           10452 pages ( 63%)
         Free space:              5932 pages ( 36%)

Swap partition /dev/disk/dsk4c:
         Allocated space:       128178 pages (1001MB)
         In-use space:           10242 pages (  7%)
         Free space:            117936 pages ( 92%)

Total swap allocation:
         Allocated space:       144562 pages (1.10GB)
         Reserved space:         34253 pages ( 23%)
         In-use space:           20694 pages ( 14%)
         Available space:       110309 pages ( 76%)
```

B.1.2.3 AIX Swap Space

To display swap space for AIX:

```
lsps -a
Page Space  Physical Volume   Volume Group    Size %Used
Active  Auto  Type
hd6         hdisk2            rootvg                  512MB      72
yes   yes    lv
```

B.1.2.4 HP-UX Swap Space

To display swap space for HP-UX:

```
swapinfo (3M)
# /usr/sbin/swapinfo
             Kb      Kb      Kb   PCT  START/       Kb
TYPE      AVAIL    USED    FREE  USED   LIMIT RESERVE  PRI  NAME
dev      524288  138260  386028   26%       0       -    1  /dev/vg00/lvol2
reserve       -   78472  -78472
memory   195132  191668    3464   98%
```

B.1.3 Calculate Disk Requirements

In addition to the swap space you just calculated, you need disk space for the following items:

- 67 MB for Caché.
- 3 MB for the Caché Server Pages (CSP).
- 3.5 MB for Caché ODBC support.

- 2.5 MB for the Caché manager sources.
- 6.6 MB for the Caché engine link libraries.
- 3 MB for WebLink.
- Space for your Caché application database.
- Approximately 12.5% of the buffer pool size for the initial size of the write image journal file. If your disk does not have enough space for the write image journal file, when you start Caché it displays a message indicating that the system did not start.
- Desired space for journal files.

Although you do not need to remove any installation files after completing the installation procedure, you can do so if you are short on disk space. The installation program tells you how much space can be saved, and asks if you want to delete the installation files.

B.1.4 Determine the Number of Global Buffers

Caché supports the following maximum values for the number of global buffers:

- For 32-bit platforms, any combination of 2-KB and 8-KB buffers that are:
  - Less than 1 GB for HP-UX
  - Less than 2 GB for other 32-bit platforms

  The 2-GB value is the total address space the operation system allocates for the process data, which includes not only shared memory, but other Caché and operating system data as well. Therefore, it represents an upper limit that is not achievable in practice.
- For 64-bit platforms: Tru64 UNIX, Solaris/UltraSPARC, HP-UX/HP-PA64, and AIX/PowerPC64

  The number of global buffers is limited only by the operating system and the available memory.

Note: Even if your configuration is using only 2-KB databases, some of the Caché databases, namely CACHESYS and CACHETEMP, are 8-KB. The system always sets aside a minimum of 200 8-KB buffers for these, regardless of what you specify. In general, increase this value, especially if you are making heavy use of Caché SQL.

Set your values to less than the maximum number of buffers listed above.

B.1.5 Determine the Number of Routine Buffers

Caché supports the following maximum value for the number of routine buffers:

65,535

Set your values to less than the maximum number of buffers listed above.

B.1.6 Determine the Maximum Number of Users

The maximum users allowed by Caché is the *lowest* of the following values:

- The license limit
- # of routine buffers - 1
- # of semaphores - 4

B.1.7 Determine the Maximum Database Size

The *ulimit* parameter in UNIX determines the maximum file size available to a process. For the Caché Manager group, the value of *ulimit* should either be unlimited or as large as the largest database the user may have.

B.2 Configure UNIX Kernel Parameters

The following sections describe issues related to tuning and performance on various UNIX platforms:

- Set Values for Tunable UNIX Parameters
- Platform Configuration Issues

B.2.1 Set Values for Tunable UNIX Parameters

Caché uses a configurable number of semaphores, in sets whose size you define. The parameters *SEMMNI*, *SEMMNS*, and *SEMMSL* reflect the number of semaphores per set and the total number of semaphores Caché uses. The UNIX/Linux parameters that govern shared memory allocation are *SHMMAX*, *SHMMNI*, *SHMSEG*, and *SHMALL*. Caché uses shared memory and allocates one segment of shared memory; the size of this segment depends on

the area set aside for global buffers and routine buffers. It uses the following formula to determine the segment's minimum size:

```
  space required for routine buffers
+  space required for global buffers
+                                4 MB
_____________________________________
=          Shared memory segment size
```

If you are distributing your data across multiple computers, Caché allocates a second segment; by default, there is no memory allocated for the second segment. (If you plan to use distributed data, contact your VAR or InterSystems support for configuration guidelines.) You can alter *NBUF* and *NHBUF* according to other system requirements. Because Caché does all its own disk buffering, you should keep *NBUF* and *NHBUF* small. The following table lists the most common names of the UNIX parameters that you may need to change, the value InterSystems recommends for each parameter, and a brief description of each. Verify that your parameter values are set to at least the minimum value. Certain parameters may not be implemented on all platforms or may be referred to differently. Refer to platform-specific tuning notes for more information.

Tunable UNIX Parameters

Kernel Parameter	Minimum Recommended Value	Definition
CDLIMIT	Number of bytes in largest virtual volume	Maximum size of a file.
MSGMAX	2	Maximum message size, in bytes.
MSGMNI	Number of Caché instances x 2; each Caché instance uses two message queues	Minimum number of uniquely identifiable message queues that may exist simultaneously.
NOFILES	35	Number of open files per process.
SEMMNI	Product of SEMMNI and SEMMSL must be greater than the # of user processes + 4	Number of semaphore identifiers in the kernel; this is the number of unique semaphore sets that can be active at any one time.
SEMMNS	128 or ...	Total number of semaphores in the system. User processes include jobbed processes and all other semaphores required by other software.
	Number of processes expected to run. If the process table might expand, use a larger number to provide for expansion.	

Kernel Parameter	Minimum Recommended Value	Definition
SEMMSL	See SEMMNI	Maximum number of semaphores per identifier list.
SHMALL	60 KB or ...	Maximum total shared memory system-wide. Units should be in KB. 1000 represents the MCOMMON shared region.
	1000 + total global buffer space+ total routine buffer space *	
SHMMNI	3	Maximum number of shared memory identifiers system-wide.
SHMSEG	3	Number of attached shared memory segments per process.
SHMMAX	60 KB or ...	Maximum shared memory segment size in KB.
	1000 + total global buffer space+ total routine buffer space	

* This is the minimum value for *SHMALL* required for Caché UNIX. You must also take into account any other applications that use shared memory. If you are unsure of other shared memory use, calculate *SHMALL* as *SHMSEG* multiplied by *SHMMAX*, in pages; this larger value will suffice in all cases.

Important: Enough swap space must be created to support the memory allocated, unless the operating system documentation explicitly state otherwise. On certain operating systems (Solaris, for example) Caché creates "locked shared memory segments" , which are not pageable. However these memory segments still need swap space unless the operating system documentation states otherwise.

B.2.2 Platform Configuration Issues

The following sections contain configuration issues for some individual platforms. For more information, consult the system documentation for your platform.

- Tru64 UNIX Platform Notes
- Sun Solaris Platform Notes
- AIX Platform Notes
- HP-UX Platform Notes

- Red Hat Linux Platform Notes
- SUSE Linux Platform Notes

B.2.2.1 Tru64 UNIX Platform Notes

For the Tru64 UNIX supported releases, set the following kernel parameters to the size of all the shared memory to allocate (including space allocated for the buffer pool, routine buffers, and other data structures):

- *shm_max*
- *per_proc_address_space*
- *max_per_proc_address_space*

The following parameters are not implemented:

- *MSGMAX*
- *MSGMNI*
- *NCALL*
- *SEMMNI*
- *SEMMNS*
- *SEMMSL*
- *SHMALL*
- *SHMMNI*

SMMAX and *SMSEG* are referred to as *SHMMAX* and *SHMSEG*, respectively.

The **`sysconfig -q subsystem [attribute]`** command displays the current values for the attributes of the specified subsystem, or the particular attribute specified. See the Tru64 UNIX documentation entry for sysconfig(8) for more information.

For example:

```
>sysconfig -q ipc
msg_max = 8192
msg_mnb = 16384
msg_mni = 64
msg_tql = 40
shm_max = 268435456
shm_min = 1
shm_mni = 128
shm_seg = 32
sem_mni = 16
sem_msl = 1000
sem_opm = 10
sem_ume = 10
sem_vmx = 32767
sem_aem = 16384
max_kernel_ports = 56864
ssm_threshold = 8388608
ssm_enable_core_dump = 1
shm_allocate_striped = 1
shm_enable_core_dump = 1
```

To permanently modify the value of an attribute, use the ***sysconfigdb -a -f stanza_file subsystem*** command to specify the stanza-formatted file that contains the subsystem, the attribute, and the new permanent attribute value. The subsystem argument specifies the subsystem whose attribute you want to modify.

The following is an example of a stanza-formatted file that changes the permanent values of the ipc subsystem attributes *shm_max* and *sem_msl*:

```
ipc:
  shm_max = 268435456
  sem_msl = 1000
```

See stanza(4) and sysconfigdb(8) for information about stanza-formatted files.

To use the new permanent value, restart the system or, if the attribute can be tuned at run time, use the **sysconfig -r** command to change the current value (see Section 3.6.4 of the Tru64 UNIX System Configuration and Tuning guide).

Setting File And Directory Ownership

For the Tru64 UNIX supported releases, unlike other UNIX file systems, group ownership does not come from the group ID of the creating process. Instead, the group ID of the file is set to the group ID of its parent directory.

However, when the vfs subsystem attribute *sys_v_mode* is set to 1, the group ID of the file is set either to the group ID of the process or, if the S_ISGID bit of the parent directory is set, to the group ID of the parent directory If the group ID of the new file does not match the effective group of the process or one of its supplementary group IDs, the S_ISGID bit of the new file is cleared. In general, this does not present a problem since the group ID of all directories created by Caché utilities is properly set to the correct group owner.

There are, however, circumstances which can cause problems. For example, if an administrator uses the **^DATABASE** utility to create a database in an nonexistent directory, the utility creates the directory, but it does not adjust the group ID of the newly-created directory, which is inherited from the parent directory. As a result, the database, with its group ID inherited from the directory, may be inaccessible to `cacheusr`. Other Caché utilities (the journal and shadow processes, for example) that create directories have the same problem.

Note: InterSystems recommends that you set the *sys_v_mode* to 1 on all file systems and directories that Caché uses to insure smooth functioning of the system. See open(2) for more information.

B.2.2.2 Sun Solaris Platform Notes

The notes for the Sun Solaris platform differ depending on what version you are running:

- Sun Solaris Release 9
- Sun Solaris Release 10

Sun Solaris Release 9

The following are notes to keep in mind when configuring and tuning Caché on the Sun Solaris version 9 platform:

1. The following parameters are not implemented:*CDLIMIT*, *NOCALL*, and *SHMALL*.
2. The *NOFILES* parameter is unlimited.
3. Edit the /etc/system file and add the following variables to increase shared memory segments:

```
set shmsys:shminfo_shmmax=value
set shmsys:shminfo_shmmin=value
set shmsys:shminfo_shmmni=value
set shmsys:shminfo_shmseg=value
set semsys:seminfo_semmap=value
set semsys:seminfo_semmni=value
set semsys:seminfo_semmns=value
set semsys:seminfo_semmsl=value
set semsys:seminfo_semmnu=value
set semsys:seminfo_semume=value
```

Solaris Shared Memory Variables

Variable	Description
shmsys:shminfo_shmmax	Maximum shared memory segment size
shmsys:shminfo_shmmin	Minimum shared memory segment size
shmsys:shminfo_shmmni	Number of shared memory identifiers
shmsys:shminfo_shmseg	Number of segments, per process
semsys:seminfo_semmap	Number of entries in the semaphore map
semsys:seminfo_semmni	Number of semaphore identifiers
semsys:seminfo_semmns	Number of semaphores in the system
semsys:seminfo_semmsl	Maximum number of semaphores, per ID
semsys:seminfo_semmnu	Number of processes using the undo facility
semsys:seminfo_semume	Maximum number of undo structures per process

4. Verify the shared memory value changes:

```
# grep shmsys /etc/system
```

5. Restart the system:

```
# init 6
```

The following sample shared memory values accommodate a system with an average amount of memory (512 MB) that is running a average database application:

```
set shmsys:shminfo_shmmax=268435456
set shmsys:shminfo_shmmin=200
set shmsys:shminfo_shmmni=200
set shmsys:shminfo_shmseg=200
set semsys:seminfo_semmap=250
set semsys:seminfo_semmni=500
set semsys:seminfo_semmns=500
set semsys:seminfo_semmsl=500
set semsys:seminfo_semmnu=500
set semsys:seminfo_semume=100
```

Sun Solaris Release 10

The Solaris 10 release no longer uses the /etc/system mechanism to tune the IPC shared memory parameters. These allocations are now automatic or configured through the resource controls mechanism.

If you try to use /etc/system on Solaris 10, you may receive the following message:

```
* IPC Shared Memory
*
* The IPC Shared Memory module no longer has system-wide limits.
* Please see the "Solaris Tunable Parameters Reference Manual" for
* information on how the old limits map to resource controls and
* the prctl(1) and getrctl(2) manual pages for information on
* observing the new limits.
```

See "Chapter 6 Resource Controls (Overview)" of the *System Administration Guide: Solaris Containers-Resource Management and Solaris Zones* on the Sun Web site for detailed information on using the **rctladm**, **prctl**, and **projects** commands to set Solaris 10 parameters.

B.2.2.3 AIX Platform Notes

This topic include the following categories of note:

- AIX Default Parameters
- AIX Tunable Parameters

AIX Default Parameters

The default settings of several AIX parameters can adversely affect performance. The settings and recommendations are detailed below for the following:

- I/O Pacing Parameters
- File System Mount Option
- Memory Management Parameters

I/O Pacing Parameters

The default values for the disk-I/O pacing high-water and low-water marks (*maxpout* and *minpout* parameters) may cause severe performance problems on Caché production systems. These default values may significantly hinder Caché Write daemon performance by inappropriately putting the Write daemon to sleep causing prolonged Write daemon cycles.

If you are using HACMP clusters, I/O pacing is automatically enabled. If your system is not part of an HACMP cluster, set both the high- (*maxpout*) and low- (*minpout*) water marks to 0 (zero) to disable I/O pacing.

View and change the current settings for the I/O pacing high-water and low-water marks by issuing the **smitty chgsys** command.

InterSystems currently recommends the following IBM calculation for determining the appropriate high-water mark:

```
high-water mark = (4 * n) + 1
```

Where *n* = the maximum number of spindles any one file (database, journal, or WIJ) spans across. Set the low-water mark to 50%-75% of the high-water mark.

For example, a CACHE.DAT database file is stored on a storage array, and the LUN (or file system) where it resides consists of 16 spindles/drives. Calculate:

```
High-water mark = (4 * 16) + 1 = 65

Low-water mark = between (.50 * 65) and (.75 * 65) = between 33 and 49
```

For more details, see the "Using Disk-I/O Pacing" section of the *AIX Performance Management Guide* at the following IBM Web page:

http://publib16.boulder.ibm.com/pseries/en_US/aixbman/prftungd/diskperf6.htm

Important: InterSystems recommends the current IBM calculation for determining the appropriate high-water mark; verify the preceding calculation with IBM, as it may change.

If you have any questions about the impact to your system, contact the InterSystems WRC Performance Team or your AIX supplier before making any changes. These recommendations are independent of Caché versions and apply to both JFS and Enhanced JFS (JFS2) file systems.

File System Mount Option

For optimal performance, place all CACHE.DAT, CACHE.WIJ, and Caché journal files on a file system with release-behind-when-reading and release-behind-when-writing capabilities by using the `rbrw` mount option.

For more information, see the "mount Command" page of the *AIX Commands Reference, Volume 3, i - m* at:

http://publib16.boulder.ibm.com/doc_link/en_US/a_doc_lib/cmds/aixcmds3/mount.htm

Memory Management Parameters

The number of file systems and the amount of activity on them can limit the number of memory structures available to JFS or JFS2, and delay I/O operations waiting for those memory structures.

To monitor these metrics, issue a **vmstat -vs** command, wait two minutes, and issue another **vmstat -vs** command. The output looks similar to the following:

```
# vmstat -vs
              1310720 memory pages
              1217707 lruable pages
               144217 free pages
                    1 memory pools
               106158 pinned pages
                 80.0 maxpin percentage
                 20.0 minperm percentage
                 80.0 maxperm percentage
                 62.8 numperm percentage
               764830 file pages
                  0.0 compressed percentage
                    0 compressed pages
                 32.1 numclient percentage
                 80.0 maxclient percentage
               392036 client pages
                    0 remote pageouts scheduled
                    0 pending disk I/Os blocked with no pbuf
                 5060 paging space I/Os blocked with no psbuf
              5512714 filesystem I/Os blocked with no fsbuf
               194775 client filesystem I/Os blocked with no fsbuf
                    0 external pager filesystem I/Os blocked with no fsbuf
```

If you see an increase in the following parameters, increase the values for better Caché performance:

- *pending disk I/Os blocked with no pbuf*
- *paging space I/Os blocked with no psbuf*
- *filesystem I/Os blocked with no fsbuf*
- *client filesystem I/Os blocked with no fsbuf*
- *external pager filesystem I/Os blocked with no fsbuf*

When increasing these parameters from the default values:

1. Increase the current value by 50%.
2. Check the **vmstat** output.
3. Run **vmstat** twice, two minutes apart.
4. If the field is still increasing, increase again by the same amount; continue this step until the field stops increasing between **vmstat** reports.

Important: Change both the current and the reboot values, and check the **vmstat** output regularly because I/O patterns may change over time (hours, days, or weeks).

See the following IBM Web pages for more detailed information:

- For a complete description of each of the fields reported by **vmstat**, see the "vmstat Command" page of *AIX Commands Reference, Volume 6, v - z* at:

http://publib16.boulder.ibm.com/doc_link/en_US/a_doc_lib/cmds/aixcmds6/vmstat.htm

- For instructions on how to increase these parameters, see the "VMM page replacement tuning" section of the *AIX Performance Management Guide* at:

  http://publib16.boulder.ibm.com/pseries/en_US/aixbman/prftungd/memperf5.htm

- For a complete description of managing I/O tunable parameters, see the "ioo Command" page of *AIX Commands Reference, Volume 3, i - m* at:

  http://publib16.boulder.ibm.com/doc_link/en_US/a_doc_lib/cmds/aixcmds3/ioo.htm

AIX Tunable Parameters

None of the following listed parameters requires tuning because each is dynamically adjusted as needed by the kernel. See the appropriate AIX operating system documentation for more information.

The following table lists the tunable parameters for the IBM pSeries AIX 5.2 operating system.

AIX Interprocess Communication Tunable Parameters

Parameter	Purpose	Dynamic Values
msgmax	Specifies maximum message size.	Maximum value of 4 MB
msgmnb	Specifies maximum number of bytes on queue.	Maximum value of 4 MB
msgmni	Specifies maximum number of message queue IDs.	Maximum value of 4096
msgmnm	Specifies maximum number of messages per queue.	Maximum value of 524288
semaem	Specifies maximum value for adjustment on exit.	Maximum value of 16384
semmni	Specifies maximum number of semaphore IDs.	Maximum value of 4096
semmsl	Specifies maximum number of semaphores per ID.	Maximum value of 65535
semopm	Specifies maximum number of operations per semop() call.	Maximum value of 1024
semume	Specifies maximum number of undo entries per process.	Maximum value of 1024

Parameter	Purpose	Dynamic Values
semvmx	Specifies maximum value of a semaphore.	Maximum value of 32767
shmmax	Specifies maximum shared memory segment size.	Maximum value of 256 MB for 32-bit processes and 0x80000000u for 64-bit
shmmin	Specifies minimum shared-memory-segment size.	Minimum value of 1
shmmni	Specifies maximum number of shared memory IDs.	Maximum value of 4096

B.2.2.4 HP-UX Platform Notes

For HP-UX release 11i the *CDLIMIT* and *NOFILES* parameters are not implemented. In some versions, *SEMMSL* is hard-coded to 500. *NCALL* is referred to as *NCALLOUT*.

Use the HP *System V IPC Shared-Memory Subsystem* to update parameters. See the HP *System V Inter-Process Communication Mechanisms* online documentation page for additional information. To change a value, perform the following steps:

1. Enter the **/usr/sbin/sam** command to start the System Administration Manager (SAM) program.
2. Double-click the **Kernel Configuration** icon.
3. Double-click the **Configurable Parameters** icon.
4. Double-click the parameter you want to change and enter the new value in the **Formula/Value** field.
5. Click **OK**.
6. Repeat these steps for all of the kernel configuration parameters that you want to change.
7. When you are finished setting all of the kernel configuration parameters, select **Process New Kernel** from the **Action** menu.

The HP-UX operating system automatically reboots after you change the values for the kernel configuration parameters.

B.2.2.5 Red Hat Linux Platform Notes

The default shared memory limit (*shmmax*) on Linux platforms is 32 MB. This value is too small for Caché, but it can be changed in the proc file system without a restart.

If the machine is being used only for Caché, InterSystems recommends setting the shared memory to approximately half the total memory.

For example, to allow 128 MB, type the following command:

```
$ echo 134217728 >/proc/sys/kernel/shmmax
```

You can put this command into a script run at startup.

Alternatively, you can use **sysctl(8)**, if available, to control this parameter. Look for a file called /etc/sysctl.conf and add a line similar to the following:

```
kernel.shmmax = 134217728
```

This file is usually processed at startup, but **sysctl** can also be called explicitly later.

Important: The *msgmni* parameter may also be set too low if you are running more than one instance of Caché on a machine. As stated in the Tunable UNIX Parameters table, set this value to two times the number of instances of Caché that run simultaneously on your system.

Other parameters are sufficiently sized for a Caché application. To view the values of other parameters, look in the files /usr/src/linux/include/asm-xxx/shmparam.h and /usr/src/linux/include/linux/sem.h.

For more information, reference the Managing Kernel Resources chapter of the *Red Hat Database: Administrator and User's Guide*.

B.2.2.6 SUSE Linux Platform Notes

The default shared memory limits (*shhmax* and *shmall*) on SUSE Linux platforms are too small for Caché, and can be changed in the proc file system without a restart.

If the machine is being used only for Caché, InterSystems recommends setting the shared memory to approximately half the total memory.

For example, to allow 512 MB, type the following commands:

```
#sets shmall and shmmax shared memory
echo 536870912 >/proc/sys/kernel/shmall     #Sets shmall to 512 MB
echo 536870912 >/proc/sys/kernel/shmmax     #Sets shmmax to 512 MB
```

You can also put these commands into a script run at startup.

Also change the settings for the system memory user limits by modifying a file called /etc/profile. Add lines similar to the following:

```
#sets user limits (ulimit) for system memory resources
ulimit -v 512000     #set virtual (swap) memory to 512 MB
ulimit -m 512000     #set physical memory to 512 MB
```

In this same file, you can permanently change the values for the *PATH* and *CLASSPATH* parameters by adding lines similar to the following:

```
#sets env values PATH and CLASSPATH
export PATH=$PATH:/usr/cache/bin:/path/to/j2sdk/bin:/.
export CLASSPATH=
      $CLASSPATH:/cache/dev/java/lib/CacheDB.jar:/path/to/otherjar/file:/.
```

C

Preparing for Caché Advanced Security

The material in this appendix is intended for those choosing to use the advanced security features of Caché 5.1. Your authentication and authorization method choices determine what tasks you need to perform to prepare the security environment before installing Caché. To help determine the level of security for your site before installing Caché, review the "Introduction" to the *Caché Security Administration Guide* for an overview of the authentication and authorization options available.

Important: If you are not using the Kerberos authentication method in your environment, you can bypass the "Preparing the Security Environment" sections. Review the information in the Initial Caché Security Settings section, particularly if you are choosing to use Normal or Locked Down Caché security.

All Caché supported platforms have versions of Kerberos supplied and supported by the vendor; see the appropriate operating system documentation for details. If you choose to use Kerberos, you must have a Kerberos KDC or a Windows domain controller available on your network. Microsoft Windows implements the Kerberos authentication protocol by integrating the KDC with other security services running on the domain controller.

If you *are* using Kerberos, see the instructions for the appropriate environment of the three outlined in the Preparing the Security Environment section: Windows-only Environment, Mixed Environment Using Windows Domain Controller, Non-Windows Environment.

Once you have defined the necessary service accounts on your Windows domain controller or tested the KDC functionality on your non-Windows Kerberos server, see the Initial Caché Security Settings section before you install Caché.

Important: If your security environment is more complex than those this document describes, contact the InterSystems Worldwide Response Center (WRC) for guidance in setting up such an environment.

C.1 Preparing the Security Environment

These sections describe the installation preparation for three types of environments:

1. Windows-only Environment
2. Mixed Environment Using Windows Domain Controller
3. Non-Windows Environment

Windows-only Environment

This configuration uses a Windows domain controller for KDC functionality with Caché servers and clients on Windows machines. A domain administrator creates domain accounts for running the Caché services on Caché servers.

See the Create Service Accounts for Windows Caché Servers with a Windows Domain Controller section for the requirements of using Windows Caché servers.

Mixed Environment Using Windows Domain Controller

This configuration uses a Windows domain controller with Caché servers and clients on a mix of Windows and non-Windows machines. See the following two sections for the requirements for using both Windows and non-Windows Cache servers:

- Create Service Accounts for Windows Caché Servers with a Windows Domain Controller
- Create Service Accounts for Non-Windows Caché Servers with a Windows Domain Controller

Non-Windows Environment

This configuration uses a UNIX-based or OpenVMS Kerberos KDC with Caché servers and clients all on non-Windows machines. See the following two sections for the requirements for using a UNIX, Mac OS, or OpenVMS KDC and Caché servers:

- Create Service Accounts for Non-Windows Caché Servers with a KDC
- Testing Kerberos KDC Functions

C.1.1 Create Service Accounts for Windows Caché Servers with a Windows Domain Controller

Before installing Caché in a Windows domain, the Windows domain administrator must create a service account for each Caché server instance on a Windows machine using the Windows domain controller. If you are running multiple instances of Caché on one Windows server, each must have a separate service account.

A suggested naming convention for these accounts is "`cache`*HOST*", which is the literal, `cache`, followed by the host computer name in uppercase. For example, if you are running a Caché server on a Windows machine called WINSRVR, name the domain account `cacheWINSRVR`.

When you create this account on the Windows domain controller, Caché requires that the account have the following characteristics:

- Set the **Password never expires** property.
- Set the **Use DES encryption types for this account** property
- Make the account a member of the **Administrators** group on the Caché server machine.

Important: If a domain-wide policy is in effect, you must add this service account to the policy for Caché to function properly.

The service account for a Caché server running on Windows must also have the right to log on as a service.

If you do have multiple Caché server instances on one machine, since the service account name must be unique, you may want to follow the default Caché practice of naming instances and use this instance name in place of the `cache` literal. For example, if you install two Caché server instances on the WINSRVR machine and use the default instance names (cache and cache2), name the two service accounts `cacheWINSRVR` and `cache2WINSRVR`.

Using a Windows client to communicate with a Caché server in a Kerberos environment, requires you to define a remote server providing a service principal name, among other security information. InterSystems has modified Windows server installation conventions to eliminate the variant Windows/UNIX/OpenVMS formats for service principal names when configuring these clients.

As a result, the domain administrator must set a service principal name for each Windows service account used to run a Caché server using the **setspn** command. The service principal name should take the form <service name>/<fully qualified hostname>, where service name

is in most cases `cache` (except when more than one Caché server instance is on one Windows machine). For example, the service principal names for the service accounts created in the previous examples become:

```
cache/WINSRVR.testdomain.com
cache2/WINSRVR.testdomain.com
```

When you add a remote server connection to the preferred server list on the Caché Cube, the Caché Server Manager pre-fills the service principal name if you choose Kerberos. Therefore, if you do not use these recommended naming conventions, take special care to enter the appropriate name in the **Service Principal Name** field. See the "Connecting to Remote Servers" chapter of the *Caché System Administration Guide* for the detailed procedure.

Note: For detailed information on the **setspn** tool, see the Microsoft Setspn.exe page for Windows 2000 or the Setspn Overview page for Windows 2003.

C.1.2 Create Service Accounts for Non-Windows Caché Servers with a Windows Domain Controller

Before you install Caché in a Windows domain, the Windows domain administrator must create a service account for each Caché server on a non-Windows machine that uses the Windows domain controller. Create one service account for each machine, regardless of the number of Caché server instances on that machine.

A suggested naming convention for these accounts is "`cache`*HOST*," which is the literal, `cache`, followed by the host computer name in uppercase. For example, if you run a Caché server on a non-Windows machine called UNIXSRVR, name the domain account `cacheUNIXSRVR`. For Caché servers on non-Windows platforms, this is the account that maps to the Kerberos service principal.

When you create this account on the Windows domain controller, Caché requires that the account have the following characteristics:

- Set the **Password never expires** property.
- Set the **Use DES encryption types for this account** property

To set up a non-Windows Caché server in the Windows domain, it must have a keytab file from the Windows domain. A keytab file is a file containing the service name for the Caché server and its key.

To accomplish this, map the Windows service account (`cacheUNIXSRVR`, in this example) to a service principal on the Caché server and extract the key from the account using the

ktpass command-line tool on the domain controller; this is available as part of the Windows support tools from Microsoft.

The command maps the account just set up to an account on the UNIX-based or OpenVMS machine; it also generates a key for the account. The command must specify the following parameters:

Parameter	Description
-princ	The principal name (in the form *cache/<fully qualified hostname>@<kerberos realm>*).
-mapuser	The name of the account created (in the form *cache<HOST>*).
-pass	The password specified during account creation.
-crypto	The encryption type to use (used the default, *DES-CBC-CRC*, unless specified otherwise).
-out	The keytab file you generate to transfer to the Caché server machine and replace or merge with your existing keytab file.

Important: The principal name on UNIX-based and OpenVMS platforms must take the form shown in the table with the literal `cache` as the first part.

Once you have generated a key file, move it to a file on the Caché server with the following characteristics:

- On MacOS and most versions of UNIX, the pathname is <cache-installation-directory>/mgr/cache.keytab. On Tru64, the pathname is /krb5/v5srvtab; on SuSE Linux, it is /etc/krb5.keytab.

  On OpenVMS, the file is cache.keytab and is located in the manager's directory.
- It is owned by the user that owns the Caché installation and the group `cacheusr`.
- On UNIX and MacOS, its permissions are `640`; on OpenVMS, its permissions are `[S:RWD,O:RWD,G:R,W:]`.

C.1.3 Create Service Accounts for Non-Windows Caché Servers with a KDC

In a non-Windows environment, you must create service principal accounts for all UNIX, Mac OS, or OpenVMS Caché servers using a UNIX, Mac OS, or OpenVMS KDC. Once you have an operational KDC, you need to add a service principal account for each Caché server.

The service principal name is of the form *cache/<fully qualified hostname>@<kerberos realm>*.

Once you have created this principal, extract its key to a key file on the Caché server with the following characteristics:

- On Mac OS and most versions of UNIX, the pathname is <cache-installation-directory>/mgr/cache.keytab. On Tru64, the pathname is /krb5/v5srvtab; on SuSE Linux, it is /etc/krb5.keytab.

  On OpenVMS, the file is cache.keytab and is located in the manager's directory.
- It is owned by the user that owns the Caché installation and the group `cacheusr`.
- On UNIX and Mac OS, its permissions are `640`; on OpenVMS, its permissions are `[S:RWD,O:RWD,G:R,W:]`.

C.1.4 Testing Kerberos KDC Functions

When using Kerberos in a system of only non-Windows servers and clients, it is simplest to use a native UNIX-based or OpenVMS KDC rather than a Windows domain controller. (Tru64 UNIX, however, can only use the Windows domain controller for KDC functionality.) Consult the vendor documentation on how to install and configure the KDC; these are usually tasks for your system administrator or system manager.

When installing Kerberos, there are two sets of software to install:

- The KDC, which goes on the Kerberos server machine.
- There also may be client software, which goes on all machines hosting Kerberos clients. This set of software can vary widely by operating system. Consult your operating system vendor documentation for what client software exists and how to install it.

After installing the required Kerberos software, you can perform a simple test using the **kadmin**, **kinit**, and **klist** commands to add a user *principal* to the Kerberos database, obtain a TGT (ticket-granting ticket) for this user, and list the TGT.

Once you successfully complete a test to validate that Kerberos is able to provide tickets for registered principals. You are ready to install Caché.

C.2 Initial Caché Security Settings

During installation, there is a prompt for one of three sets of initial security settings: Minimal, Normal, and Locked Down. This selection determines the initial authorization configuration settings for Caché services and security, as shown in the following sections:

- Initial User Security Settings
- Initial Service Properties

If you select Normal or Locked Down for your initial security setting, you must provide additional account information to the installation procedure. If you are using Kerberos authentication, you must select Normal or Locked Down mode. See the User Account Configuration section for details.

C.2.1 Initial User Security Settings

The following tables show the user password requirements and settings for predefined users based on which security level you choose.

Initial User Security Settings

Security Setting	Minimal	Normal	Locked Down
Password Pattern	3.32ANP	3.32ANP	8.32ANP
Enable _SYSTEM User	Yes	Yes	No
Roles assigned to UnknownUser	%All	None	None

Password Pattern

When Caché is installed, it has a default set of password requirements. For locked-down installations, the initial requirement is that a password be from 8 to 32 characters, and can consist of alphanumerics or punctuation; the abbreviation for this is 8.32ANP. Otherwise, the initial requirement is that the password be from 3 to 32 characters, and can consist of alphanumerics or punctuation (3.32ANP).

Enable _SYSTEM User

In versions of Caché prior to 5.1, all installed systems included an SQL System Manager user named _SYSTEM with a password of `SYS`. Caché 5.1 creates the _SYSTEM and additional

predefined users using the password you provide during the installation as shown in the following table.

Initial Password Settings for Predefined Users

Password Setting	Minimal	Normal	Locked Down
_SYSTEM	SYS	Same as installing user	Same as installing user
Admin	SYS	Same as installing user	Same as installing user
SuperUser	SYS	Same as installing user	Same as installing user
CSPSystem	SYS	Same as installing user	Same as installing user

For more details on these predefined users, see the Predefined User Accounts section of the "Users" chapter of the *Caché Security Administration Guide*.

Roles Assigned to UnknownUser

When an unauthenticated user connects, Caché assigns a special name, `UnknownUser`, to $USERNAME and assigns the roles defined for that user to $ROLES. The UnknownUser is assigned no roles when choosing a security level other than Minimal.

For more details on the use of $USERNAME and $ROLES, see the "Users" and "Roles" chapters of the *Caché Security Administration Guide*.

After installation, you can view and maintain these settings at the **[Home] > [Security Management] > [Users]** page of the System Management Portal.

C.2.2 Initial Service Properties

Services are the primary means by which users and computers connect to Caché. For detailed information about the Caché services see the "Services" chapter of the *Caché Security Administration Guide*.

Initial Service Properties

Service Property	Minimal	Normal	Locked Down
Use Permission is Public	Yes	Yes	No
Requires Authentication	No	Yes	Yes
Enabled Services	Most	Some	Fewest

Use Permission is Public

If the Use permission on a service resource is Public, any user can employ the service; otherwise, only privileged users can employ the service.

Requires Authentication

For installations with initial settings of locked down or normal, all services require authentication of some kind (Caché login, operating-system–based, or Kerberos). Otherwise, unauthenticated connections are permitted.

Enabled Services

The initial security settings of an installation determine which of certain services are enabled or disabled when Caché first starts. The following table shows these initial settings:

Initial Enabled Settings for Services

Service	Minimal	Normal	Locked Down
%Service_Bindings	Enabled	Enabled	Disabled
%Service_CSP	Enabled	Enabled	Enabled
%Service_CacheDirect	Enabled	Disabled	Disabled
%Service_CallIn	Enabled	Disabled	Disabled
%Service_ComPort	Enabled	Disabled	Disabled
%Service_Console*	Enabled	Enabled	Enabled
%Service_DCP	Disabled	Disabled	Disabled
%Service_DDP	Disabled	Disabled	Disabled
%Service_ECP	Disabled	Disabled	Disabled
%Service_LAT*	Disabled	Disabled	Disabled
%Service_MSMActivate	Disabled	Disabled	Disabled
%Service_Monitor	Disabled	Disabled	Disabled
%Service_Shadow	Disabled	Disabled	Disabled
%Service_Telnet*	Disabled	Disabled	Disabled
%Service_Terminal†	Enabled	Enabled	Enabled
%Service_WebLink	Disabled	Disabled	Disabled

* Service exists on Windows servers only

† Service exists on non-Windows servers only

After installation, you can view and maintain these services at the **[Home] > [Security Management] > [Services]** page of the System Management Portal.

C.2.3 User Account Configuration

If you select Normal or Locked Down for your initial security setting, you must provide additional information to the installation procedure:

1. **User Credentials** for *Windows* server installations only — Choose an existing Windows user account under which to run the Caché service. You can choose the default system account, which runs Caché as the Windows Local System account, or enter a defined Windows user account.

   Important: If you are using Kerberos, you must enter a defined account that you have set up to run the Caché service. InterSystems recommends you use a separate account specifically set up for this purpose as described in the Create Service Accounts for Windows Caché Servers section.

   The installation verifies the following if you enter a defined user account:

   - The account exists on the domain.
   - You have supplied the correct password.
   - The account has local administrative privileges on the server machine.

2. **Caché Users Configuration** for *Windows* installations — The installation creates a Caché account with the %All role for the user that is installing Caché to grant that user access to services necessary to administer Caché.

   Owner of the instance for *non-Windows* installations — Enter a user name under which to run Caché. Caché creates an account for this user with the %All role.

   Enter and confirm the password for this account. The password must meet the criteria described in the Initial User Security Settings table.

   Setup creates the following Caché accounts for you: _SYSTEM, Admin, SuperUser, CSPSystem, and the user account you enter in this step. See the Initial Password Settings for Predefined Users table for more information about these accounts.

Important: If you select Minimal for your initial security setting on a *Windows* installation, but Caché requires network access to shared drives and printers, you must manually change the Windows user account under which to run the Caché service, choosing an existing or creating a new account the has local administrative privileges on the server machine.

The instructions in the platform-specific chapters of this guide provide details about installing Caché. After reading the *Caché Security Administration Guide* introduction and following the procedures in this appendix, you are prepared to provide the pertinent security information to these installation procedures.

D

Upgrading Caché from Versions Prior to 4.1

This appendix is intended for customers who are upgrading from versions of Caché prior to release 4.1. Topics in this document include:

- Supported Upgrade Paths
- Upgrade Tasks

Please review the Caché 5.1 Conversion Guide for issues that may apply to your site.

D.1 Supported Upgrade Paths

Indirect Upgrades

The following *indirect upgrade* paths to Caché 5.1 are supported:

- Caché 4.0, 4.0.1, 4.0.3
- Caché 3.2, 3.2.1, 3.2.2, 3.2.3
- Caché 3.1.2
- Caché 3.0
- Caché 2.3
- Caché 2.1.6, 2.1.7, 2.1.8, 2.1.9

Indirect Conversion and Upgrades

The following *indirect conversion and upgrade* paths to Caché 5.1 are supported:

- DSM 7.2
- DTM 4.10, 6.6
- ISM 5.10, 6.4
- MSM 4.4.1

D.2 Upgrade Tasks

Tasks necessary to upgrade from these previous releases of Caché are grouped by the following topics:

- Pre-Installation Upgrade Tasks
- Upgrading from ISM
- Post-Installation Upgrade Tasks

The **^GBLOCKCOPY** routine can be used to perform several tasks associated with upgrading.

See the Performing Upgrade Tasks with ^GBLOCKCOPY section of the *Using the Caché ^GBLOCKCOPY Routine* article for additional information.

D.2.1 Pre-Installation Upgrade Tasks

The following upgrade tasks are necessary regardless of the platform on which you are upgrading and running Caché. Perform these tasks before you run the Caché installation procedures:

1. *Obtain an updated license key* — the key structure is new in Caché 5.1, so upgrades to Caché 5.1 require an updated key.
2. *Backup system* — before upgrading, InterSystems recommends that you run a complete backup of your system. Use your customary full operating system backup procedures.
3. *Check system integrity* — run a system integrity check on existing directories to ensure there is no corruption in any of the databases.

4. *Save custom routines and globals* — to prevent your own routines and globals in the %SYS namespace from being affected by the upgrade installation, ensure that they have names that begin with "Z" , "z" , "%Z" , or "%z" . All .int and .obj routines (except for Z*, z*, %Z*, and %z*) are deleted from the %SYS namespace when upgrading.

   On an upgrade, the CACHELIB, CACHETEMP, DOCBOOK, and SAMPLES databases are completely replaced.

   Any .mac or .inc routines are not affected during the upgrade.

5. *Save user files* — additional files and directories are also deleted and/or replaced. Verify that any user files are in User directories before upgrading. See the Files Deleted or Replaced on Upgrade section for a list of specific files; this list may change, so it is safest to put your files in User directories.

6. *Export custom classes* — classes in the %SYS namespace are *not* saved, even if they begin with "Z" , "z" , "%Z" , or "%z" . To save your classes when upgrading, export them before the upgrade and import them back into Caché when the upgrade is complete.

7. *Check routine labels* — routines which used to compile with no errors, but had duplicate labels, now receive errors during compilation, but run without error. The previous utility to check all application routines for duplicate labels, **%LBLRDEF**, is not shipped with Caché 5.1.

8. *Check routine references* — Older versions of Caché truncated references at 8 characters. Check all application routines for routine references over 8 characters (including function calls, **DO** routine calls, **$ZT** calls, and so on). Caché now recognizes up to 63 significant characters in routine names, and this could possibly cause *<NOROUTINE>* errors at compile and/or run time.

Perform additional tasks outlined in the appropriate section if you are Upgrading Caché on OpenVMS.

D.2.1.1 Files Deleted or Replaced on Upgrade

In addition to the CACHELIB, CACHETEMP, DOCBOOK, and SAMPLES databases, the following files and directories are deleted during the installation of the indicated components.

Files Deleted or Replaced on Windows Upgrades

Installed Component	File or directory deleted
Caché Application Development (Object utilities)	<CacheSys>\Bin\Cache*.oca
	<CacheSys>\Bin\CacheList.ocx
	<CacheSys>\Bin\CacheQuery.ocx
	<CacheSys>\Bin\VISM.ocx
	<CacheSys>\Bin\atl.dll
	<CacheSys>\Bin\Cacheie.dll
	<CacheSys>\Bin\ice.exe
	<CacheSys>\Bin\iceres.dll
	<CacheSys>\Java\
	<CacheSys>\Corba\
	<CacheSys>\CSPKit\
Web Server (CSP) Gateway or Caché Engine	<CacheSys>\CSP\broker\
	<CacheSys>\CSP\cachelib*
	<CacheSys>\CSP\RunTime*
Caché Engine Link Libraries	<CacheSys>\Source\Cache\Win95*
	<CacheSys>\Source\Cache\i386*
	<CacheSys>\Source\Cache\WinP4\
	<CacheSys>\Source\Cache\alpha*
WebLink	<CacheSys>\Weblink\Doc\
	<CacheSys>\Weblink\Scripts\
	<CacheSys>\Weblink\i386\
	<CacheSys>\Weblink\alpha\
	<CacheSys>\mgw.ini is renamed mgwsave.ini

Installed Component	File or directory deleted
Caché Tools and Utilities	<CacheSys>\Dev\
	<CacheSys>\CSP\samples\
	<CacheSys>\Bin\CCtrlPnl.exe*
	<CacheSys>\Bin\CExplore.exe*
Documentation	<CacheSys>\Docs\

* No longer exists in Caché 5.1

Replace *<CacheSys>* with the name of your installation directory; CacheSys is the default.

The following directories are deleted when upgrading Caché on UNIX and OpenVMS platforms:

- <CacheSys>/dev/
- <CacheSys>/csp/cachelib/

D.2.1.2 Upgrading Caché on OpenVMS

There is an additional upgrade task for OpenVMS clustered systems:

Cleanup Cluster PIJ Files

Before upgrading a member of a Caché for OpenVMS cluster system to Caché 5.1, cleanly shut down all members of the Caché cluster and remove the CACHE.PIJ file. If you do not remove this file, the installation is not upgraded and error messages are written in the cconsole.log for startup:

```
Cache (2100036c) Tue Aug 1 14:28:59 2004
Activating Namespaces
Cache (21000404) Tue Aug 1 14:28:59 2004 Cluster image journal
is incompatible with this version
Cache (21000404) Tue Aug 1 14:28:59 2004 Unable to join the cluster
Cache (21000404) Tue Aug 1 14:29:00 2004
ENQdaemon exited due to VMS error code (decimal) 0
```

Cluster Configuration Changes

Prior to Caché 5.0 when configuring a Caché cluster it was necessary to define the network type for the cluster (UDP or Ethernet) and define DCP connections between the cluster members. This is no longer necessary. DCP over UDP or Raw Ethernet can still be used for Caché cluster network traffic, but it is preferable to use ECP networking. ECP is the default for new installations.

After the upgrade to 5.1 from a version earlier than 5.0, a Caché cluster configuration needs to be changed manually. The changes are in two places of the System Management Portal:

- Enable the ECP service from the **[Home] > [Security Management] > [Services]** page. Click **%Service_ECP**, select the **Service enabled** check box, and click **Save**.
- From the **[Home] > [Configuration] > [Legacy Network Connections]** page, delete any DCP connections from the network table that you do not need; they were only there to support clusters.

The upgrade does not automatically make these changes because it cannot detect which DCP connections support clusters and which might be for communicating with machines that are still running a prior release of Caché. With ECP, networking the cluster automatically configures the network tables as needed; it is not necessary to define any ECP connections between the cluster members to support Caché clusters. However, the user cannot access the ECP connection created automatically. If the configuration requires ECP, to gain access to privately mounted databases on another cluster member, you must define those connections.

D.2.2 Upgrading from ISM

If you are upgrading to Caché 5.1 from ISM 6.4 or 5.10, run the preconversion routine, install Caché, and then convert the existing ISM database to a Caché database. The preconversion routine examines the current ISM configuration and builds a Caché configuration file that is used for the upgrade. See the section appropriate for your operating system:

- Run the Preconversion Routine on UNIX
- Run the Preconversion Routine on OpenVMS

If you migrate between platforms, either of the same or different endian types, the conversion utility cannot deduce the chaining between the primary and extension database volumes, which is based on the actual location of each file. The only way to indicate the new location is by relabeling each volume with the **^LABEL** utility.

If there is only one volume, manual file renaming and label correction are not required.

The appropriate steps to convert a multivolume database are:

1. If the database volumes have been copied from a machine with different endian orientation than the current one, run **cvendian** to convert them. Provide the directory names of all volumes in a single call to **cvendian**. For example:

```
cvendian tstdir1/OPENM.DAT tstdir2/OPENM.EXT tstdir3/OPENM.EXT
```

2. Manually rename the primary volume file to CACHE.DAT and each extension volume file to CACHE.EXT. You can do this before running step 1, if desired.
3. Run **^LABEL** and follow the instructions for correcting volume labels and links, starting with the primary volume, then each extension.
4. Run **START^%SYSCONV(primary_volume)** as described in the Run the System Conversion Utility section that follows.

D.2.3 Run the Preconversion Routine on UNIX (UNIX ISM Upgrades Only)

If upgrading an ISM 5.10 or 6.4 system, perform the following procedures; otherwise bypass this section.

After running your backup and transferring the Caché files from the distribution media, run the preconversion routine. The preconversion routine examines the current ISM configuration and builds a Caché configuration file that is used for the upgrade. To run the preconversion script:

1. Start ISM:

   ```
   ./mstart
   ```

2. Copy the cpreconv script from the distribution media into the current manager's directory.
3. Copy the cvtcfg program from the distribution media either into the current manager's directory or into the directory where you plan to install Caché.
4. Use the **%G** utility to check the *^SYS("UCI")* global. Make sure it properly represents all current application databases.
5. Start the preconversion routine using the following operating system command:

   ```
   ./cpreconv
   ```

6. The script prompts for the Caché installation directory, which is where the routine stores the Caché configuration file that it creates. If the directory does not exist, the routine creates it. You can install into any directory, the default is the current directory.

   The routine creates one cache.cpf file for each system configuration and the default network and namespace configurations associated with that system configuration.

7. Shut down ISM:

   ```
   ./mstop
   ```

D.2.4 Run the Preconversion Routine on OpenVMS (OpenVMS ISM Upgrades Only)

If upgrading an ISM 5.10 or 6.4 system, perform the following procedures; otherwise bypass this section.

After running your copy and transferring the Caché files from the distribution media, run the preconversion routine. The preconversion routine examines the current ISM configuration and builds a Caché configuration file that is used for the upgrade. To run the preconversion script:

1. Start ISM:

   ```
   @MSTART
   ```

2. Copy the CPRECONV script from the distribution media into the current manager's directory.

3. Copy the CVTCFG program from the distribution media either into the current manager's directory or into the directory where you plan to install Caché.

4. Use the **%G** utility to check the *^SYS("UCI")* global. Make sure it properly represents all current application databases.

5. Start the preconversion routine using the following operating system command:

   ```
   @CPRECONV
   ```

6. The script prompts for the Caché installation directory, which is where the routine stores the Caché configuration file that it creates. If the directory does not exist, the routine creates it. You can install into any directory, the default is [CACHESYS].

   The routine creates one cache.cpf file for each system configuration and the default network and namespace configurations associated with that system configuration.

7. Shut down ISM:

   ```
   @MSTOP
   ```

D.2.5 Post-Installation Upgrade Tasks

After the installation is complete, there are additional upgrade tasks:

- *Convert Legacy Databases* — If you are upgrading from an InterSystems database product earlier than Caché 2.1.6, convert your databases to the new CACHE.DAT file format using **GBLOCKCOPY** or **%SYSCONV**.
- *Recompile Objects* — you must recompile any Caché Objects applications after installation if you are upgrading from a prior version of Caché. There are additional upgrade and compatibility issues relating to Caché Objects that can be found in the New Caché Object Features chapter of the *Release Notes*.
- *Adjust Database Memory Cache* — when upgrading from Caché 4.0.4 or earlier, the size of the 2-KB global buffer pool remains the same, and a minimal amount of 8-KB memory cache is allocated. If you plan on creating new 8-KB databases or converting your old format (2 KB) to new format (8 KB), then adjust your allocated 8-KB database memory cache accordingly by clicking **Manually** from the **[Home] > [Configuration] > [Memory and Startup]** page of the System Management Portal and updating the memory allocations.
- *Convert the System Manager's Directory* — for ISM upgrades, convert the system manager's directory. You can run the system conversion utility, **%SYSCONV**, or use the **^GBLOCKCOPY** utility. See the Performing Upgrade Tasks with ^GBLOCKCOPY section of the *Using the Caché ^GBLOCKCOPY Routine* article for details.
- *Upgrade CSP Gateway* — If your CSP Gateway is on a separate machine from the Caché server you are upgrading, you must also upgrade the CSP Gateway on that machine.

D.2.5.1 Run the System Conversion Utility

Once the installation is complete on an ISM upgrade, run the system conversion utility, **%SYSCONV**, from the system manager's namespace. This automatically runs all required conversions and re-collates routine and object code globals. A converted database may use slightly more disk storage after conversion. Before running the **%SYSCONV** utility:

- Make sure that the current manager's directory contains a valid license key file. The **%SYSCONV** utility starts several processes to speed conversion of the databases, if permitted by your license type.
- If you use National Language Support (NLS), configure your locale properly on Caché before running the ISM conversion utility. After the upgrade, you need to define any proprietary NLS tables you may have again.

To run the system conversion utility:

1. Start Caché, ensuring that no other users have access to Caché during the conversion.
2. At the Caché prompt (indicated below by ">"), run the **%SYSCONV** routine.

To convert all databases, change to the manager's namespace, %SYS, and run:

```
> Do ALL^%SYSCONV
```

To convert a single database, run:

```
> Do START^%SYSCONV(<directory-name>)
```

The **ALL^%SYSCONV** procedure upgrades only databases recorded in the UCI list. To convert a database that is not in the UCI list, use the **START^%SYSCONV** procedure.

3. To check the status of the conversion, run:

```
> Do STATUS^%SYSCONV
```