InterSystems® Ensemble

Preparing to Use Ensemble

Version 2018.1
2019-09-20

For Support questions about any InterSystems products, contact:

InterSystems Worldwide Response Center (WRC)

Tel:	+1-617-621-0700
Tel:	+44 (0) 844 854 2917
Email:	support@InterSystems.com

Table of Contents

About This Book

Welcome and thank you for using Ensemble.

This book provides a road map of tasks and information required to install and prepare your Ensemble environment whether you are a first-time Ensemble user or upgrading an existing Ensemble instance. It provides important tips for getting started and contains the following chapters:

- Road Map to Using Ensemble

- Installing Ensemble

- Upgrading Ensemble

- Planning an Ensemble Server Deployment

For a detailed outline, see the table of contents.

InterSystems Ensemble shares many underlying core technologies with InterSystems Caché. This book refers you to books in both documentation sets. The Caché books are particularly important as you first set up the system. After initial configuration, the Ensemble books become your primary source of information.

The following books are referenced throughout this guide in the appropriate places for the task you are performing:

- The online *InterSystems Supported Platforms* document for this release

- *Ensemble Release Notes*

- *Caché Release Notes and Upgrade Checklist*

- *Caché Installation Guide*

- *Caché System Administration Guide*

- *Caché High Availability Guide*

- *Caché Distributed Data Management Guide*

For general information, see *Using InterSystems Documentation*.

You may also want to view the following title in the InterSystems Video Library: *Getting Started with Ensemble: The Basic Components*

1

Road Map to Using Ensemble

Depending on how you are using Ensemble or how long you have been using it, you most likely are interested in reading different topics. The following sections provide an outline of the documentation resources where you may want to start:

- New Ensemble Developer
- Experienced Ensemble Developer
- New Ensemble Administrator
- Experienced Ensemble Administrator

1.1 New Ensemble Developer

If you are a developer preparing to use Ensemble for the first time, see the following resources:

1. "Installing Ensemble" chapter in this guide, which provides you with instructions on installing Ensemble and points you to the appropriate topics in the Caché documentation.

2. *Introducing Ensemble* provides an overview of Ensemble and options that you may want to include in your Ensemble productions, with pointers to more detailed information.

3. *Ensemble Best Practices* describes best practices for designing, developing, and maintaining Ensemble productions.

4. *Developing Ensemble Productions*, which, in turn, points you to specific topics in the *Developer Guides* and *Adapter Guides*.

5. *Configuring Ensemble Productions* describes the details of configuring items in an Ensemble production.

1.2 Experienced Ensemble Developer

If you are a developer upgrading your version of Ensemble, see the following resources:

1. *Ensemble Release Notes* to see how upgrading to this release affects your existing development environment.

2. "Upgrading Ensemble" chapter in this guide.

3. *Ensemble Best Practices* describes best practices for designing, developing, and maintaining Ensemble productions.

4. *Developing Ensemble Productions*, which, in turn, points you to specific topics in the *Developer Guides* and *Adapter Guides*.

5. *Configuring Ensemble Productions* describes the details of configuring items in an Ensemble production.

1.3 New Ensemble Administrator

If you are an administrator preparing to work with Ensemble productions for the first time, see the following resources:

1. "Installing Ensemble" chapter in this guide.

2. *Introducing Ensemble* provides an overview of product features.

3. *Managing Ensemble* explains how to use the browser-based Management Portal to manage the Ensemble environment and points to detailed descriptions of the specific functions performed using the Management Portal.

4. *Configuring Ensemble Productions* explains how to build and configure Ensemble productions and production components as well as update configurations of existing productions.

5. *Monitoring Ensemble* explains how to monitor your Ensemble environment and the various Ensemble production components.

1.4 Experienced Ensemble Administrator

If you are an administrator upgrading your Ensemble production environment, see the following documentation resources:

1. *Ensemble Release Notes* to see how upgrading to this release affects your existing Ensemble productions.

2. "Upgrading Ensemble" chapter in this guide.

3. *Managing Ensemble* explains how to use the browser-based Management Portal to manage the Ensemble environment and points to detailed descriptions of the specific functions performed using the Management Portal.

4. *Configuring Ensemble Productions* explains how to build and configure Ensemble productions and production components as well as update configurations of existing productions.

5. *Monitoring Ensemble* explains how to monitor your Ensemble environment and the various Ensemble production components.

2

Installing Ensemble

The Ensemble installation is almost identical to Caché. This chapter points to the appropriate sections of the Caché documentation for your particular environment and highlights some additional facts you should consider for your Ensemble installation. To install Ensemble, perform the following steps:

1. Ensemble runs on several different platforms. Before installing, check the the online *InterSystems Supported Platforms* document for this release document provided with the Ensemble kit to verify it runs on your particular version of the supported operating system. The document also lists browsers, web servers, and other technologies for use with InterSystems products.

2. To upgrade from previous releases of Ensemble, see the "Upgrading Ensemble" chapter.

3. For detailed installation instructions, see the appropriate *Caché Installation Guide* chapter for Windows, , UNIX®, or Mac. The primary difference for Ensemble is that, where the Caché installation sequence provides a choice between 8-bit and Unicode, Ensemble automatically performs a Unicode installation.

   On Windows, the default installation directory is c:\InterSystems\Ensemble rather than c:\InterSystems\Cache.

   There are additional considerations as described in the following topics:

   • Licenses

   • Namespaces

   • UNIX®

   • High Availability

If you have any questions or encounter any problems while installing Ensemble, contact the InterSystems Worldwide Response Center (WRC) for support.

2.1 Licenses

You can enter an Ensemble license key during installation, or at any time after installation. Use the instructions in "Managing Caché Licensing" in the *Caché System Administration Guide*.

If you have questions or encounter any problems in this regard, contact the WRC.

2.2 Namespaces

In InterSystems products, a *namespace* is a collection of data and programs in a virtual work space. InterSystems documentation provides a great deal of information about namespaces.

You can use Ensemble only within specific namespaces. For details, see "Environmental Considerations" in *Developing Ensemble Productions*.

2.3 UNIX® Considerations

If you are running the Ensemble server on a UNIX® platform, follow the instructions in the chapter "Installing Caché on UNIX® and Linux" in the *Caché Installation Guide*. Pay special attention to "Post-installation Tasks."

The installation procedure installs a private Apache Web server so that you can access the Management Portal; therefore, a UNIX® system does not require a client on a Windows machine to perform system configuration and management tasks. You do, however, require an Ensemble client (or a server, which includes the client) on a Windows system to use the Studio development tool. For instructions, see "Install Caché Client on Windows for Development" in the chapter "Installing Caché on UNIX® and Linux" in the *Caché Installation Guide*.

2.4 High Availability Considerations

The following Caché books contain information on how to install Ensemble on multiple clustered machines to provide failover capabilities in case of problems on the primary server. They also describe other Caché technologies that help provide high availability:

- *Caché System Administration Guide*
- *Caché Distributed Data Management Guide*
- *Caché Data Integrity Guide*
- *Caché High Availability Guide*

If you require further information to help you develop a failover and backup strategy tailored for your environment, or to review your current practices, please contact the InterSystems Worldwide Response Center (WRC).

3
Upgrading Ensemble

When deciding to upgrade to the current release of Ensemble you may want to review the new features and enhancements as well as any compatibility issues of this and any intervening releases since you last installed Ensemble. See the following for this information:

- New Features and Enhancements in *Ensemble Release Notes*

- Upgrade Compatibility Issues in *Ensemble Release Notes*

- Descriptions of new features in previous releases in *Ensemble Release Notes Archive*

If you are using the InterSystems Mirroring technology with Ensemble, see the "Upgrading an Ensemble Mirror" section for additional specific instructions.

Ensemble runs on top of Caché. This means that, in addition to changes in Ensemble, the new Ensemble release includes a large number of changes in the underlying Caché technologies as outlined in the following table.

IF you are upgrading from ...	THEN the underlying changes resemble ...
Ensemble 2017.2 to Ensemble 2018.1	Caché 2017.2 to Caché 2018.1
Ensemble 2017.1 to Ensemble 2018.1	Caché 2017.1 to Caché 2018.1
Ensemble 2016.2 to Ensemble 2018.1	Caché 2016.2 to Caché 2018.1
Ensemble 2016.1 to Ensemble 2018.1	Caché 2016.1 to Caché 2018.1
Ensemble 2015.2 to Ensemble 2018.1	Caché 2015.2 to Caché 2018.1
Ensemble 2015.1 to Ensemble 2018.1	Caché 2015.1 to Caché 2018.1
Ensemble 2015.1 to Ensemble 2018.1	Caché 2015.1 to Caché 2018.1
Ensemble 2014.1 to Ensemble 2018.1	Caché 2014.1 to Caché 2018.1
Ensemble 2013.1 to Ensemble 2018.1	Caché 2013.1 to Caché 2018.1
Ensemble 2012.2 to Ensemble 2018.1	Caché 2012.2 to Caché 2018.1
Ensemble 2012.1 to Ensemble 2018.1	Caché 2012.1 to Caché 2018.1
Ensemble 2010.2 to Ensemble 2018.1	Caché 2010.2 to Caché 2018.1
Ensemble 2010.1 to Ensemble 2018.1	Caché 2010.1 to Caché 2018.1
Ensemble 2009.1 to Ensemble 2018.1	Caché 2009.1 to Caché 2018.1
Ensemble 2008.2 to Ensemble 2018.1	Caché 2008.2 to Caché 2018.1

IF you are upgrading from ...	THEN the underlying changes resemble ...
Ensemble 2008.1 to Ensemble 2018.1	Caché 2008.1 to Caché 2018.1
Ensemble 2007.1 to Ensemble 2018.1	Caché 2007.1 to Caché 2018.1
Ensemble 4.0.1 to Ensemble 2018.1	Caché 5.2.3 to Caché 2018.1
Ensemble 4.0 to Ensemble 2018.1	Caché 5.2.1 to Caché 2018.1

Important: If you are upgrading from Ensemble 2008.2 or an earlier version, you cannot upgrade directly to Ensemble 2018.1, but must first upgrade to an intermediate version. For details, see Supported Upgrade Paths in the *Caché Installation Guide*.

Note: There was no 2011.1 release of Ensemble.

The *Caché Release Notes and Upgrade Checklist* book in the Caché documentation is helpful when you are upgrading from previous Ensemble versions.

Beginning with Ensemble 2007.1, Ensemble runs on top of the same numbered version of Caché.

Ensemble Upgrade Paths

Before running an upgrade installation, review the information referenced in this chapter to determine if there are any tasks to complete before upgrading. The following upgrade paths to this release of Ensemble are supported:

- Ensemble 2017.1
- Ensemble 2016.2
- Ensemble 2016.1
- Ensemble 2015.2
- Ensemble 2015.1
- Ensemble 2014.1
- Ensemble 2013.1
- Ensemble 2012.2
- Ensemble 2012.1
- Ensemble 2010.2
- Ensemble 2010.1
- Ensemble 2009.1
- Ensemble 2008.2
- Ensemble 2008.1
- Ensemble 2007.1

To upgrade to this release of Ensemble, follow the instructions in Upgrading from Ensemble 2007.1 or Later.

The following upgrade paths to this release of Ensemble require additional steps in the upgrade procedure:

- Ensemble 4.0.1
- Ensemble 4.0

Additionally follow the instructions in Upgrading from Ensemble 4.0 or 4.0.1.

If you have an Ensemble version prior to 4.0, or if you have been using a field test version of Ensemble and you want to upgrade it to the most recent Ensemble product, contact the InterSystems Worldwide Response Center (WRC) for guidance.

3.1 Upgrading from Ensemble 2007.1 or Later

If you are upgrading from Ensemble 2007.1 or later to this release of Ensemble, in addition to deleting the files outlined in the Caché upgrade documentation, the upgrade procedure deletes the following Ensemble files:

- Any user files in the ENSDEMO namespace

- Any user files in the packages normally reserved for Ensemble (CSPX, Demo, Ens, or EnsLib)

If you have customized any files in these namespaces or added new files to these namespaces, you should preserve them before upgrading. To preserve such files, export them before proceeding with the upgrade. You may import them to any namespace after the upgrade is complete.

Important: You cannot upgrade an Ensemble instance that has a running production; in fact, the upgrade procedure shuts down Ensemble if it is running. It is best to stop any running productions and then cleanly shut down Ensemble before you begin the upgrade installation.

Note: In some environments, you need to create namespaces in Ensemble and then remove the Ensemble mappings from the namespace. You should only do this if you understand the impact of removing these mappings. If you have removed these mappings, you must also disable Ensemble for the namespace. You can create a new namespace that is disabled for Ensemble by clearing the **Make this an Ensemble namespace** check box when creating the namespace. For more information, see "Create a Namespace on an Ensemble Instance" in the *Caché System Administration Guide*. If you have already created a namespace and have removed the Ensemble mappings, you can disable Ensemble in the namespace by opening a Terminal window and issuing the following command:

```
Do ##class(%Library.EnsembleMgr).DisableNamespace("namespace",1)
```

If you have removed the mappings, but not disabled Ensemble, upgrading Ensemble re-creates the mappings. For more information on this topic contact the InterSystems Worldwide Response Center.

To upgrade an existing Ensemble installation to this release of Ensemble:

1. Stop all running productions.

2. Disable the production auto-start option for all productions in all namespaces. See "Managing Production Auto-Start" in *Managing Ensemble*.

3. Perform exports to preserve files that are deleted on upgrade.

4. If you use custom search tables and are upgrading from Ensemble 2010.2 or earlier, run the following command from a terminal in the ENSLIB namespace *before* the upgrade to prepare the existing metadata for relocation:

```
ENSLIB>Merge ^%SYS("Ensemble","Upgrade","SearchTable","Data") = ^Ens.Config.SearchTablePropD
```

Ensemble checks this location during upgrade and adds any of the existing metadata into namespaces where the originating search table classes exist. This occurs before any compilation of search tables in the namespace. For details, see "Updated Search Table Validation" in the chapter "Upgrade Compatibility Issues" in the *Ensemble Release Notes*.

Ensemble automatically recompiles all search table classes during the upgrade process, so do not Kill the global nor import the contents after the upgrade. The upgrade process ensures that Ensemble correctly retains search table metadata after the upgrade. It also ensures that the converted data is consistent, so InterSystems strongly recommends that you

thoroughly test your upgrade process so that you can address any issues in the original metadata prior to the actual upgrade.

5. If you have a large message warehouse and have either run Tune Table or have set the extent size and selectivity values manually, you should either verify that the new default values match your existing system or run Tune Table after the upgrade. If you have not run Tune Table or manually set the *Selectivity* value, the default values are likely to be an improvement on your performance.

   If you have taken action to optimize access to this table, take the following actions to ensure that the system performs well after the upgrade:

   a. Record the *ExtentSize* and *Selectivity* values of your current system. One way to do this is to open the Ens.MessageHeader class in Studio. *ExtentSize* and *Selectivity* are listed at the end of the class definition in the code window.

   b. If your values are significantly different than the new defaults, then after upgrading, either run Tune Table or use Studio or the Management Portal to manually update the *ExtentSize* and *Selectivity* values to describe your system.

   Important: You can run Tune Table against a running system as long as you select the **Keep class up to date** check box.

   For details on using Tune Table, see "ExtentSize, Selectivity, and BlockCount" in the chapter "Optimizing Tables" in the *Caché SQL Optimization Guide*.

6. Prepare for an upgrade to Ensemble as described in "Upgrading Caché" in the *Caché Installation Guide*.

7. Install Ensemble using the instructions in the appropriate platform-specific chapter of the *Caché Installation Guide*:

   • Installing Caché on Microsoft Windows

   • Installing Caché on UNIX® and Linux

   • Installing Caché on Mac

   Each chapter provides special instructions for upgrades.

8. If you exported custom schemas or any other production components or classes from a previous Ensemble installation, import them using Studio or the Management Portal. Be sure to import each item into the same namespace from which you exported it.

9. After upgrading Ensemble, you must recompile all objects that you have defined. One way to do this is to recompile all objects in namespaces that contain your custom code. To recompile all objects in a namespace, open a Terminal window and issue the following commands:

   ```
   ZN "namespace"
   DO $system.OBJ.CompileAll("u")
   ```

   This method both upgrades and then compiles the class dictionaries in the specified namespace.

   You could also use the **$SYSTEM.OBJ.CompileAllNamespaces()** method to recompile all namespaces, but you should not do this when upgrading a HealthShare installation. You should not recompile a namespace used for HealthShare.

Important: It is important to compile classes and schemas *before* compiling the DTL or BPL that uses them. In a complex application, you might have other dependencies to consider as well.

The recommendation to use **$SYSTEM.OBJ.CompileAll()** assumes that all information about dependencies between items being compiled is entirely contained within the declarations of the items themselves. You can establish such information by including compiler keywords such as DependsOn and CompileAfter. Alternatively, you can create routines to compile the code so that dependencies unavailable to the compiler are accounted for, especially in relation to schemas and domain-specific languages.

10. The Ensemble upgrade process reports important information regarding upgrades into a log file. After upgrading, review the contents of the ensinstall.log file in the \mgr subdirectory of your Ensemble instance to review the results of your upgrade.

11. Regenerate proxy classes if you previously generated them using the Java Gateway or one of the Caché language bindings, by following the instructions in the appropriate guides:

    • *Using the Java Gateway*

    • Other books in the *Caché Language Bindings* set

12. In some previous releases, in order to automatically start productions or execute code on production start, you inserted this code in the Caché user startup routine **^%ZSTART**. This code should now belongs in other locations; consequently, remove all Ensemble-related code from the user startup routine **^%ZSTART**. Use the following replacement mechanisms:

    • To automatically start productions, use the Ensemble production auto-start options instead of your custom code. For details, see "Managing Production Auto-Start" in *Managing Ensemble*.

    • Identify any code that you want to run for a production when it is started. Override the **OnStart()** callback method in the production class and place that code into this method.

    • Identify any code that is designed to act on specific business hosts upon production startup. Override the **OnProductionStart()** callback method in the corresponding business service, business process, or business operation class and place that code into this method.

    For information on these callback methods, see "Less Common Tasks" in *Developing Ensemble Productions*.

13. Re-enable the production auto-start options, as desired. See "Managing Production Auto-Start" in *Managing Ensemble*.

3.2 Upgrading from Ensemble 4.0 or 4.0.1

Upgrading from Ensemble 4.0 or 4.0.1 to the current release of Ensemble uses the same procedure as upgrading from later versions of Ensemble, but with some additional considerations. These are:

• The upgrade procedure deletes any HL7 custom schema definitions (*.HL7).

    CAUTION: To preserve such files, export them before proceeding with the upgrade. You may import them to any namespace after the upgrade is complete.

• For each Ensemble-enabled namespace, run these commands from your Terminal session:

```
ZN "nextEnsembleNamespace"
DO ##class(Ens.MessageHeader).%PurgeIndices()
DO ##class(Ens.MessageHeader).%BuildIndices()
DO ##class(EnsLib.HL7.Message).%PurgeIndices()
DO ##class(EnsLib.HL7.Message).%BuildIndices()
DO ##class(EnsLib.HL7.Message).%BuildIndices(($LB("Extent")))
```

These commands are required for the ENSDEMO and ENSEMBLE namespaces, and for any user-defined Ensemble-enabled namespaces.

3.3 Upgrading Ensemble Mirror Members

This section documents the steps to follow to upgrade an Ensemble mirror configuration. You should be familiar with standard Caché upgrade procedures. For general information on running Ensemble on a mirror, see "Ensemble Considerations for Mirroring" in the *Caché High Availability Guide* and for upgrade procedures, see "Minimum Downtime Upgrade with Mirroring" in the *Caché Installation Guide* for details.

Any upgrade involving a mirror includes two kinds of changes:

1. Changes that should occur on all mirror members, such as replacement of library code in CACHELIB and ENSLIB.

2. Changes that affect the mirrored data and should occur only once (on the primary failover member) because database access is read-only on the backup mirror members.

The Ensemble upgrade code is sensitive to this distinction and may need to make both kinds of changes during the course of an upgrade. It is important to note that the Ensemble upgrade code performs upgrades to Ensemble-enabled namespaces on a namespace-by-namespace basis, and checks whether each namespace contains mirrored data. If a namespace contains *any* mirrored data, then the "only once" steps only occur for that namespace when the upgrade code is running on the primary failover member. As such, it is crucial that the data Ensemble uses to determine whether a namespace has been upgraded is shared across the mirror.

The following sections describe the details of the data as well as the important issues and procedures necessary to upgrade the Ensemble instances in a mirror configuration:

- Important Mirror Global Data

- Mirror Considerations

- Performing the Mirror Upgrade

3.3.1 Important Mirror Global Data

The most important location for upgrade data which need to be shared is the *^Ens.Mirror* global located in each namespace that contains any mirrored data. The *^Ens.Mirror* global is used to determine which upgrade steps need to occur for that namespace. As such, you *must* ensure that each namespace containing mirrored data includes this global in the mirrored database(s). More specifically, for each namespace which contains mirrored data, you must map the *^Ens.Mirror* global in that namespace to a mirrored database so both members of the mirror can determine whether "only once" steps in the upgrade process for that namespace are complete.

You should also map the *^Ens.AutoStart* global to a mirrored database for each namespace containing mirrored data. The *^Ens.AutoStart* global specifies the name of the Ensemble production to start automatically after startup and, generally, it makes sense to have such an auto-start be identical across mirror members.

It may be simpler to illustrate what is meant by the above two directives using an example. Assume a given Ensemble namespace makes use of two user databases: DATA1 and DATA2. (The namespace also includes mappings to ENSLIB, but that is handled by Ensemble.) Assume DATA2 is a mirrored database, but DATA1 is not mirrored. In this situation, you must

explicitly map both *^Ens.AutoStart* and *^Ens.Mirror* to the mirrored DATA2 database so the contents of the globals are visible to both members of the mirror.

Before the upgrade commences, disable the auto-start of productions for namespaces containing mirrored data on both instances. If, as recommended, the *^Ens.AutoStart* global is mirrored across the two instances for a given namespace, then you only need to perform this step on the primary mirror member; otherwise, perform this step on both failover members. This is in keeping with the standard Ensemble upgrade procedure and prevents any productions from being started during the upgrade.

Note: Use the following commands to identify if the two important globals are mapped correctly for a given *NAMESPACE*:

```
Write ##class(SYS.Database).%OpenId($P(##class(%SYS.Namespace).GetGlobalDest
    ("NAMESPACE","Ens.Mirror"),"^",2)).Mirrored
Write ##class(SYS.Database).%OpenId($P(##class(%SYS.Namespace).GetGlobalDest
    ("NAMESPACE","Ens.AutoStart"),"^",2)).Mirrored
```

Important: In this case, do not use the system-wide configuration startup setting *EnsembleAutoStart* to turn off auto-start, because this is in the CPF file and is not mirrored. Instead, disable auto-start for each namespace, one at a time.

3.3.2 Mirror Considerations

Ensemble upgrades may require data to be moved to the %SYS namespace during the upgrade. Perform the steps on both nodes to ensure that both nodes are in a position to run all the upgrade steps if needed.

You should also be cognizant of other data or code that needs to be present on both machines, and whether any user application changes also need to occur.

3.3.3 Performing the Mirror Upgrade

"Minimum Downtime Upgrade with Mirroring" in the *Caché Installation Guide* describes the different types of mirror upgrades. If you are doing a major version upgrade and have Ensemble namespaces with mirrored data, you must choose one of the procedures that includes mirrored database changes.

4

Planning an Ensemble Server Deployment

This chapter describes the major issues you must account for when deploying a production Ensemble server. Although the preceding chapters of this document cover some installation and deployment topics, they are intended for installing a development version of Ensemble. While those topics cover the mechanics of installation and deployment they do not cover many issues that are critical to deploying Ensemble on a production server in a robust and secure manner.

If you are responsible for planning an Ensemble server deployment, this chapter can serve as a checklist of items to plan for, although there will be additional items you will need to consider as well.

This checklist identifies some critical issues that must be dealt with to deploy a reliable, efficient, and maintainable Ensemble system, but does not attempt to provide detailed guidance on these issues. This document organizes the issues into the following checklists:

- Capacity plan and checklist—ensures that the Ensemble server is able to efficiently handle your peak load.

- Robustness checklist—ensures that the Ensemble server has a high availability configuration and can recover from a disaster.

- Security checklist—ensures data privacy and resistance to attacks.

- Maintenance checklist—ensures that the Ensemble server continues to function well over long periods of time and with software and hardware updates.

4.1 Capacity and Performance Checklist

The performance of an Ensemble server is measured by its ability to handle the peak message load. The performance of an Ensemble server is dependent on the complex interaction between many components and settings. The load of an Ensemble server is dependent chiefly on:

- Number and size of messages—both the peak load and daily load are important.

- Processing required for each message—In most cases, you want to streamline the processing of messages. For example, while there are advantages to validating messages, complete validation can add a significant processing load to handling each message.

In many cases, the message load on an Ensemble system increases over time. This increase can be due to supporting more business functions in the Ensemble production or by an increase in business volume. The capacity of a server to handle this load is dependent on a complex interaction between many components and configuration settings including number of CPU cores, multiprocessor architecture, storage size and speed, network bandwidth and configuration, operating system buffer allocation, and Ensemble and Caché configuration. There is no simple formula that can predict the performance of

an Ensemble server because it is a complex interaction, but you can estimate, plan, prototype, and track performance to ensure that the Ensemble server is meeting your business needs.

To ensure that your Ensemble server has sufficient capacity and can efficiently handle its load, you should:

1. Estimate load—What are the number of messages that the Ensemble system will process? Is the load going to gradually increase after starting the server? How long do messages need to be preserved before they can be archived and removed from server storage?

2. Plan capacity—Planning skills depend on experience implementing similar Ensemble servers. If your organization does not have this experience, you should work with someone who has this experience: a consultant or an InterSystems Sales Engineer. You can contact InterSystems Worldwide Response Center (WRC) for a referral.

3. Prototype server and load testing—Once you have estimated load and planned the needed capacity, it is important to run a prototype system and monitor its performance. The prototype should confirm your capacity plan and provide you with a baseline to compare performance of the deployed system.

4. Plan disk layout for code and databases—By default all code and data created in an Ensemble namespace are stored in the same database file. By mapping data to multiple database files, you can gain more control over where the data is stored, which can help with performance of high end systems as well as making it easier to upgrade to new versions. It is also important to store journal files on a different disk than the database files to ensure a disk failure doesn't cause loss of data.

5. Deploy server—Install and configure your live system including any redundant failover machines.

6. Track load and performance—It is important to track the server performance to establish a baseline before there are any performance issues that need to be solved. You should collect metrics such as overall and peak message load, CPU utilization, disk free space, and average elapsed time to process a message.

7. Solve performance problems before they become critical—By tracking performance and forecasting growth, you should be able to plan upgrades and efficiency improvements before performance problems become major roadblocks in your organization's performance.

4.2 Robustness Checklist

Robustness is the ability of an Ensemble server to remain available and to be able to recover quickly from any disasters. Robustness is dependent on the following issues:

- Ensuring that the server has high availability. See the Caché High Availability Guide for more information.

- Backup of data so that it can be recovered and the server restarted in case of failure.

- Redundant network access so server can continue functioning if there is a network failure.

- Use a robust web server. Do not use the limited Apache server installed with Ensemble. It is provided as a convenience for use on development systems and is not a fully capable web server.

- Disaster recovery procedures and documentation.

4.3 Security Checklist

Security is the ability to control access to data and to protect the server from malicious attacks. In addition to maintaining privacy for business reasons, there are often privacy compliance requirements. Potential attacks can be aimed at gaining

access to confidential information, maliciously updating or deleting information, or compromising system performance. Security is dependent on the following issues:

- User accounts and password policies—Ensures that users who access the system are authenticated.

- Careful definition of permissions and roles—Ensure that users have the correct authorization and that they have access that they need, but not any greater access.

- Audit trail to track all configuration changes—Auditing provides a mechanism to track changes to the system that could potentially compromise security.

- Documentation that may be required to meet privacy compliance.

- User and operator security training—The most secure system can be compromised if users are not vigilant about security.

- Apply operating system and application security patches and upgrades in a timely manner.

- Control physical and network access to the server—Security requires robust firewalls, network protection, and limited physical access to server and network hardware.

- Database and journaling encryption—Although the firewall around a data center protects the security and integrity of the data, encrypting databases and journal files provides an extra level of security.

4.4 Maintenance Checklist

In addition to ensuring that after deploying an Ensemble server, it robustly and securely handles its load, you need to ensure that it continues to perform well over time. You need procedures to handle updates to software and hardware and how to respond to unexpected demands. Maintenance is dependent on the following issues:

- Regular message purging and backup—There are trade-offs between retaining messages after they have been processed so that they are available for viewing and purging messages to free storage for new messages.

- Backup and Restore—Perform regular backups and occasional testing of the restore from backup process.

- Hardware, software, and application updates—Plan to allow these updates without compromising system performance or security. Issues to consider include:

  - Schedule hardware maintenance, software patches and upgrades without losing server access at critical times.

  - Plan the deployment of components from a development system to a test environment, and finally to a live running production. This staging can include the use of system default settings, the export for deployment functionality, a source control system or all three. It is important to test the installation procedure as well as the updates on a test system before applying to the production server.

  - Source control provides a mechanism to control, monitor, and stage production upgrades. This is especially important where multiple people are updating related components, but is also often used as part of the promotion from development to production and for version control.

- Active monitoring procedures to detect any problems early—You should have defined procedures on how to respond to any potential problems discovered through monitoring. Monitoring can include:

  - Production monitoring—Operations staff should become familiar with the various monitoring screens

  - Enterprise monitoring—If you have multiple namespaces or systems, you can use the Enterprise monitor to provide an overview of how the overall system is performing. The Enterprise Message Bank and Enterprise Message Viewer provide a way to monitor messages from multiple productions.

– Alerts—Ensemble alerts can be used to quickly alert the right people to a failure without having operators monitoring a screen. However, generating too many alerts can be counterproductive and the right balance has to be found. Alert Management provides a mechanism to track resolution of alerts.

4.5 Other Resources

Although planning and deploying an Ensemble server is a challenging process, it is an easier process than trying to fix a critical Ensemble deployment that is in a troubled state. The Ensemble and Caché documentation and class library documentation provides detailed information on features and installation. The following are key documents for deploying a server:

- Monitoring Ensemble

- Managing Ensemble

- Caché Installation Guide

- Caché Monitoring Guide

- Caché Security Administration Guide

- Caché Distributed Data Management Guide

- Caché Data Integrity Guide

- Caché High Availability Guide

InterSystems provides the following resources to help you plan and deploy Ensemble:

- The InterSystems Worldwide Response Center (WRC) can provide guidance on deploying Ensemble servers and can connect you with additional resources when needed. To access WRC Direct, go to: http://wrc.InterSystems.com and enter your username and password. Contact the WRC (support@intersystems.com or +1.617.621.0700) for a username and password if you do not already have them.

- InterSystems Learning Services provides classes on Ensemble.

- The InterSystems Developer Connection and support communities provide a way for you to get your questions answered by InterSystems employees and by other InterSystems customers who may have experienced similar issues.