Introducing Ensemble

Version 2017.2
2020-06-26

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

This book introduces Ensemble and options that you may want to include in your Ensemble productions. This book contains the following sections:

- Introduction to Ensemble

- Connectivity Options

- Business Processes and Business Logic

- Other Ensemble Options

For a detailed outline, see the table of contents.

Many additional materials — white papers, fact sheets, brochures, and independent industry reports — describe the benefits of using Ensemble. To find these documents on the InterSystems web site, navigate to the Ensemble product page at www.intersystems.com/ensemble/.

For general information, see the *InterSystems Documentation Guide*.

1

Introduction to Ensemble

The purpose of Ensemble is to enable you to develop, configure, deploy, and manage Ensemble productions. This chapter introduces productions and some of the basic Ensemble terminology. It contains the following sections:

- Introduction to Ensemble Productions
- Settings
- Message Flow in a Production
- Ensemble and Caché
- Data Persistence and Recovery

1.1 Introduction to Ensemble Productions

The purpose of Ensemble is to enable you to develop, configure, deploy, and manage Ensemble productions.

An Ensemble production integrates multiple, disparate software systems. A production includes elements that communicate with these external systems, as well as elements that perform processing that is internal to the production.

The elements in a production are known as *business hosts*. Ensemble has three kinds of business hosts, with different purposes as follows:

- *Business services* accept requests from entities outside the production and relay them to host classes inside Ensemble for processing and fulfillment.

- *Business processes* accept requests from host classes within the production — business services or business processes — and either process the requests or relay them to other host classes inside Ensemble for processing.

- *Business operations* accept requests from host classes inside Ensemble — business services or business processes — and either process the requests or relay them to entities outside Ensemble for processing.

The following figure provides a conceptual overview of a production and business hosts.
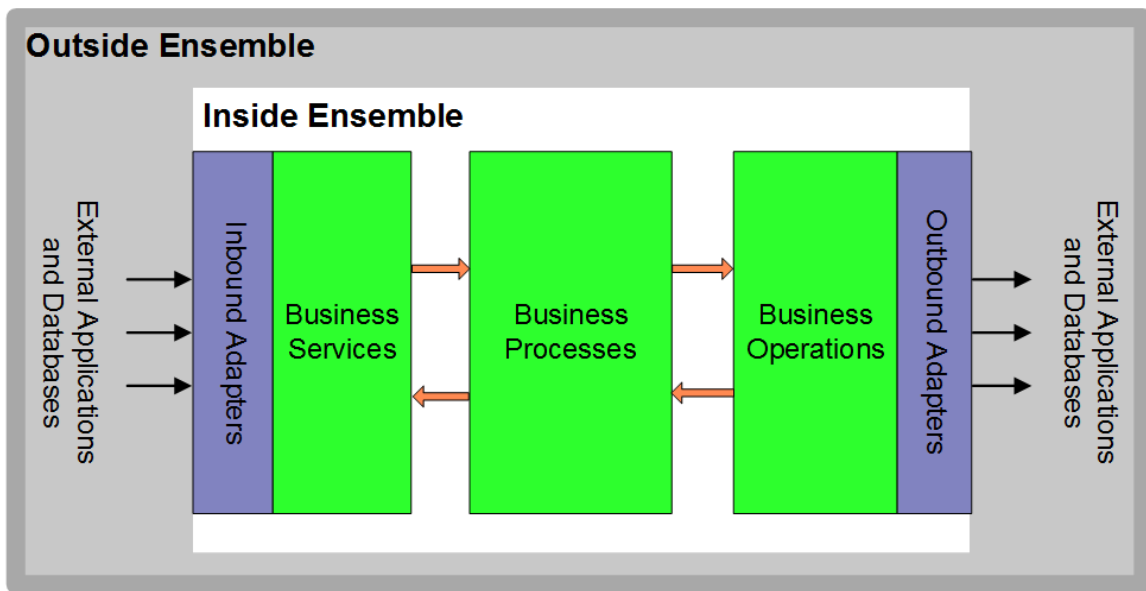
Business hosts communicate with each other via Ensemble messages. All Ensemble messages are stored in the Ensemble message warehouse and can be seen via the Management Portal.

In most cases (but not all), a business service has an associated *inbound adapter*. The role of an inbound adapter is to accept input from entities external to the production. Similarly, a business operation usually has an associated *outbound adapter*. The role of an outbound adapter is to send output to entities external to the production. Ensemble provides a large set of adapters to handle different technologies. For example, you use a different adapter for files than you do for FTP. It is also possible to define your own adapters.

The following figure shows an actual production, as seen in the Management Portal:
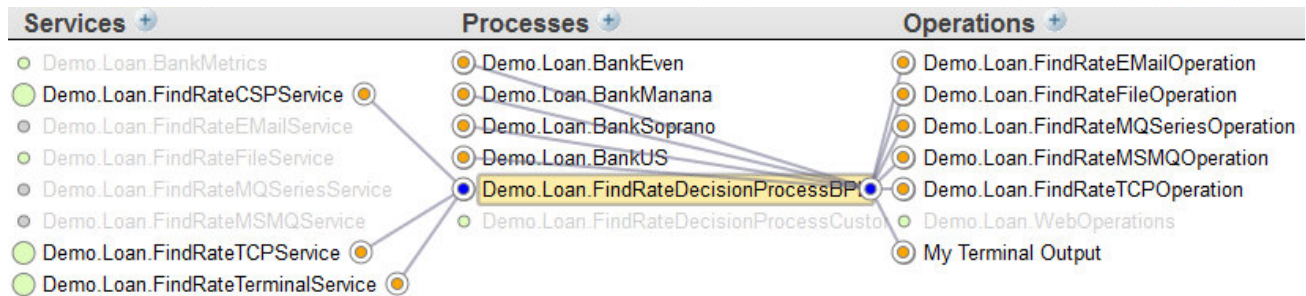
This view shows all the connections to and from one business host, **Demo.Loan.FindRateDecisionProcessBPL**. Note that this view does not display adapters because these are incorporated into the business service and business operation definitions.

1.2 Settings

An Ensemble production typically includes a large number of settings. *Settings* are configurable values that control the behavior of a production. Settings can affect a production in many ways. For example, a setting can specify:

- The TCP port on which a business service should listen.
- How frequently to check for new input.
- The external data source name (DSN) to use.
- The SSL configuration to use when connecting to an external entity.

- How long to stay connected.

- And so on.

An important feature of Ensemble is that a system administrator can modify settings while a production is running. The changes take effect immediately. The following shows an example of the web page that the system administrator uses to make such changes:

The production and its business hosts have settings provided by Ensemble; they correspond to properties of the production and business host classes. You can define additional settings in exactly the same way, by defining your own subclasses of Ensemble classes. You can also remove settings so that the corresponding properties are hardcoded and not configurable.

1.3 Message Flow in a Production

An Ensemble production typically processes incoming events as follows:

1. An inbound adapter receives an incoming event, transforms it into an Ensemble message object, and passes it to its associated business service.

2. The business service creates a follow-on request message, and passes this new message to a business process or business operation within Ensemble.

3. A business process that receives a request message executes a predefined set of activities, in sequence or in parallel. These activities may include sending follow-on messages to other business hosts. Business processes are also responsible for most or all of the business logic in the production. The next chapter provides more detail.

4. A business operation encapsulates the capabilities of a resource outside Ensemble, usually an external software application. The business operation transforms properties of the Ensemble request message object into a format usable by the external application API.

5. An outbound adapter manages the details of communicating with a specific external system or application from within Ensemble. It transmits the API call to the external entity.

6. The response from the external system or application can trigger a cascade of response messages back to the external entity that started the flow of events. Details depend on the design choices made by the production developers.

As a demonstration, the following figure shows a trace of a set of related messages, which Ensemble sent in response to an initial message (in this case sent by the Ensemble testing service rather than an exterior source):
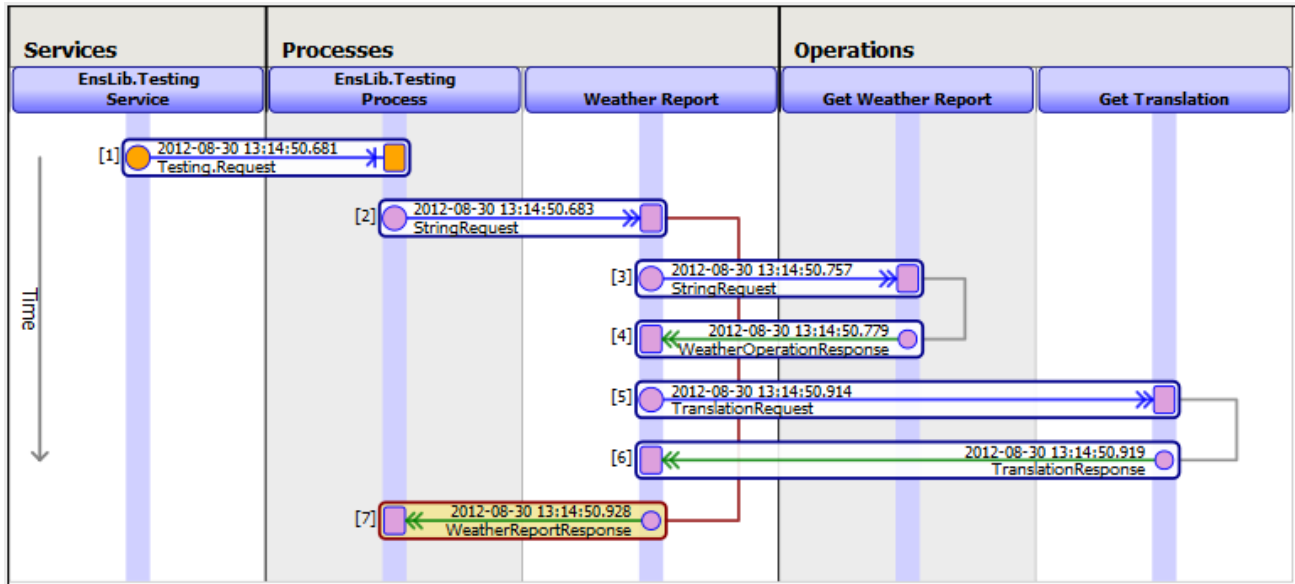
The processing can also include workflow, which makes it possible to incorporate human interaction into automated business processes. Uses of workflow within the enterprise might include order entry, order fulfillment, contract approval, or help desk activities. The chapter "Other Ensemble Options" provides more information.

1.4 Ensemble and Caché

Ensemble is built on top of Caché — that is, an Ensemble installation contains a complete installation of Caché.

Caché is a full-featured database system; it includes all the features needed for running mission-critical applications (including journaling, backup and recovery, and system administration tools). To help reduce operating costs, Caché is designed to require significantly less database administration than other database products.

For Ensemble, the Management Portal includes all the options that are present in a Caché installation, as well as additional menus that are specific to Ensemble.

Ensemble uses the security tools provided in Caché. Caché provides a complete security model that includes authentication, authorization, auditing, and database encryption. Additional tools provide secure communications to and from external entities.

Caché (and therefore Ensemble) also provides several built-in, general-purpose programming languages that can be used to create class definitions and routines. This means that Ensemble productions can provide highly customized behavior.

For an introduction to Caché, see *Introduction to Caché*.

1.5 Data Persistence and Recovery

In the event of a planned or unplanned shutdown, Ensemble provides a high degree of recoverability. That is, when the system is restarted, it can resume execution with little to no interruption in service. The degree of recovery depends on several factors: the robustness of the external applications; implementation details of the various business services, business processes, and business operations, and the nature of the outage itself.

In addition to storing the production definition and all supporting code and data, Ensemble stores runtime production data to aid in recovery of the system if some sort of breakdown should occur. This runtime data includes the following:

- Messages.
- State information related to long-running business processes, which may take days or weeks to complete. That is, stopping and starting a production does not interfere with this processing.
- Business rule log entries.
- The Event Log.

All this data is persistent across system outages.

Typically, you purge data at periodic intervals, as described in *Managing Ensemble*.

2

Connectivity Options

Ensemble enables you to define inbound and outbound connections to a wide variety of technologies. There are three general categories of options:

- Adapters
- Specialized Business Host Classes
- Gateways

Many of these tools are built on the Caché tool set, which also includes many options that Ensemble does not use.

If the built-in Ensemble tools do not meet your needs, you can create your own components based on any suitable combination of Caché and Ensemble tools. For a compact, combined list of Caché and Ensemble tools, see the table of contents of the InterSystems Technology Reference.

2.1 Adapter Library

Ensemble provides many adapters, and you can define your own. Ensemble provides adapters for the following scenarios:

Email

Receives email messages via the POP3 protocol and sends email messages via the Simple Mail Transfer Protocol (SMTP).

See *Using Email Adapters with Ensemble*.

File

Reads and writes files on the local network. Able to open, create, delete, modify, and move files. The file content can be character or binary data.

See *Using File Adapters with Ensemble*.

Also see "Business Host Classes That Use File Adapters," later in this chapter.

FTP

Receives and sends files between local and remote systems via the File Transfer Protocol (FTP). The file content can be characters or binary data. The adapters support FTP over SSL and SFTP.

See *Using FTP Adapters with Ensemble*.

Also see "Business Host Classes That Use FTP Adapters," later in this chapter.

HTTP

Provides an HTTP listener for custom port listening, XML listening, or raw HTML handling. Supports the standard HTTP operations Post, Get, Put, and Delete. Allows the use of proxy servers.

See *Using HTTP Adapters with Ensemble*.

Also see "Business Host Classes That Use HTTP Adapters," later in this chapter.

LDAP

The outbound LDAP adapter can send requests to an LDAP server and receive responses.

See the EnsLib.LDAP.OutboundAdapter entry in the *Class Reference*.

MQSeries

Receives and sends messages in IBM WebSphere MQ (MQ Series) format. Message content can be a specific data type or a binary data stream. The adapter can simply send the message, or send it and then pull the corresponding response from the message queue.

See *Using IBM WebSphere MQ Adapters with Ensemble*.

Pipe

Able to execute a shell command and communicate with it via pipes. Capable of handling character data or a binary data stream.

See the EnsLib.Pipe.InboundAdapter and EnsLib.Pipe.OutboundAdapter entries in the *Class Reference*.

SAP

Allows Ensemble to interface with the SAP Java Connector (SAP JCo).

See *Using the SAP Java Connector with Ensemble* for details.

Siebel

A subclass of the HTTP adapter that sends requests to a Siebel server and returns responses.

See the EnsLib.Siebel.HTTPOutboundAdapter entry in the *Class Reference*.

SOAP

Listens for SOAP requests on a local TCP port or via a standard web server. Dispatches outbound requests by acting as a SOAP client to an external SOAP server. In this special case, adapters are not necessarily involved.

See *Creating Web Services and Web Clients with Ensemble*.

Also see "Business Host Classes That Use SOAP," later in this chapter.

SQL

Executes SQL statements against a remote database via an ODBC-defined or JDBC-defined connection.

See *Using SQL Adapters with Ensemble*.

TCP

Manages an incoming or outgoing TCP connection. Allows simultaneous handling of multiple connections. Supports character and binary data streams, and counted data blocks. The adapters also supported framed protocols.

See *Using TCP Adapters with Ensemble*.

Also see "Business Host Classes That Use TCP Adapters," later in this chapter.

Telnet

Directly initiates and manages a Telnet connection.

See the EnsLib.Telnet.OutboundAdapter entry in the *Class Reference*.

To use an adapter, you define a business host class that uses the adapter class. For a general description of this process, see *Developing Ensemble Productions*. The books in the previous list provide details that apply to the specific adapters.

As noted previously, you can also define your own adapters. For information, see *Developing Ensemble Productions*.

2.2 Business Host Classes That Use Adapters

For specific scenarios, Ensemble provides specialized business service classes and business operation classes that already use specific adapters:

- File adapters

- FTP adapters

- HTTP adapters

- SOAP (in this special case, adapters are not necessarily involved)

- TCP adapters

To use these business hosts, it is not generally necessary to perform any coding.

For practical reasons, the following sections might not cover all the specialized business host classes that Ensemble provides. To find all the specialized business host classes that specify an adapter, perform a search in Studio, and look for `Parameter ADAPTER` within the EnsLib package. Studio then displays the names of the classes, which generally indicate which adapter they use.

2.2.1 Business Host Classes That Use File Adapters

Ensemble provides business host classes that use the file adapters. The following table summarizes the content types that Ensemble supports and indicates where to find information.

File Contents	See
ASTM documents	*Ensemble ASTM Development Guide*
DICOM documents	*Ensemble DICOM Development Guide*
EDIFACT documents	*Ensemble EDIFACT Development Guide*
HL7 version 2 messages	*Ensemble HL7 Version 2 Development Guide*
X12 documents	*Ensemble X12 Development Guide*
XML documents	*Ensemble XML Virtual Document Development Guide*
Any	"Using the File Passthrough Service and Operation Classes" in *Using File Adapters with Ensemble*

2.2.2 Business Host Classes That Use FTP Adapters

Ensemble provides business host classes that use the FTP adapters. The following table summarizes the content types that Ensemble supports and indicates where to find information.

File Contents	See
ASTM documents	*Ensemble ASTM Development Guide*
EDIFACT documents	*Ensemble EDIFACT Development Guide*
HL7 version 2 messages	*Ensemble HL7 Version 2 Development Guide*
X12 documents	*Ensemble X12 Development Guide*
XML documents	*Ensemble XML Virtual Document Development Guide*
Any	EnsLib.FTP.PassthroughService and EnsLib.FTP.PassthroughOperation in the *Class Reference*

2.2.3 Business Host Classes That Use HTTP Adapters

Ensemble provides business host classes that use the HTTP adapters. The following table summarizes the content types that Ensemble supports and indicates where to find information.

File Contents	See
EDIFACT documents	*Ensemble EDIFACT Development Guide*
HL7 version 2 messages	*Ensemble HL7 Version 2 Development Guide*

2.2.4 Business Host Classes That Use SOAP

Ensemble provides business host classes that communicate via SOAP (in this special case, adapters are not necessarily involved). The following table summarizes the content types that Ensemble supports and indicates where to find information.

File Contents	See
HL7 version 2 messages	*Ensemble HL7 Version 2 Development Guide*

2.2.5 Business Host Classes That Use TCP Adapters

Ensemble provides business host classes that use TCP adapters. The following table summarizes the content types that Ensemble supports and indicates where to find information.

File Contents	See
ASTM documents	*Ensemble ASTM Development Guide*
HL7 version 2 messages	*Ensemble HL7 Version 2 Development Guide*
X12 documents	*Ensemble X12 Development Guide*
Any	EnsLib.TCP.PassthroughService and EnsLib.TCP.PassthroughOperation in the *Class Reference*

2.3 Gateways

Caché and Ensemble both provide the SQL Gateway, which provides access from Caché to external databases via JDBC and ODBC. You can, for example, link to tables and views in external sources and access the data in them in the same way you access any local data. See *Using the SQL Gateway*.

Ensemble provides the following additional gateways:

- The *Object Gateway for .NET* provides an easy way for Ensemble to interoperate with Microsoft .NET Framework components. The .NET Gateway can instantiate an external .NET object and manipulate it as if it were a native object within Ensemble.

  See *Using the Object Gateway for .NET with Ensemble*.

- The *Java Gateway* provides an easy way for Ensemble to interoperate with Java components. The Java Gateway can instantiate an external Java object and manipulate it as if it were a native object within Ensemble.

  The Java Gateway also allows Ensemble to interact with JMS (Java Message Service) systems.

  See *Using the Java Gateway*.

3

Business Processes and Business Logic

This chapter describes the kinds of logic supported in business processes. It includes the following sections:

- Introduction
- Types of Business Processes
- Data Transformations
- Business Rules

3.1 Introduction

Business processes are the middle part of any production. They accept requests from host classes inside the production — business services or business processes — and then either process the requests or relay them to other host classes inside Ensemble for processing.

InterSystems recommends the following division of labor within a production: Use business services to receive input from outside of the production and simply forward it (as messages) into the production. Use business processes to handle any needed business logic. Use business operations to receive messages from within the production and simply generate output for destinations outside of the production. That is, centralize the business logic within the business processes.

Accordingly, Ensemble provides extensive support for complex logic within business processes, and this logic can be defined by nontechnical users.

First, a business process can contain its own complex logic. It can also use the following reusable items:

- *Data transformations* calculate and apply changes to message contents.
- *Business rules* change the behavior of business processes at decision points or send messages to specific destinations based on message type, message contents, or where the message came from.

Ensemble provides tools that enable nontechnical users to define business processes, data transformations, and business rules. These users can view and edit the logic visually without programming or diagramming skills.

Note that there is overlap among the lower-level options available in business processes, data transformations, and business rules. For a comparison, see "Comparison of Business Logic Tools" in *Developing Ensemble Productions*. It is worthwhile to review these options before deciding how to organize your logic.

3.2 Types of Business Processes

Ensemble provides the following general types of business process:

- *BPL processes*, which are based on the class Ens.BusinessProcessBPL. To create these processes, you use a graphical editor that is intended for use by nontechnical users. This editor is available in the Management Portal and Studio. See *Developing BPL Processes*.

    The name of these processes comes from BPL (Business Process Language), which is the XML-based language that Ensemble uses to represent their definitions.

- *Routing processes*, which are based on the class EnsLib.MsgRouter.RoutingEngine or EnsLib.MsgRouter.VDocRoutingEngine.

    Ensemble provides a set of classes to route specific kinds of messages. The following books indicate the routing process to use for different kinds of messages:

Message Type	See
ASTM	*Ensemble ASTM Development Guide*
DICOM	*Ensemble DICOM Development Guide*
EDIFACT	*Ensemble EDIFACT Development Guide*
HL7	*Ensemble HL7 Version 2 Development Guide*
HL7v3	*Ensemble HL7 Version 3 Development Guide*
X12	*Ensemble X12 Development Guide*
XML	*Ensemble XML Virtual Document Development Guide*

    To use these classes, no coding is generally necessary. It is, however, necessary, to provide a set of business rules, introduced later in this chapter.

- *Custom business processes*, which are based on the class Ens.BusinessProcess. Note that all the previously listed business process classes inherit from this class. In this case, it is necessary to use Studio.

    The EnsLib and Demo packages provide many examples.

    For information on defining custom business processes, see "Developing Business Processes" in *Developing Ensemble Productions*.

A production can include any mix of these business processes.

The following shows a partial example of a BPL business process, as displayed in the BPL editor:
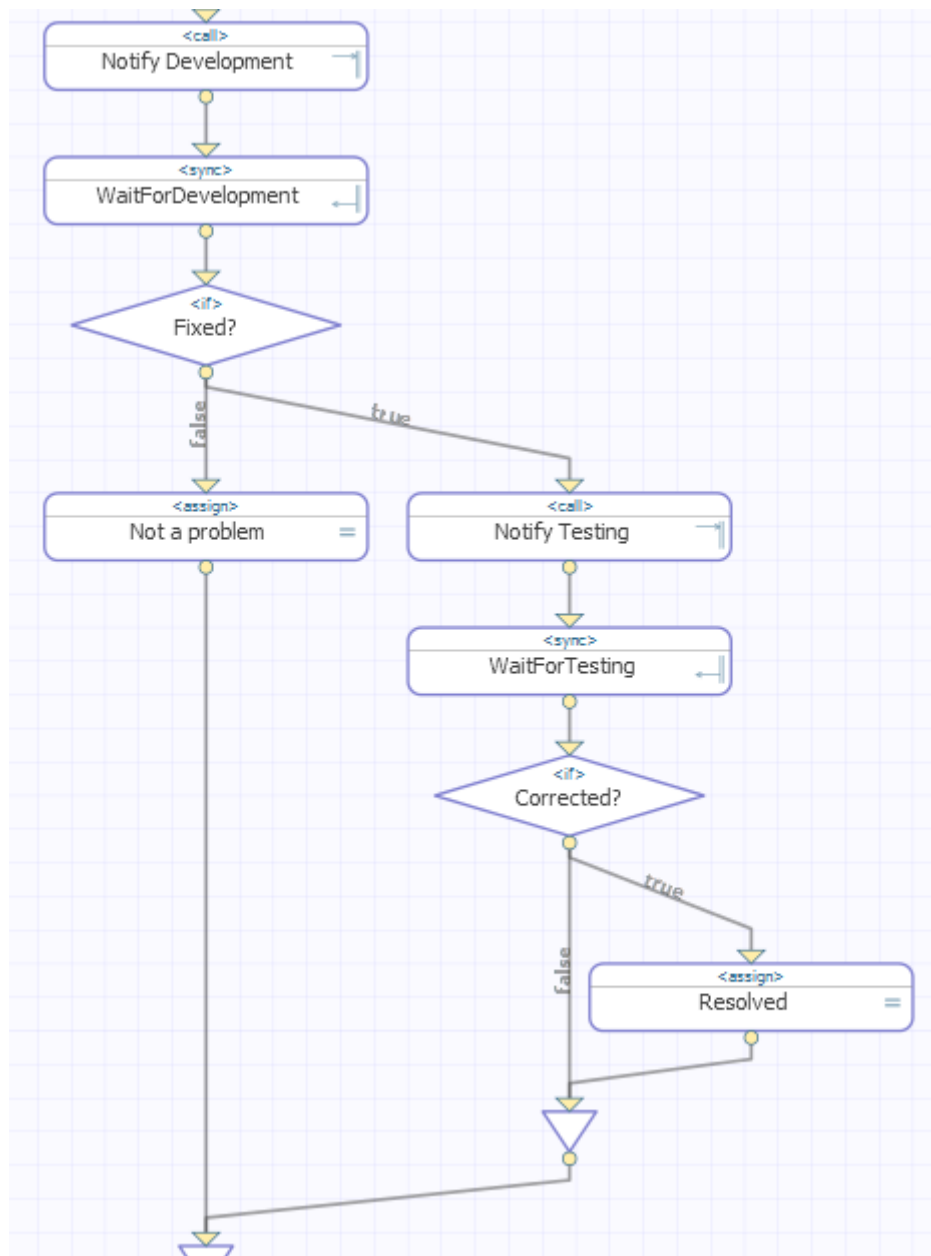
3.3 Data Transformations

A *data transformation* creates a new message that is a transformation of another message. You can invoke a data transformation from a business process, another data transformation, or a business rule.

When you *transform* a message, your data transformation swaps out the old message body object (the source) and exchanges it for a new one (the target). Some of the transformations that occur during this process can include:

- Copying values from properties on the source to properties on the target.

- Performing calculations using the values of properties on the source.

- Copying the results of calculations to properties on the target.

- Assigning literal values to properties on the target.

- Ignoring any properties on the source that are not relevant to the target.

A data transformation is a class based on Ens.DataTransform or its subclass, Ens.DataTransformDTL.

- If you use Ens.DataTransformDTL, the transformation is called a *DTL transformation*. To create these transformations, you use a graphical editor that is intended for use by nontechnical users. This editor is available in the Management Portal and Studio. For details, see *Developing DTL Transformations*.

  The name of these transformations comes from DTL (Data Transformation Language), which is the XML-based language that Ensemble uses to represent their definitions.

- If you use Ens.DataTransform, the transformation is a *custom transformation*. In this case, you must use Studio. For information on defining these, see *Developing Ensemble Productions*.

The following shows an example DTL transformation, as seen in the DTL editor:
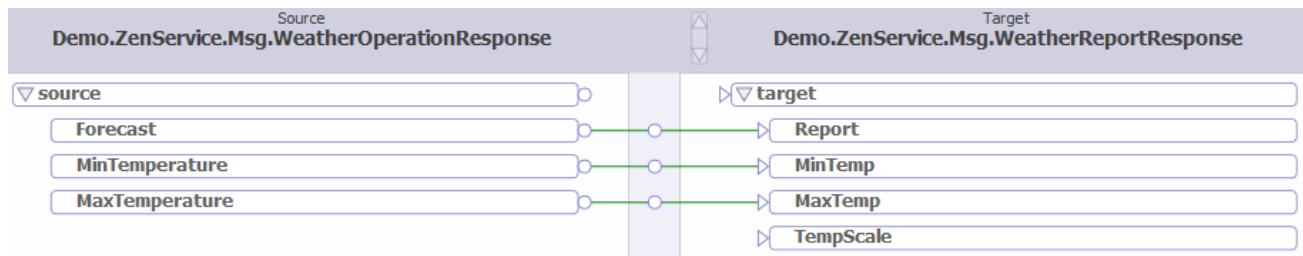
3.4 Business Rules

A *business rule* (also known as a *business rule set*) can return a value, transform data, or both. You can invoke a business rule from a business process or from another business rule.

A business rule is a class based on Ens.Rule.Definition. You define these in the Management Portal, which provides a visual editor for the benefit of nontechnical users. The following shows a partial example, as seen in this editor:

For details, see *Developing Business Rules*.

4

Other Ensemble Options

This chapter provides an overview of other options within Ensemble. It is intended to alert you to features that you may want to include in your Ensemble productions. This chapter discusses the following topics:

- User Portal

- Workflow

- Business Activity Monitoring

- Alerts

- Publish and Subscribe Message Delivery

- Message Bank

4.1 User Portal

The User Portal (formally called the DeepSee User Portal) is intended for direct use by end users, in contrast to such back end tools as Studio and the Management Portal. This portal is intended for general use and has a general appearance, despite its specific name. It is not labeled with "DeepSee." The following shows an example:

The User Portal is designed to enable end users to do the following tasks:

- View and dashboards, which can include Ensemble business metrics, introduced later in this chapter.

- View and manage workflow tasks, introduced in the next section.

- Send messages to other users of the User Portal, as seen in the upper left corner of the previous figure.

For details, see the DeepSee User Guide.

4.2 Workflow

Ensemble provides support for workflow. *Workflow* makes it possible to incorporate human interaction into automated business processes. Uses of workflow within the enterprise might include order entry, order fulfillment, contract approval, or help desk activities.

An Ensemble production achieves workflow using a business process and set of business operations. A business process receives a request, organizes the tasks required to fulfill that request, and then calls upon business operations to perform these tasks.

Each of these business operations assigns the task to a workflow role, which places the task into the workflow inbox of each Ensemble user who belongs to that role. To access this inbox, users work with the User Portal, introduced in the previous topic.

When a user marks as a task as complete, the workflow engine continues with the next processing step.

For details, see *Defining Workflows*.

4.3 Business Activity Monitoring

Ensemble includes all the elements required to include *business activity monitoring* (*BAM*) as part of an enterprise integration project.

First, it provides *business metrics*, which are specialized business service classes that calculate specific values at a configurable time interval. These calculations can take a variety of forms, involving calls to business operations, federated databases, the Ensemble database, or any arbitrary source code contained within the business metric class. See "Defining Business Metrics" in *Developing Ensemble Productions*.

Second, it provides *dashboards*, which are web pages that provide a real-time, graphical display of metric values. Each dashboard presents one or more meters, and these display business metrics. Ensemble provides a complete set of built-in meters that includes speedometers, light bars, traffic lights, and others.

The following shows an example dashboard that displays business metrics:
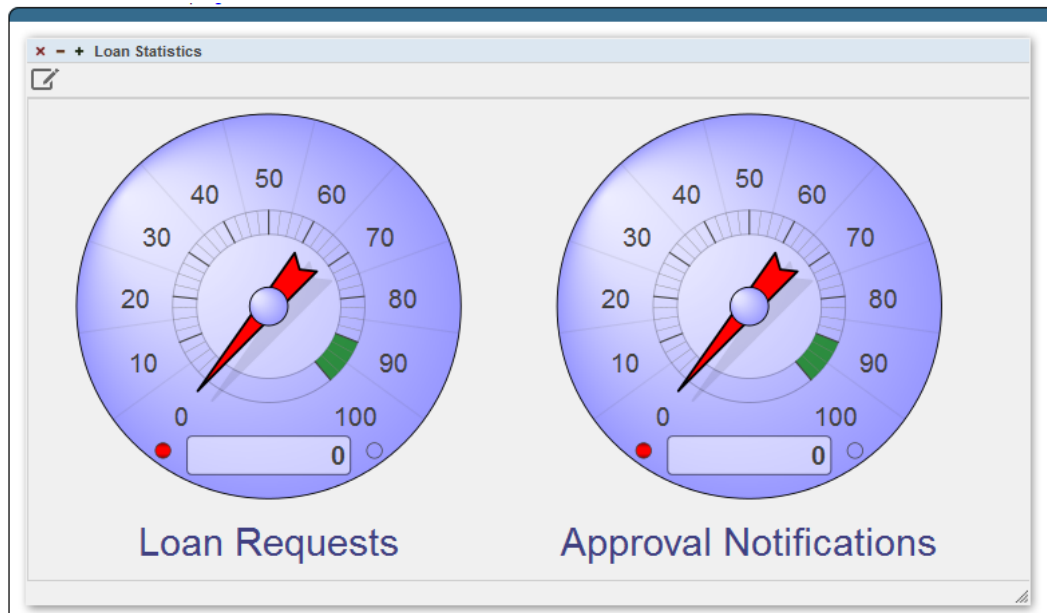
To getting started with dashboards, see "Creating Dashboards" in *Configuring Ensemble Productions*. For complete information, see *Creating DeepSee Dashboards*.

4.4 Alerts

An *alert* sends notifications to applicable users while an Ensemble production is running, in the event that an alert event occurs. The intention is to alert a system administrator or service technician to the presence of a problem. Alerts may be delivered via email, text pager, or another mechanism. For details, see "Programming in Ensemble" in *Developing Ensemble Productions*.

4.5 Publish and Subscribe Message Delivery

Ensemble also supports *publish and subscribe* message routing. This technique routes a message to one or more subscribers based on the fact that those subscribers have previously registered to be notified about messages on a specific topic. For details, see *Managing Ensemble*.

4.6 Message Bank

The *Enterprise Message Bank* is an optional remote archiving facility where you can collect messages, Event Log items, and search table entries from *multiple* Ensemble client productions. It consists of the following components:

- The Message Bank server, which is a simple Ensemble production consisting exclusively of a Message Bank service that receives submissions from any number of client productions.

- A client operation (the Message Bank operation) that you add to an Ensemble production and configure with the address of a Message Bank server.

To get started, see "Defining an Enterprise Message Bank" in *Developing Ensemble Productions*.