



Ensemble Release Notes

Version 4.0
27 July 2006

Ensemble Release Notes

Ensemble Version 4.0 27 July 2006

Copyright © 2006 InterSystems Corporation.

All rights reserved.

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.



The Caché product and its logos are registered trademarks of InterSystems Corporation.



The Ensemble product and its logos are registered trademarks of InterSystems Corporation.



The InterSystems name and logo are trademarks of InterSystems Corporation.

This document contains trade secret and confidential information which is the property of InterSystems Corporation, One Memorial Drive, Cambridge, MA 02142, or its affiliates, and is furnished for the sole purpose of the operation and maintenance of the products of InterSystems Corporation. No part of this publication is to be used for any other purpose, and this publication is not to be reproduced, copied, disclosed, transmitted, stored in a retrieval system or translated into any human or computer language, in any form, by any means, in whole or in part, without the express prior written consent of InterSystems Corporation.

The copying, use and disposition of this document and the software programs described herein is prohibited except to the limited extent set forth in the standard software license agreement(s) of InterSystems Corporation covering such programs and related documentation. InterSystems Corporation makes no representations and warranties concerning such software programs other than those set forth in such standard software license agreement(s). In addition, the liability of InterSystems Corporation for any losses or damages relating to or arising out of the use of such software programs is limited in the manner set forth in such standard software license agreement(s).

THE FOREGOING IS A GENERAL SUMMARY OF THE RESTRICTIONS AND LIMITATIONS IMPOSED BY INTERSYSTEMS CORPORATION ON THE USE OF, AND LIABILITY ARISING FROM, ITS COMPUTER SOFTWARE. FOR COMPLETE INFORMATION REFERENCE SHOULD BE MADE TO THE STANDARD SOFTWARE LICENSE AGREEMENT(S) OF INTERSYSTEMS CORPORATION, COPIES OF WHICH WILL BE MADE AVAILABLE UPON REQUEST.

InterSystems Corporation disclaims responsibility for errors which may appear in this document, and it reserves the right, in its sole discretion and without notice, to make substitutions and modifications in the products and practices described in this document.

Caché, InterSystems Caché, Caché SQL, Caché ObjectScript, Caché Object, Ensemble, InterSystems Ensemble, Ensemble Object, and Ensemble Production are trademarks of InterSystems Corporation. All other brand or product names used herein are trademarks or registered trademarks of their respective companies or organizations.

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

1 Getting Started	1
1.1 First-Time Users	1
1.2 Installing Ensemble	1
1.2.1 Supported Platforms	2
1.2.2 Installation Procedures	2
1.2.3 OpenVMS Considerations	2
1.2.4 UNIX Considerations	3
1.2.5 Failover Clusters	4
1.3 Upgrading Ensemble	4
1.3.1 Existing Productions	4
1.3.2 Business Rules	5
1.3.3 Java Gateway	5
1.4 About the Release	6
2 New Features	7
3 Enhancements	9
3.1 Namespaces	9
3.2 Default Login	11
3.3 Command Line	11
3.4 Management Portals	11
3.5 BPL Syntax	13
3.5.1 Message Header Description Property	13
3.5.2 <call> Element timeout Attribute	14
3.5.3 <branch> and <label> Elements	14
3.5.4 <milestone> Element	15
3.6 BPL Visual Editor	15
3.7 BPL Components	16
3.7.1 The component Attribute	17
3.7.2 The IsComponent Method	17
3.7.3 Call Wizard and Components	17
3.7.4 View Target Definition	18
3.7.5 Show Components	19
3.7.6 Add Business Process Component	19
3.8 Utility Functions	19

3.9 DTL Visual Editor	20
3.9.1 DTL Wizards	20
3.9.2 Test Transformation	20
3.9.3 Utility Functions	21
3.10 Adapters	21
4 Documentation	23
4.1 Documentation Changes	23
4.2 Online Documentation	24
4.3 Printed Documentation	25
4.3.1 Ensemble Documentation	25
4.3.2 Other InterSystems Documentation	25
5 Known Problems	27
5.1 Enhancements in English Only	27
5.2 Importing 3.1 Business Rules into 4.0	27
6 Release History	29
6.1 Ensemble 1.0	29
6.2 Ensemble 2.0	30
6.3 Ensemble 2.1	30
6.4 Ensemble 3.0	31
6.5 Ensemble 3.1	31
6.6 Ensemble 4.0	32

1

Getting Started

Welcome and thank you for using Ensemble 4.0!

This book provides important tips for getting started with the new release.

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.

1.1 First-Time Users

For an overview of product features, see the book *Introducing Ensemble*.

To prepare to work on an Ensemble project, see the book *Getting Started with Ensemble*. It outlines development tasks and identifies sources of information.

1.2 Installing Ensemble

This topic describes how to install Ensemble 4.0.

To upgrade from previous releases of Ensemble, see the next topic, [Upgrading Ensemble](#).

Important: If at any time you need assistance with Ensemble installation, contact the [InterSystems Worldwide Response Center](#).

1.2.1 Supported Platforms

To double-check the list of supported platforms, browsers, web servers, and other technologies for use with InterSystems products, see the *Caché Supported Platforms* book provided with the Ensemble kit.

1.2.2 Installation Procedures

For installation instructions, see the *Caché Installation Guide* for Windows, OpenVMS, UNIX, or Macintosh. In general, the only difference for Ensemble is that, where the Caché installation sequence offers a choice between 8-bit and Unicode, Ensemble automatically performs a Unicode installation.

There are additional considerations on various platforms as described below:

- [OpenVMS](#)
- [UNIX](#)
- [Failover clusters](#)

1.2.3 OpenVMS Considerations

If you are running the Ensemble 4.0 server on OpenVMS, extra steps are required. The following list provides an overview. InterSystems suggests that you use the instructions in the *Caché Installation Guide* chapter *Installing Caché on OpenVMS*. Pay special attention to the *Post-Installation Tasks* topic, which guides you to the additional references listed below:

- When you install the Ensemble 4.0 server on OpenVMS, you must install an Ensemble 4.0 client (or a server, which includes the client) on a Windows system so that you can use browser-based interfaces such as the System Management Portal, Ensemble Management Portal, BPL Visual Editor, and DTL Visual Editor.

For instructions, see the *Caché System Administration Guide* chapter *Using Caché on OpenVMS*, topic *Accessing the System Management Portal*.

- There is no internal or private Web server installed with Ensemble 4.0 on OpenVMS. Therefore, in order to use the System Management Portal and the Ensemble Management Portal, you must configure an external Web server.

Follow the instructions in the *Caché Installation Guide* chapter *Installing Caché on OpenVMS*, topic *Install Caché on a Web Server*.

- After completing the Caché procedure, for Ensemble 4.0 on OpenVMS you must complete configuration of the external Web server with the following steps:

- Copy all files and subdirectories found in this Ensemble server directory:

```
[installDir.CSP.ENSEMBLE]
```

To the physical directory on the Web server that corresponds to this virtual directory:

```
/csp/ENSEMBLE
```

- Copy all files and subdirectories found in this Ensemble server directory:

```
[installDir.CSP.ENSDEMO]
```

To the physical directory on the Web server that corresponds to this virtual directory:

```
/csp/ENSDEMO
```

Detailed background information about CSP, remote servers, and physical and virtual directories is available in the *Caché System Administration Guide* chapter Connecting to Remote Servers, topic Using Caché Server Pages with a Remote Web Server.

- Each time you create a new namespace on OpenVMS, the namespace is automatically Ensemble-enabled, but Ensemble user interfaces will not work properly in that namespace until you copy files and directories as described above for ENSEMBLE and ENSDEMO.

For each new namespace, copy from this Ensemble server directory:

```
[installDir.CSP.newNamespace]
```

To the physical directory on the Web server that corresponds to this virtual directory:

```
/csp/newNamespace
```

1.2.4 UNIX Considerations

If you are running the Ensemble 4.0 server on UNIX, InterSystems suggests that you use the instructions in the *Caché Installation Guide* chapter Installing Caché on UNIX and Linux, paying special attention to the Post-Installation Tasks.

When you install the Ensemble 4.0 server on UNIX, you must install an Ensemble 4.0 client (or a server, which includes the client) on a Windows system so that you can use browser-based interfaces such as the Ensemble Management Portal, BPL Visual Editor, and DTL Visual Editor. For instructions, see the *Caché Installation Guide* chapter Installing Caché on UNIX and Linux, topic Install Caché Client on Windows for Development.

1.2.5 Failover Clusters

The *Caché High Availability Guide* explains how to install Ensemble 4.0 on multiple clustered machines to provide failover capabilities in case of problems on the primary server. Chapters include:

- Cluster Management
- Cluster Journaling
- Caché Clusters on Tru64 UNIX
- Caché and Windows Clusters
- ECP Failover

1.3 Upgrading Ensemble

To upgrade an existing Ensemble 3.1 installation to Ensemble 4.0, use the procedure for upgrading Caché 5.0.x to Caché 5.2 as described in the *Caché Installation Guide* chapter Upgrading Caché.

There are some upgrade considerations that apply to Ensemble independently of Caché. Please read through all of the information in this topic to understand these considerations before beginning the upgrade procedure.

Important: The supported upgrade path is from Ensemble 3.1 to Ensemble 4.0 only. If you have an Ensemble version *other than 3.1* and you want to upgrade it to the released Ensemble 4.0 product, contact the [InterSystems Worldwide Response Center](#) for assistance.

1.3.1 Existing Productions

Before beginning the upgrade procedure, use Studio to export all user-defined production elements from your Ensemble 3.1 installation to XML. Be sure that your export operation(s) include all of the following user-defined production elements:

- Class code (*.cls)
- Business rules (*.rul)

- HL7 custom schema definitions (*.hl7)

After you install Ensemble 4.0, import the Ensemble 3.1 classes, rules, and schema definitions into an [Ensemble-enabled namespace](#) in the new installation. Be sure to recompile everything in Ensemble 4.0.

1.3.2 Business Rules

If you used business rules in Ensemble 3.1, consult the [Known Problems](#) chapter later in this book. It describes how to import Ensemble 3.1 business rules into Ensemble 4.0.

1.3.3 Java Gateway

If you used the Java Gateway in Ensemble 3.1, you must address the following issues:

- In previous releases, the physical location of CacheDB.jar was:

```
installDir\Dev\Java\Lib\CacheDB.jar
```

where *installDir* was the directory into which you installed Ensemble.

The Ensemble 4.0 location for CacheDB.jar is different. This change requires you to reconfigure the classpath for any Java Gateway services within any upgraded productions. The new classpath must reflect the new location, which on Windows is:

```
installDir\Dev\Java\Lib\version\CacheDB.jar ; .
```

Where:

- *installDir* is the directory into which you installed Ensemble
- *version* is either JDK14 or JDK15, depending on the Java version you are using.

On UNIX, the Ensemble 4.0 classpath is similar but with different case:

```
installDir/dev/java/lib/version/CacheDB.jar ; .
```

- Ensemble proxy classes generated from a 3.1 system are obsolete within a 4.0 deployment. These proxy classes must be deleted and regenerated after upgrading Ensemble to 4.0.

The fastest way to regenerate proxies is Using the Java Gateway Wizard as described in the book *Using the Java Gateway*.

1.4 About the Release

The remaining chapters in this book summarize the contents of the Ensemble 4.0 release in terms of changes since Ensemble 3.1:

- [New Features](#)
- [Enhancements](#)
- [Documentation](#)
- [Known Problems](#)
- [Release History](#)

2

New Features

Ensemble 4.0 runs on top of Caché 5.2. This new pairing of InterSystems products brings to Ensemble 4.0 a number of Caché 5.2 features that were not available in Ensemble 3.1.

IF you are upgrading from...	THEN the underlying changes resemble...
Ensemble 3.0 to Ensemble 4.0	Caché 5.0.5 to Caché 5.2
Ensemble 3.1 to Ensemble 4.0	Caché 5.0.16 to Caché 5.2

The range of benefits from this change is extremely broad. New features include security, .NET, and Java bindings, just to name a few.

To learn about the Ensemble changes relating to Caché, begin with the InterSystems online documentation set, *Getting Started with Caché*. All of the following books from this set are required reading when you upgrade from Ensemble 3.1 to Ensemble 4.0:

- *Caché 5.1 Release Notes*
- *Caché 5.2 Release Notes*
- *Caché 5.1 Upgrade Checklist*
- *Caché 5.2 Upgrade Checklist*

3

Enhancements

Ensemble 4.0 offers enhancements to the following existing features:

- [Namespaces](#)
- [Default Login](#)
- [Command Line](#)
- [Management Portals](#)
- [BPL Syntax](#)
- [BPL Visual Editor](#)
- [BPL Components](#)
- [Utility Functions](#)
- [DTL Visual Editor](#)
- [Adapters](#)

3.1 Namespaces

In previous releases, the Ensemble installation sequence created one **Ensemble-enabled namespace** — a namespace that provided Ensemble classes in the packages Demo, Ens, EnsLib, and CSPX. That namespace had the name ENSEMBLE by default, but you could give it any name you preferred during installation. If you wanted additional Ensemble-enabled

namespaces, you could create them by restarting the installation program and choosing the option **Load Ensemble classes into a namespace**.

Ensemble 4.0 installation provides three Ensemble-enabled namespaces: ENSLIB, ENSDEMO, and ENSEMBLE. These namespaces are organized as follows:

- ENSLIB provides a library of all Ensemble classes in the packages Ens, EnsLib, and CSPX. The contents of ENSLIB are mapped to any Ensemble-enabled namespace. Because these classes and data are shared among all Ensemble-namespaces, any change in them will be reflected in all namespaces.

CAUTION: ENSLIB is for InterSystems use only; it is not a valid namespace for work. InterSystems strongly recommends that you *do not* edit classes or perform any work in the ENSLIB namespace.

- ENSDEMO provides the Ensemble sample production classes in a package called Demo. When you want to view the Ensemble sample code in Studio, you must change to the ENSDEMO namespace.
- ENSEMBLE is where Ensemble users do all of their programming.

Important: InterSystems recommends that, when programming with Ensemble, you *do not* use the package names Demo, Ens, EnsLib, CSPX, or any class name that already exists in these packages. Any of these actions may cause your work to be deleted the next time you upgrade Ensemble.

Unlike previous releases, Ensemble 4.0 does not allow you to choose another name for the ENSEMBLE namespace while installing Ensemble. However, you can create additional Ensemble-enabled namespaces from the new [System Management Portal](#). Navigate to the **[Home] > [Configuration] > [Namespaces]** page and choose **Create New Namespace**. The namespace you create is automatically Ensemble-enabled.

There may be a delay after you create a new namespace before it is available to be viewed in the Ensemble Management Portal. This delay occurs while the portal pages are copied and compiled in the new namespace.

Important: If your Ensemble server runs on OpenVMS, you must complete configuration of a new namespace by copying files and directories as described in the [Installing Ensemble](#) chapter, [OpenVMS Considerations](#) topic.

3.2 Default Login

The default login for any namespace is the username `_SYSTEM` with a password of `SYS`. Formerly this default password was `sys`. As an Ensemble developer, you will encounter this login most often when starting Studio.

CAUTION: To protect the security of your system, at the first opportunity you should change the default password for the `_SYSTEM` account in each namespace.

3.3 Command Line

If you have been controlling Ensemble from the operating system command line, you need to be aware that the commands `enscontrol` and `enssession` have changed.

The commands work exactly as they did in the past, except that:


- `enscontrol` is now `ccontrol`
- `enssession` is now `cssession`

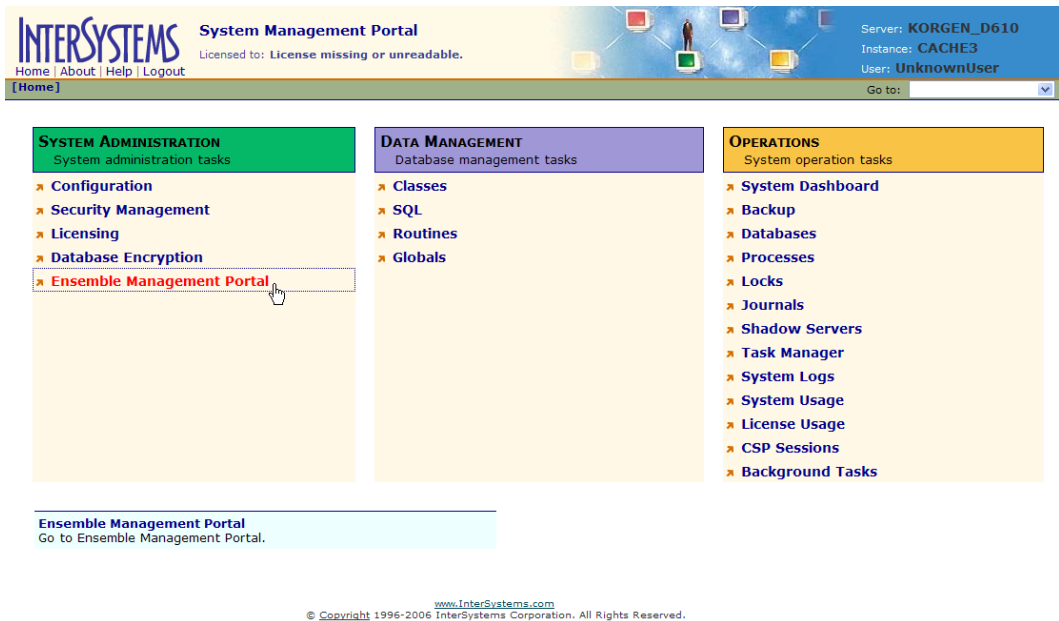
3.4 Management Portals

Ensemble 4.0 offers an entirely new user interface for administration tasks.

The Terminal and Studio tools that you used for application development are still present. However, the InterSystems Explorer, Configuration Manager, SQL Manager, and Control Panel tools are gone. A single, unified, graphically-oriented System Management Portal replaces them.


Note: For details, see the *Caché System Administration Guide*, chapter Using the System Management Portal.

You can start the new portal from the  Ensemble cube in the Windows system tray. Choose the **System Management Portal** option. The System Management Portal home page displays:



The System Management Portal *does not replace* the Ensemble Management Portal.

The Ensemble Management Portal is unchanged since Ensemble 3.1, but the Ensemble 4.0 version uses a different stylesheet. Some of the background colors have changed. This will become clear to you as you begin using the portal.

You can no longer start the Ensemble Management Portal from the  Ensemble cube.

Ensemble 4.0 offers two ways to start the Ensemble Management Portal:

- Start the System Management Portal. On the home page, at the bottom of the **System Administration** column on the left-hand side, click the **Ensemble Management Portal** option. (The above figure highlights this choice.) The login screen displays.
- Start Studio. Change to an [Ensemble-enabled namespace](#). From the **Utilities** menu, choose **Ensemble Management**. The Ensemble Management Portal login screen displays.

After you log into the Ensemble Management Portal, its Home page displays. A new field has been added to the title bar on this page. You can see this field in the figure below. The **Namespace** drop-down list at upper right allows the user to switch to a different Ensemble-enabled namespace from the Home page.

Important: Only the Home page offers this feature; other Ensemble Management Portal pages simply identify the currently-active namespace in the **Namespace** field.

3.5 BPL Syntax

The Ensemble Business Process Language (BPL) offers the following enhancements.

3.5.1 Message Header Description Property

The Ensemble message header object now includes a property that contains an optional text description of the message. BPL business processes provide text in this property automatically, based on the type of BPL activity that generated the message object. In the case of <call>, <sync>, and <delay> elements, this text is the value of the *name* attribute for the element.

3.5.2 <call> Element timeout Attribute

The <call> element now includes an optional *timeout* attribute. This allows a timeout to be set on a synchronous call. The <call> *timeout* value is used only when the <call> *async* attribute is set to 0 (false).

If the call is required to be asynchronous, then the <call> *async* value is set to 1 (true). In this case, the only way to set a timeout period is to set the *timeout* attribute on the <sync> element that is being used to collect the asynchronous response(s) from the call.

3.5.3 <branch> and <label> Elements

The new <branch> element causes an immediate change in the flow of execution if the value of its *condition* expression is true. Control passes to the new <label> element, whose name is specified as the value of the *label* attribute in the <branch>.

In the following BPL example, if the *condition* expression is true, the flow of control shifts directly from the <branch> with the *label* value TraceSkipped to the <label> with the *name* value TraceSkipped, while the intervening <trace> element is ignored:

```
<branch condition="myVar='1'" label="TraceSkipped" />  
<trace value="Ignore me when myVar is 1..." />  
<label name="TraceSkipped" />
```

If the <branch> *condition* expression is false, control simply passes to the next BPL statement following the <branch>, in this case <trace>.

A destination <label> must be in the same scope as the <branch> that references it. Thus:

- Each <sequence> element within a <flow> has its own <label> scope. The BPL execution engine prevents any attempt to <branch> to a <label> outside the current <sequence> container.
- There are similar restrictions on any other BPL container element that controls the flow of execution at runtime. Each container has its own <label> scope.

In addition to these restrictions, each <label> *name* value must be unique across the entire BPL business process, not just within the current scope.

CAUTION: As is true in all programming languages, the BPL branch mechanism must be used with care. The BPL editor does not prevent basic programming mistakes such as infinite loops or invalid branch cases.

3.5.4 <milestone> Element

The new <milestone> element writes a message to the Ensemble database for later retrieval. <milestone> works very much like <trace>. BPL provides a <milestone> element in addition to <trace> so that the programmer can distinguish important messages that need to be stored and accessed apart from <trace> messages.

Often a programmer uses <trace> messages for diagnostic purposes, whereas <milestone> messages can be helpful to track the progress of a correctly-operating, long-running business process.

3.6 BPL Visual Editor

The BPL Visual Editor has been modified to support the above syntax changes.

When you right-click on a BPL diagram in Studio, the BPL Visual Editor offers a context-sensitive menu of useful options that includes **Add Activity**, **Zoom**, and many others as described in the book *Developing Ensemble Productions*, chapter Business Processes, topic BPL Diagram Context Menu.

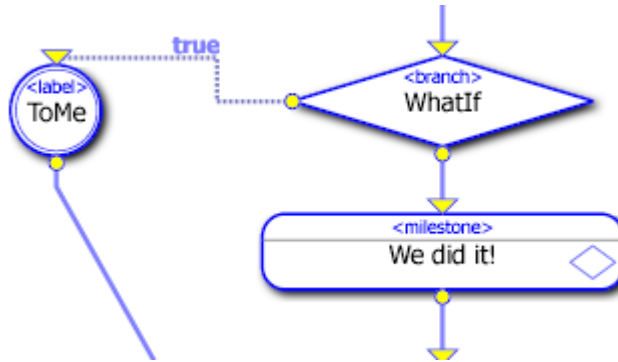
All three of the new elements — <branch>, <label>, and <milestone> — can be created using the **Add Activity** context menu option. Look for <branch> and <label> at the bottom of the **Add Activity** list, grouped with the control activities <if> and <switch>.

As described in the previous topic, the new <branch> element is a decision activity with two possible outcomes. In the BPL diagram, a <branch> element has two output connections:

- The main output is a solid line connecting to the next activity in the current sequence. This arrow indicates the flow of execution when the <branch> condition evaluates to false.
- The other output is a dotted line connecting to a <label> activity somewhere within the current sequence. This arrow indicates the flow of execution when the <branch> condition evaluates to true. If the <label> referenced by the <branch> is on a different page of the diagram, the <branch> shows a short, truncated connector.

When editing a diagram with the BPL Visual Editor, you can connect a <branch> to a <label> by dragging a line from the output of the <branch> to the input of the <label> (but not the other way). You can also use the Studio Inspector window to set the *label* attribute of the <branch> to the name of the <label>.

The visual appearance of new additions to the BPL diagrams is as follows:



A new option on the BPL Visual Editor context menu is the **Call Wizard**, which appears just after **Add Activity** near the top of the menu. When you select **Call Wizard**, Studio displays a dialog whose drop-down choices help you to properly construct a new <call> to one of the available business processes or business operations in the production. The **Help** button in this dialog displays the online documentation for the BPL <call> element.

For details about the **Call Wizard**, see the next topic.

3.7 BPL Components

Ensemble 4.0 now supports the concept of a “reusable component.”

A **business process component** or **BPL component** is a BPL business process that a programmer wishes to identify as a modular, reusable sequence of steps in the BPL language. A BPL component is analogous to a function, macro, or subroutine in other programming languages.

Only another BPL business process can call a BPL component. It does this using the BPL <call> element. The BPL business process component performs tasks, then returns control to the BPL business process that called it.

The Ensemble architecture has always allowed one BPL business process to call another BPL business process. The difference in Ensemble 4.0 is that the optional “component” designation allows you to classify certain BPL business processes as simpler, lower-level components that:

- Are not intended to run as stand-alone business processes (although nothing in the architecture prevents this)

- May be reusable (in the sense of a function, macro, or subroutine)

Business processes that are not “components” are assumed to have more complex, special-purpose designs, and to operate at a higher conceptual level than components. It is expected that BPL non-components will call BPL components to accomplish tasks.

Important: There is no requirement that you use the “component” designation for any BPL business process. It is available as a convenience for any BPL programmer who prefers it.

The remainder of this topic describes the specific enhancements that Ensemble 4.0 provides to support BPL components.

3.7.1 The component Attribute

The BPL `<process>` element, the top-level container for all BPL elements, offers a new optional attribute which:

- Is called *component*
- Is of type Boolean
- Has the default value 0 (false)

3.7.2 The IsComponent Method

Ens.BusinessProcessBPL, the base class for all BPL business processes, offers a new class method which:

- Is called **IsComponent()**
- Returns type Boolean
- Returns 1 (true) if the BPL business process is a component, 0 (false) if not

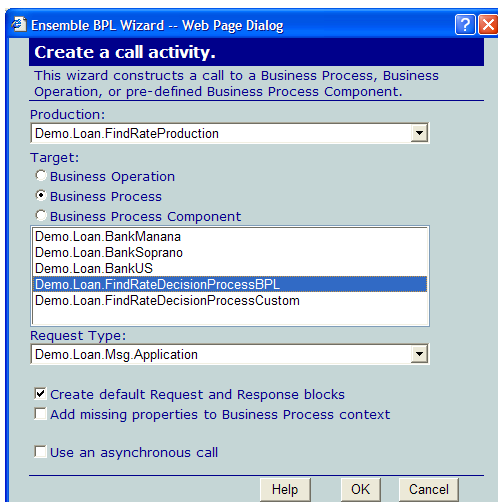
3.7.3 Call Wizard and Components

In previous releases, a BPL business process invoked the `<call>` element to send an Ensemble message to a business operation or to another business process. In Ensemble 4.0, this is still true. However, a called business process may or may not be a “component” business process as described in this topic.

The BPL Visual Editor context menu offers a new option, **Call Wizard**, that assists in creating valid <call> elements for BPL business processes. The <call> wizard supports the new BPL component feature by offering the following choices:

- Business Operation
- Business Process
- Business Process Component

Depending on which you choose, the dialog displays a list of the available items of that type; you may select one of these as the **Target** of the <call>. For full instructions, see *Developing Ensemble Productions*, chapter Business Processes, topic Using the Call Wizard.



3.7.4 View Target Definition

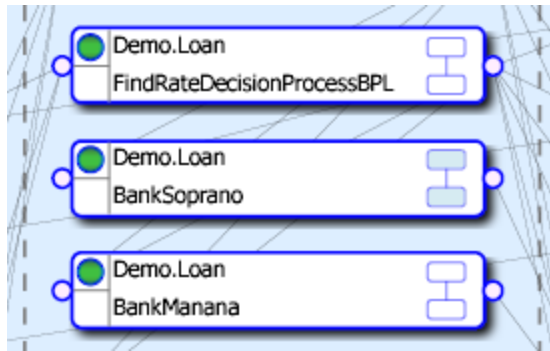
The BPL Visual Editor context menu offers a new option, **View Target Definition**. To use this option, select a <call> activity that calls another BPL business process. Right-click, then choose **View Target Definition** from the context menu. The BPL diagram for the referenced BPL business process displays in a separate Studio window. You can switch back to the diagram of the calling business process by selecting its name in the Studio **Window** menu.

Note: When you first open a BPL diagram and choose **View Target Definition**, a message displays asking you to use the **Call Wizard** to set the production name. Choose the context menu **Call Wizard** option, click on the production name that contains the BPL diagram, and then click **Cancel**.

3.7.5 Show Components

As a convenience in viewing production diagrams, the Ensemble Management Portal Productions page offers a new context menu option, **Show Components**. This option toggles between a production diagram that shows all types of BPL business processes, and a view that hides any BPL business processes that are components.

In the BPL diagram, components have a slightly different icon from regular BPL business processes. The blocks in the icon are filled rather than outlined, as in the center item below:



3.7.6 Add Business Process Component

When you add a business process to a production using the Host Wizard in the Ensemble Management Portal, there is a new option, **Business Process Component**, that offers choices of only those BPL business processes that have been designated as components. For regular BPL business processes use the **Other** option.

3.8 Utility Functions

In previous releases, a limited number of utility functions were available for use in business rules. These included mathematical or string processing functions such as you may be accustomed to using in other programming languages.

Ensemble 3.1 provided the utility functions *Min*, *Max*, *Round*, and *SubString*. Ensemble 4.0 adds more utility functions and makes the full set of functions available in both business rules and DTL data transformations.

The business rule syntax for utility functions differs from the DTL syntax in the following significant way:

- Business rules reference the utility functions simply by name:

```
ToUpper (value)
```

- DTL uses two leading dots on the function name, as if the function were a method:

```
..ToUpper (value)
```

The book *Developing Ensemble Productions*, chapter Ensemble Productions, topic Utility Functions lists the available functions and shows the proper syntax for using them in business rules or data transformations. It also explains how to extend the list of built-in utility functions by adding custom code.

3.9 DTL Visual Editor

In Ensemble 4.0, the DTL Visual Editor offers expanded convenience features. For details about any of these new features, see the book *Developing Ensemble Productions*, chapter Data Transformation, Abstraction, and Persistence, topic Using the DTL Visual Editor.

3.9.1 DTL Wizards

In Ensemble 4.0, the DTL Visual Editor provides an improved set of wizards to assist DTL programmers in the construction of valid `<assign>`, `<if>`, and `<foreach>` statements.

3.9.2 Test Transformation

Ensemble 4.0 offers a new option on the DTL Visual Editor context menu called **Test Transformation**. When you right-click to reveal the context menu and select **Test Transformation**, Studio displays a dialog with an **Input Message** at the top and an **Output Message** at the bottom.

Initially the **Output Message** window is blank and the **Input Message** window contains a text skeleton in a format appropriate to the source message:

- For HL7 or other EDI messages, the window displays raw text
- For regular Ensemble messages, the window displays an XML skeleton with an entry for each of the properties in the message object.

Enter data in the **Input Message** window and click **Test**. The **Output Message** window displays the results of the data transformation. You may repeat these steps as needed.

3.9.3 Utility Functions

As explained in the [Utility Functions](#) topic above, Ensemble 4.0 now provides functions for use in DTL data transformations. Previously, these functions were available in business rules only.

3.10 Adapters

The list of built-in adapters is the same for Ensemble 4.0 as for Ensemble 3.1, except that an outbound LDAP adapter is now provided.

Timestamp formats for input and output filenames have changed from their Ensemble 3.1 conventions. For the Ensemble 4.0 conventions see the book *Using File Adapters with Ensemble*, appendix Timestamps in Filenames.

In Ensemble 4.0, the class `EnsLib.TCP.InboundAdapter` is abstract. Prior to this release, the class was not abstract. Now, you cannot use `EnsLib.TCP.InboundAdapter` as the adapter for a business service unless you subclass it. See the book *Using TCP Adapters with Ensemble*.

4

Documentation

Ensemble 4.0 provides documentation in HTML format and in Adobe Page Description Format (PDF). The [HTML files](#) can be served up any time Ensemble is running. The [PDF files](#) can be viewed or printed as you choose. In either format, the contents are identical.

4.1 Documentation Changes

Ensemble 4.0 introduces the following new books:

- *Ensemble 4.0 Release Notes* (this book)
- *Using File Adapters with Ensemble*
- *Using TCP Adapters with Ensemble*

Ensemble 4.0 updates the following existing books:

- *Developing Ensemble Productions*
- *Managing Ensemble Productions*
- *Ensemble Business Process Language Reference*
- *Ensemble Data Transformation Language Reference*

New technical material for Ensemble 4.0 is concentrated in the following books and chapters:

- *Using File Adapters with Ensemble* (entire book)
- *Using TCP Adapters with Ensemble* (entire book)


Ensemble 4.0 deletes the following book:

- *Ensemble Installation Guide* — The *Caché Installation Guide* now covers these topics.

4.2 Online Documentation

When the local Ensemble server is online, it serves documentation in HTML format via a Web browser. The online documentation system gives access to information about core InterSystems technologies and the InterSystems products Ensemble and Caché.



You can access the online documentation in any of the following ways:

- Find the  Ensemble cube in the Windows system tray. Click on it to display the Ensemble cube menu. If Ensemble is stopped, start it. Then choose **Documentation** from the menu.
- Entering the following URL into a browser page on the local Ensemble system, where 1972 is the port number for which your Ensemble server is configured:

```
http://localhost:1972/csp/docbook/DocBook.csp
```

- When running Ensemble Studio, choose the **Help** menu **On-line Documentation** option.

The online documentation home page lists information organized by topic and provides navigational aids that you can use to find information easily. For example:

Link or Icon	Purpose
	View an alphabetical index of topics
	Perform a keyword search across the entire formal documentation set
<i>Class Reference Information</i>	View class documentation

4.3 Printed Documentation

InterSystems strongly recommends that if you want printed documentation, you *do not* print HTML pages from the browser, but instead take advantage of the pre-formatted PDF files, whose content is identical to the HTML.

4.3.1 Ensemble Documentation

The PDF files for Ensemble books are found in a subdirectory under the directory where you installed Ensemble, usually C:\EnsembleSys\Docs\Ensemble*.pdf or similar.

4.3.2 Other InterSystems Documentation

Other InterSystems documents that appear in the Ensemble online documentation set are also available in PDF format. If you discover there is an InterSystems document that you want, and that does not appear in your Ensemble kit, contact the [InterSystems Worldwide Response Center](#).

5

Known Problems

This topic describes the known software problems in the Ensemble 4.0 release.

Important: For an updated list of known problems in this release, contact the [InterSystems Worldwide Response Center](#).

5.1 Enhancements in English Only

The Ensemble 4.0 [enhancements](#) to existing features are available in English only. The underlying Caché 5.2 is fully localized as described in the *Caché Supported Platforms* topic Supported Languages.

5.2 Importing 3.1 Business Rules into 4.0

If you have used previous Ensemble releases, you know that you can use Studio commands to export all or part of a production to an XML file, then import this XML representation back into Ensemble at some future time. This XML can include business rule definitions.

In Ensemble 4.0, when you import a production that was exported from Ensemble 3.1, there is a problem with the business rules in the XML file. If the result of any rule in any rule set is an “assign” action, the Studio import operation fails with an “unresolved type” error message.

To work around this problem, start a Terminal session in your Ensemble 4.0 instance and change to the desired namespace. Import the Ensemble 3.1 XML file by typing the following command at the Terminal prompt. In place of `C:\temp\ThisProductionUsesRules.xml` in the example, type the full pathname of the XML file:

```
Do $system.OBJ.Load("C:\temp\ThisProductionUsesRules.xml", "/checkschema=0")
```

6

Release History

The Ensemble product was developed to meet the need for a comprehensive, high-performance enterprise application and data integration platform with tightly integrated development, management, and supervisory tools.

A brief history of Ensemble releases is outlined below. If you are viewing this document online, you can display a description of a feature by clicking on its name in the list.

6.1 Ensemble 1.0

Ensemble 1.0 was released to InterSystems customers in 2002.

This release introduced the following features:

- Application Integration
- Data Integration
- Data Abstraction
- Persistence Engine
- Storage Engine
- SQL Gateway
- Studio

6.2 Ensemble 2.0

Ensemble 2.0 was released to InterSystems customers in August 2003.

This release enhanced existing features and introduced the following new features:

- Universal Services Architecture
- Business Hosts
- Messaging Engine
- Adapter Library
- Business Process Language
- Data Transformation Language
- Management Portal
- Message Visual Trace
- Code Generation Wizards
- BPL Visual Editor
- DTL Visual Editor
- Event Log
- Monitoring Service
- Testing Service
- Documentation
- Sample Code

6.3 Ensemble 2.1

Ensemble 2.1 was released to InterSystems customers in November 2003.

This release enhanced the following existing features:

- Business Process Language
- BPL Visual Editor

- Management Portal
- Adapter Library

6.4 Ensemble 3.0

Ensemble 3.0, November 2004, was the first public release of Ensemble.

The essentials of the release are described in *Introducing Ensemble*

Ensemble 3.0 introduced the following new features:

- Business Activity Monitoring — see *Using Dashboards with Ensemble*
- Workflow Management — see *Using Workflow with Ensemble*
- Business Rules — see *Using Business Rules with Ensemble*
- Java Gateway — see *Using the Java Gateway*

Ensemble 3.0 enhanced the following existing features:

- Business Hosts
- Business Process Language
- BPL Visual Editor
- Management Portal

6.5 Ensemble 3.1

Ensemble 3.1 was released in April 2006.

Ensemble 3.1 introduced the following new features:

- HL7 Support — see the *Ensemble HL7v2 Development Guide*, *User Interface Guide*, and *Technical Reference*; also the *Ensemble HL7v3 Development Guide*.

Ensemble 3.1 offered the following major enhancements:

- Message Contents

- [How Data is Purged](#)
- [Business Process Context](#)
- [Business Process Actor Pools](#)
- [Business Hosts](#)
- [Ensemble Management Portal](#)
- [Configuration Page](#)
- [Dashboards](#)
- [Business Rules](#)
- [Java Gateway](#)
- [DTL Visual Editor](#)
- [DTL Syntax](#)
- [BPL Visual Editor](#)
- [BPL Syntax](#)
- [Adapters](#)
- [Workflow](#)

6.6 Ensemble 4.0

The current release introduces the following new features:

- [Caché 5.2 features](#)

Ensemble 4.0 offers the following enhancements:

- [Namespaces](#)
- [Default Login](#)
- [Command Line](#)
- [Management Portals](#)
- [BPL Syntax](#)
- [BPL Visual Editor](#)

-
- [BPL Components](#)
 - [Utility Functions](#)
 - [DTL Visual Editor](#)
 - [Adapters](#)

