SeeBeyond ICAN Suite

SeeBeyond ICAN Suite Primer

Release 5.0.5

SeeBeyond®

Contents

Chapter 4

eInsight Enterprise Service Bus   31

Chapter 5

eVision Studio   36

Chapter 12

eBAM Studio 80

List of Figures

Introduction

This chapter provides an overview of the SeeBeyond® Integrated Composite Application Network (ICAN) Suite™ and general information about this guide .

What's in This Chapter

- **About the SeeBeyond ICAN Suite** on page 10
- **What's in This Document** on page 11
- **SeeBeyond Web Site** on page 12
- **SeeBeyond Documentation Feedback** on page 13

1.1 About the SeeBeyond ICAN Suite

In today's economic climate, every business must optimize operations. This means that organizations will profit more by making better use of systems already in use, connecting seamlessly with partners, suppliers, and customers; adjusting to tremendous transaction volumes and managing inventory at peak efficiency.

SeeBeyond provides a single integration solution that connects, integrates, and optimizes—quickly and simply. The SeeBeyond ICAN Suite, depicted in Figure 1, is the most comprehensive and unified solution on the market.

Figure 1   SeeBeyond ICAN Suite
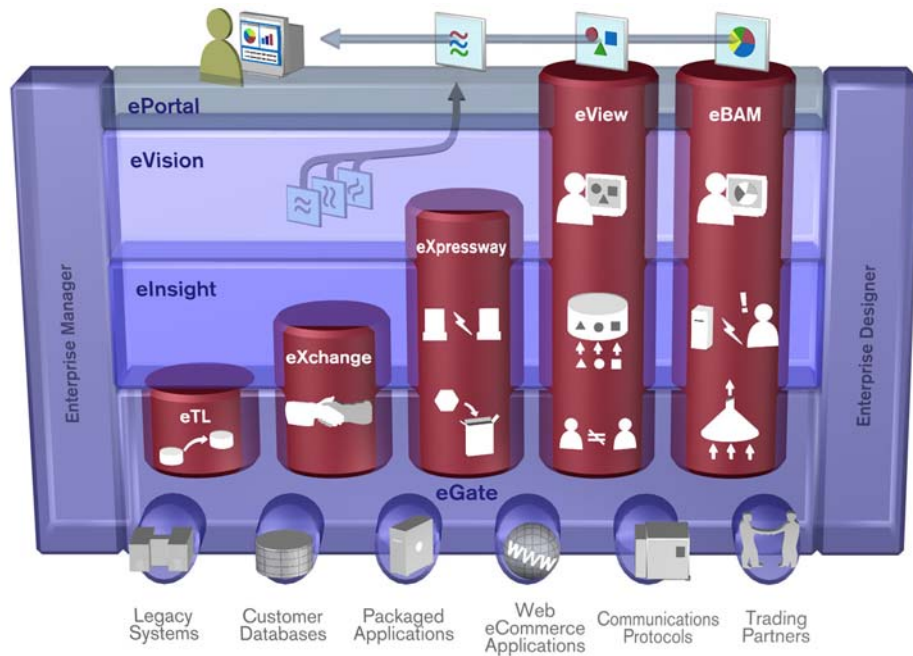
With the SeeBeyond ICAN Suite, you can:

- **Connect** to partners, suppliers, and customer organizations rapidly and securely to enable the exchange of business critical information. Quickly establish trading partner relationships and dynamically manage those relationships.

- **Integrate** all critical business data, systems, and disparate applications among multiple independent business units into a common platform to deliver real-time information to partners, suppliers, and customers.

- **Optimize** core internal and external business processes to create an agile organization able to quickly react to changing business conditions in real-time.

The ICAN Suite currently represents the most comprehensive, unified eBusiness infrastructure available for companies seeking to improve business operations and drive rapid return on investment. It delivers seamless application integration, dynamic business-to-business connectivity, and robust business process management capabilities across the extended enterprise for customers, suppliers, and partners.

1.2   What's in This Document

This document overviews products in the ICAN Suite. It includes the following chapters:

- **Chapter 1 "Introduction"** provides an overview of this document.

- **Chapter 2 "eGate Integrator"** describes the architecture, components, and functionality of the SeeBeyond eGate Integrator application.

- **Chapter 3** **"eInsight Business Process Manager"** describes the features of the eInsight Business Process Manager.

- **Chapter 4** **"eInsight Enterprise Service Bus"** describes the features of the eInsight Enterprise Service Bus.

- **Chapter 5** **"eVision Studio"** describes the features of the eVision graphical design studio, a graphical design studio for the WYSIWYG creation of integrated Web applications.

- **Chapter 6** **"eXchange Integrator"** describes the features of the eXchange Integrator application. The eXchange Integrator application provides eBusiness protocol support, enables trading partner management, and ensures secure eBusiness communications.

- **Chapter 7** **"B2B Protocol Manager Composite Applications"** describes the features of the Protocol Manager Composite Applications. Protocol-specific composite applications are designed using eXchange tools, and expose all components so they can be customized as needed.

- **Chapter 8** **"eView Studio"** describes the features of the eView Studio. eView Studio is an application building tool that enables you to design, configure, and create a master index that will uniquely identify and cross-reference the business objects stored in your system databases.

- **Chapter 9** **"eIndex Global Identifier Composite Application"** describes the features of eIndex Global Identifier Composite Application (eIndex). eIndex is a healthcare-oriented, enterprise-wide master person index that maintains the most current information about the patients in your organization.

- **Chapter 10** **"eTL Integrator"** describes the features of the eTL Integrator. eTL Integrator provides the functionality to *extract*, *transform*, and *load* data between databases.

- **Chapter 11** **"ePortal Composer"** describes the features of the ePortal Composer. ePortal Composer is used to create custom portals and allows an organization to present a single, unified view of enterprise data and applications to employees, customers, and partners.

- **Chapter 12** **"eBAM Studio"** describes the features of eBAM. eBAM provides *business activity monitoring* functionality for collection, aggregation, and presentation of business activity data according to specified key performance indicators. eBAM Studio provides the tools for generating custom digital dashboards for defining and monitoring these key performance indicators.

The **Glossary** on page 84 lists SeeBeyond eGate Integrator terms and their definitions.

1.3 SeeBeyond Web Site

The SeeBeyond Web site is your best source for up-to-the-minute product news and technical support information. The site's URL is:

http://www.seebeyond.com

1.4   SeeBeyond Documentation Feedback

We appreciate your feedback. Please send any comments or suggestions regarding this document to:

docfeedback@seebeyond.com

eGate Integrator

The SeeBeyond ICAN Suite is based on eGate Integrator, the most widely deployed Enterprise Application Integration (EAI) platform on the market. eGate quickly solves complex connectivity issues and enables the dynamic delivery of information across applications and systems to partners and customers with unmatched performance, flexibility, and ease of use.

What's in This Chapter

2.1 Introduction

SeeBeyond® eGate™ Integrator is a fully J2EE certified and Web services-based, distributed integration platform that serves as the foundation of the SeeBeyond Integrated Composite Application Network (ICAN) Suite™. eGate Integrator provides the core integration platform, comprehensive systems connectivity, guaranteed messaging and robust transformation capabilities while providing a unified, single sign-on environment for integration development, deployment, monitoring and management. eGate Integrator supports portability of integrations across common J2EE application servers through a completely open, J2EE-certified and Web services-based architecture.

Figure 2 eGate Integrator
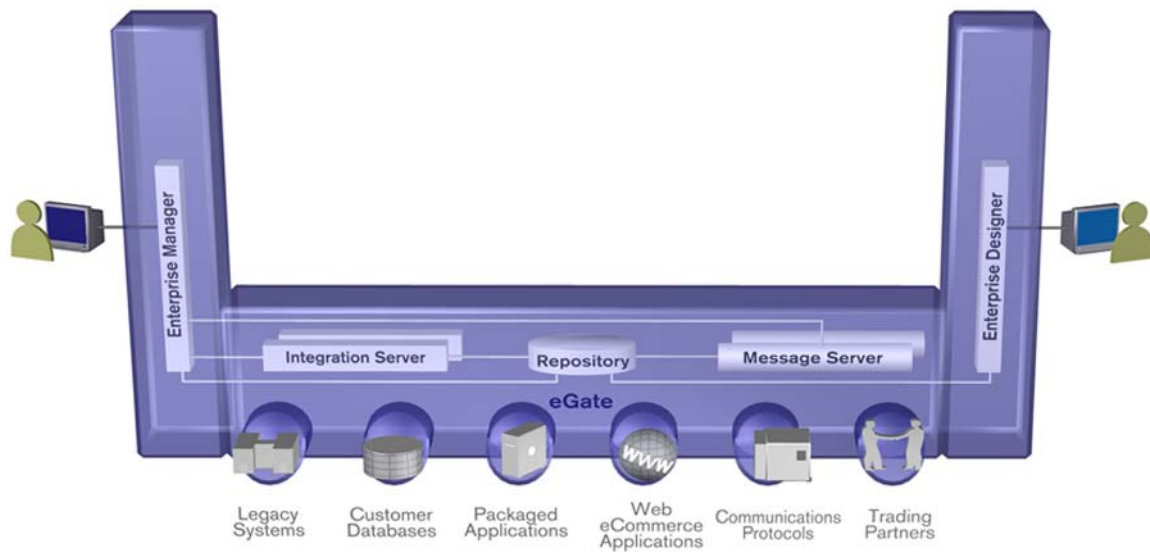
As shown in Figure 2, the heart of eGate Integrator is the Repository, which is a comprehensive store of information common to the entire enterprise. An integrated UDDI-compliant server allows publication and discovery of Web services. The run-time environment employs J2EE-compliant integration servers as operational engines and JMS-compliant message servers for the propagation of messages. The flexibility of the eGate system allows the option of deployment to a SeeBeyond run-time environment or to third-party application servers across a distributed network of hardware platforms.

Enterprise Manager provides a unified, browser-based framework for managing all aspects of the run-time environment, as well as installing and updating all ICAN Suite components. Enterprise Designer provides a unified, graphical development environment for integrating systems and developing composite applications using Web services.

eGate Integrator can communicate with and link multiple applications and databases across a variety of different operating systems. eGate performs with a wide variety of hardware, message standards, operating systems, databases, and communication protocols in both real-time and batch (scheduled) integration modes.

2.2   Integration Model

SeeBeyond addresses application integration by means of an eGate Project, which contains the business logic required to solve the specific problem. The Project contains the various logical components and supporting information required to perform the routing, processing, and caching of messages containing the relevant data from one application to another. All Project information is stored in the Repository.

Projects are created using tools contained within Enterprise Designer and, once deployed, can be run and monitored using Enterprise Manager. Projects can also be set

up to be run from the business process level using the SeeBeyond eInsight Business Process Manager, if that product is also installed.

Projects are run within individual sets of system definitions, referred to as Logical Hosts. These are defined within Environments, which represent the physical resources required to implement the Project. Projects are mapped to the individual Environments by means of deployment profiles, which are defined within the Enterprise Designer and become part of the Project. Activating the deployment profile deploys the Project to the associated Environment.

This structure of Projects, Environments, and deployment profiles isolates each implementation into logical and physical components. This provides you with extensive flexibility and efficiency in designing eGate Integrator implementations. For example, once you build your Projects and Environments, you have the flexibility to change each component without having to make changes to the other component.

The finished Project, of course, will run in your production Environment; separate Environments, having the same structure as the production Environment, should be created for development and testing. You may also want some additional Environments, such as staging. The following figure illustrates the eGate Integrator implementation model using a healthcare-related example.

Figure 3 eGate Integrator Implementation Model
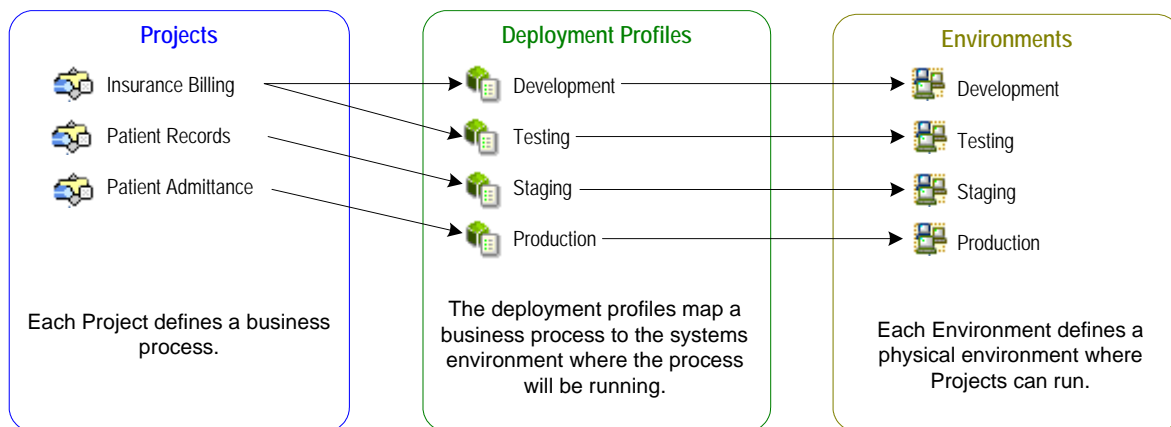
In the figure above, any of the Projects can be deployed to any of the Environments via the mapping defined in the deployment profiles. The example in the figure above shows that the patient admittance Project is already in the production phase and therefore was deployed using the production deployment profile. The patient records Project is in the staging phase and was therefore deployed to the staging Environment using the staging deployment profile. The insurance billing Project is still being developed and tested, and therefore it is deployed to development and testing via the development and testing profiles.

In broad outline, implementing an integration Project using eGate Integrator includes the following steps:

1   **Design your Project.**

2   **Define your Environments.**

3 **Create your Deployment Profiles.**

4 **Deploy the eGate Project.**

These implementation steps are all accomplished using the eGate Enterprise Designer.

2.3 System Architecture

SeeBeyond's eGate Integrator employs a versatile architecture that is ideally suited to distributed computing environments. As a result, the various components of an eGate Integrator system can reside on the same hardware platform (assuming adequate system resources), or be distributed across several different hardware platforms in the enterprise network. Figure 4 shows an example system implementation that is highly distributed.

Figure 4 Distributed eGate Integrator System
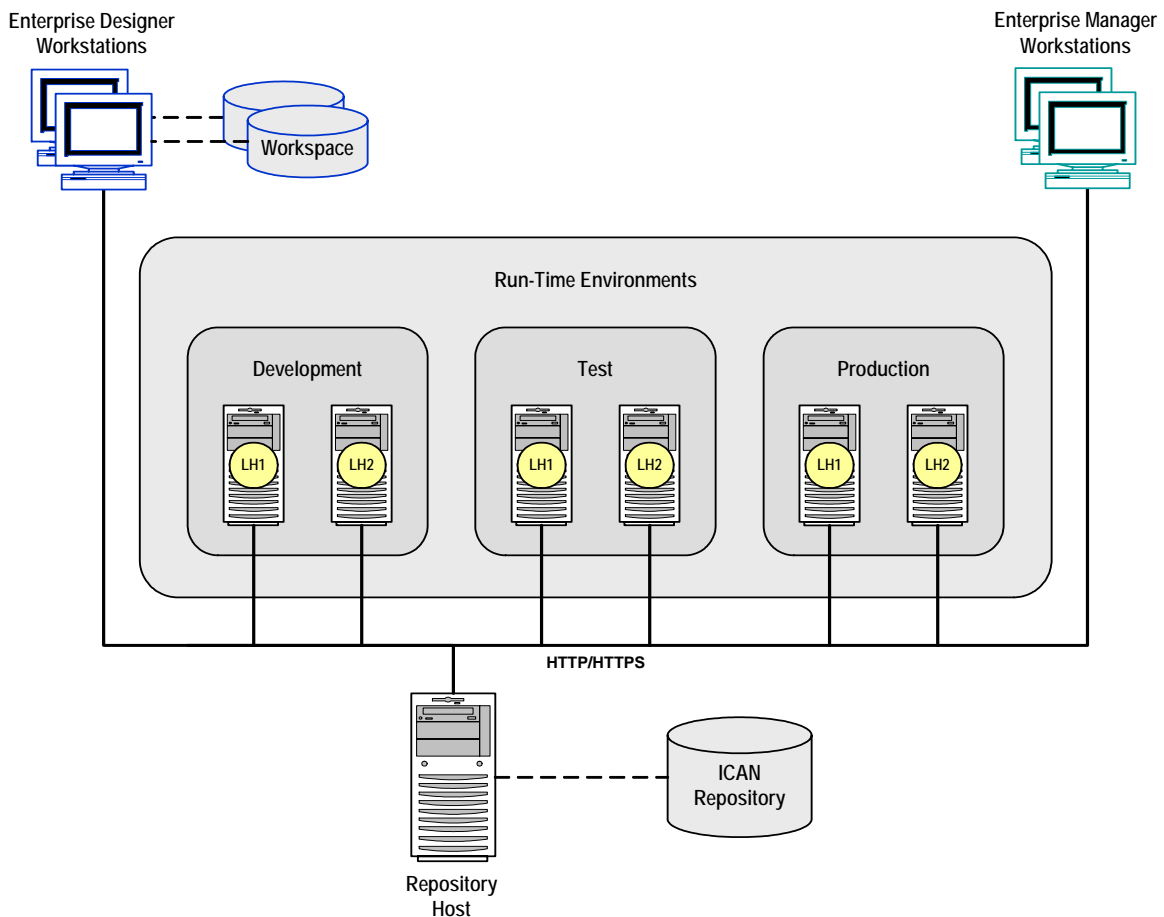
Note: *In this scenario it is assumed that all instances of eGate are of the same release.*

2.3.1 Repository

The setup, components, and configuration information for the elements of a Project are stored in the Repository. The Repository also stores all of the product binary files that are required at run time by the Logical Hosts. The components and configurations are downloaded to the Logical Host during the initial bootstrap process and as needed after design-time configuration changes are made.

As shown in Figure 4, a single Repository serves the entire enterprise. This common Repository is used for development, testing, and production purposes. Communication between the Repository and other ICAN components can be configured to use either HTTP or HTTPS. The Enterprise Designer and Enterprise Manager clients can communicate with the Repository through a firewall. The Repository makes Web Services available via a UDDI registry.

2.3.2 Run-Time Environments

An eGate Environment represents the total system required to implement a Project. It consists of a collection of Logical Hosts, capable of hosting components of the ICAN Suite, along with information about external systems involved in the implementation.

- **Logical Hosts**
  Each Environment contains one or more system definitions. Each definition must include one or more **integration servers** such as the SeeBeyond Integration Server, which are the engines that run eGate Collaborations and eWays, and one or more **message servers** such as the SeeBeyond JMS IQ Manager, which manage JMS topics (publish-and-subscribe messaging) and queues (point-to-point messaging). Each collection of integration servers and message servers, plus additional software modules, comprise what is known as a *Logical Host*.

- **External Systems**
  An external system is a representation of a real, physical system that exists within the specific Environment, with configuration properties for locating and accessing that system.

In the example system shown in Figure 4, the production environment is split across two hardware platforms, each supporting a single Logical Host. Separate environments for development and testing should duplicate the structure of the production environment. The test environment should be supported by hardware similar to that supporting the production environment, to allow performance and load testing to give representative throughput results. The hardware supporting the development environment, however, does not usually have the same performance requirements as that supporting the test and production environments.

An eGate Project is created within the development environment, then migrated to the test environment, and finally to the production environment. This migration path is a necessary and highly critical practice in implementing a working system.

Note again that there is no requirement for the components shown in Figure 4 to run on separate systems; all could run on a single system, provided that resources (CPU, memory and disk) are sufficient to support concurrent usage.

2.4 Projects

An eGate Project represents the logical system designed to solve either all or part of a business problem. Projects are created using tools contained within the Enterprise Designer such as the Connectivity Map Editor, shown in Figure 5.

Figure 5   Connectivity Map Editor
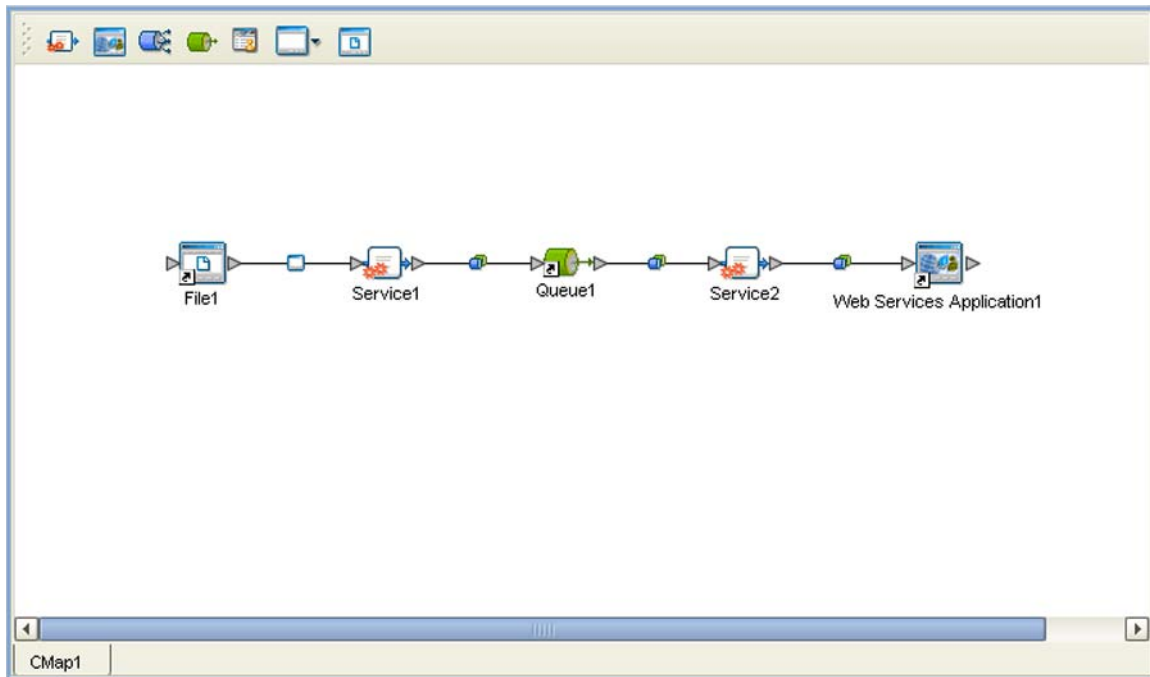
The Connectivity Map contains information about how components are connected. Connectivity Maps include one or more services (such as Collaborations), queues, topics, external applications, and eWays. An eWay is a component that connects eGate and an external business application. These components implement high-performance, distributed exchange of data with external systems.

2.4.1 Project Components

External Application Proxies

The basic purpose of eGate Integrator is to facilitate the interchange of data between external business applications. These business applications are collectively referred to as *external applications*, and are represented in the Project by logical proxies for the specific applications involved. An external application can be identified with an ERP application such as SAP or PeopleSoft, a DBMS such as Oracle or SQL, or with a particular communications protocol, such as TCP/IP or HTTPS.

Collaborations

A *Collaboration* is a logical operation taking place between a message destination and an external application, two message destinations, or two external applications, having a publication and subscription relationship with those entities. The operation is defined by a Collaboration Definition, which can be encoded in either Java or XSLT.

Collaboration Definitions

A Collaboration Definition defines the logical operation taking place in the related Collaboration. It is created in either the Java Configuration Editor or the XSLT Configuration Editor, and is based on an Object Type Definition.

Object Type Definitions

Object Type Definitions (OTDs) are a set of rules that define the encoding of an object. They describe:

- Messages that are propagated through eGate, and the methods available for operating on them.
- Interactions with external APIs.

OTDs can function as instances in eGate Collaboration Definitions, or as services in eInsight Business Process Manager. They are created using the OTD Wizard and customized in the OTD Editor.

Message Destinations

A Message Destination is a container for stored data, and can follow either the topic or queue JMS model.

Topics

A topic is a message destination that conforms to the publish-and-subscribe messaging paradigm.

Queues

A queue is a message destination that conforms to the point-to-point messaging paradigm.

Logical Connections

eWays

An eWay Intelligent Adapter provides the logical connection between external systems (applications or protocols) and a service. The identity of the External Application placed in a Connectivity Map dictates the type of eWay adapter assigned to the link between it and a Service such as a Collaboration, and can be accessible via multiple transport mechanisms.

JMS Clients

The JMS Client Connection configuration specifies the connection properties for the linked service (for example, publisher or subscriber).

2.5   User Interfaces

eGate Integrator provides two basic graphical user interfaces (GUIs), each of which addresses a different set of users. Enterprise Manager is an interface used by the entire ICAN Suite, the primary users of which are system administrators. Enterprise Designer is used by personnel who are involved in defining a software system for integrating the various enterprise applications using eGate and other ICAN Suite products.

2.5.1   Enterprise Manager

Enterprise Manager is a Web-based application that works within Microsoft Internet Explorer. It is used throughout the SeeBeyond ICAN Suite for:

- Installing and updating ICAN Suite products
- Accessing ICAN Suite product documentation
- Managing and monitoring runtime components

Figure 6 shows the Enterprise Manager **Admin** page, used in product installation.

Figure 6   SeeBeyond Enterprise Manager
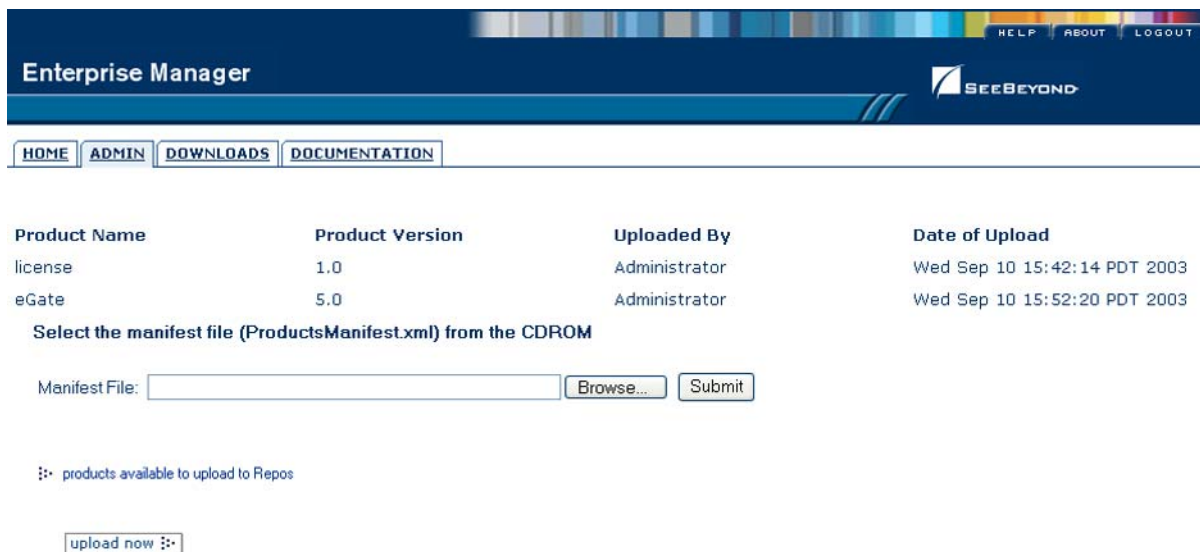
2.5.2. **Enterprise Designer**

The SeeBeyond Enterprise Designer is used to create and configure the logical components and physical resources of an eGate Project. Using this graphical user interface (see Figure 7), you can develop Projects to process and route data through an eGate Integrator system. The Connectivity Map Editor shown in the figure is only one of several different editors provided in Enterprise Designer.

Figure 7 Enterprise Designer
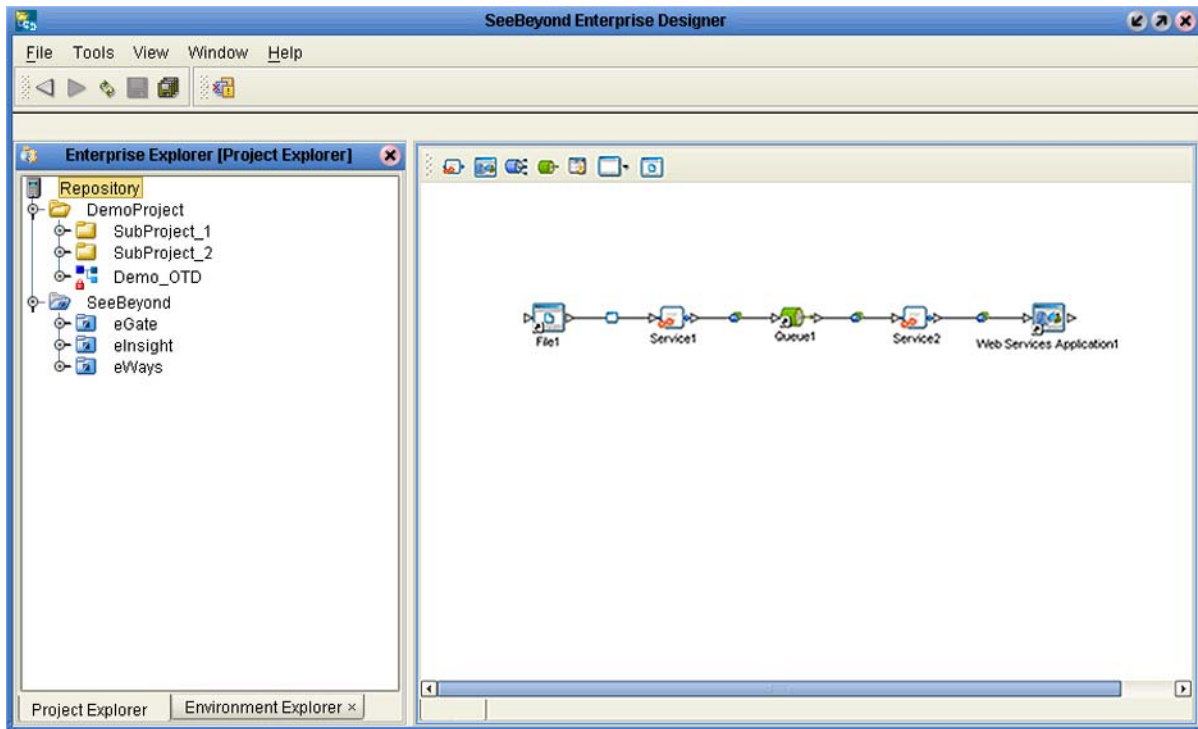
Enterprise Designer is used by other components of the ICAN Suite, such as eInsight and eXchange, as well as eGate Integrator. Examples of Enterprise Designer used with these ICAN products can be found in the respective sections of this Primer.

eInsight Business Process Manager

This chapter describes the features and functionality of the eInsight Business Process Manager (eInsight).

What's in This Chapter

3.1 Summary of Features

eInsight provides your business with a powerful assortment of features:

- Maximizes business process efficiency by enabling business process owners to directly model, monitor, manage, analyze, and optimize business processes using an easy-to-use, drag and drop graphical user interface.

- Manages long-lived business processes and ensure process integrity, including the ability to compensate for failed processing steps.

- Abstracts the complexities of the technical integration using open standards (BPMN) for the graphical notation of a business process elevating the business logic into the process layer to ensure a flexible, business process-driven implementation.

- Automates Web Services orchestration implementing BPEL4WS to assemble Web Services into larger composite application processes.

- Automatically provides all the inter-operability benefits of Web Services standards without requiring developers to learn SOAP, WSDL, UDDI and BPEL4WS.

- Guarantees process integrity and eliminates processing errors by ensuring every step in the business process either completes successfully with full traceability and auditability or is handled by robust workflow and exception handling functionality.

- Accelerates decision making and human involvement through robust workflow support including support for custom task assignment, user roles, and organizational hierarchies.

3.2   eInsight and the ICAN Suite

eInsight Business Process Manager (eInsight) is part of the SeeBeyond ICAN Suite of products. eInsight delivers business process management features and functions to the ICAN Suite.

Business process management is a strategic orchestration of the movement of information and the flow of complex processes between participants (systems, users, and organizations) to accomplish larger business objectives.

3.2.1. ICAN Integration

eInsight is tightly integrated with the ICAN Suite and runs as a component within the ICAN Suite environment. Figure 8 illustrates some of the eInsight and ICAN components that work together.

Figure 8   eInsight and the ICAN Suite
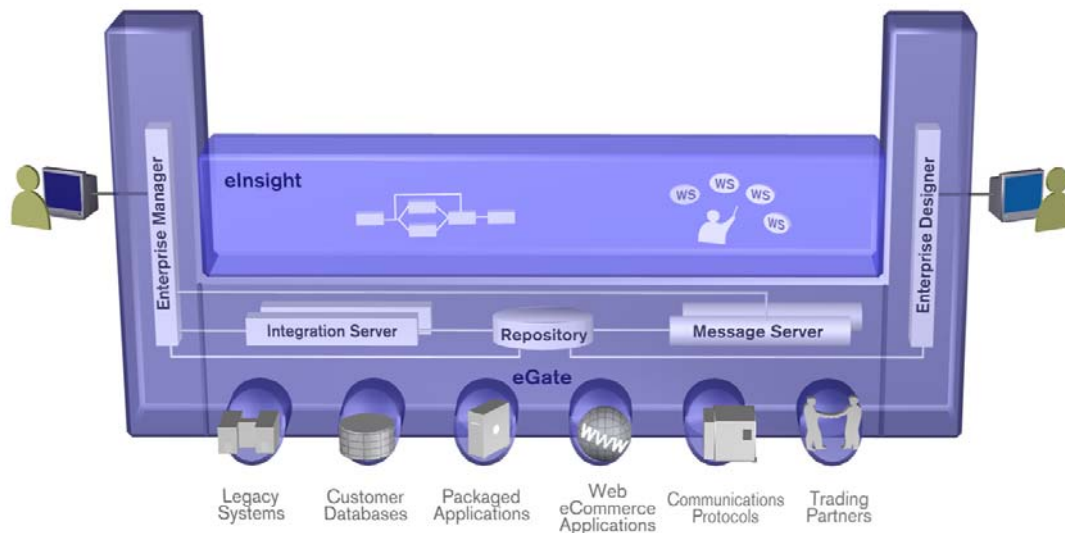
- The Business Process Designer runs as a component within the **Enterprise Designer**.

- Business process definitions, components, and deployment profiles are stored in the **SeeBeyond Repository**.

- The eInsight Engine, which coordinates all business process related activity of a deployed project, runs within the **SeeBeyond Integration Server.**

- Web-based business process monitoring is available throughout the **Enterprise Manager**, which provides an interface to access current instance data.

3.2.2. ICAN Suite Services

The ICAN Suite has a wide range of functions that it shares with all of the ICAN Suite products. eInsight is able to leverage many platform level services, such as:

- **Resource Management** - The ICAN Suite uses a distributed and open architecture that enables components to access system resources (memory and processing power) as needed and in conjunction with other components.

- **Security** - The ICAN Suite provides a security module to fulfill security needs such as authentication and authorization access to eInsight functions.

- **Repository storage and access** - The setup, component, and configuration information for the elements of a Project, including business process and related components, are stored in the Repository.

- **Deployment abilities** - The ICAN Suite provides deployment profiles that contain the information necessary to activate eInsight Project business processes and associated components. When a deployment profile is activated, eInsight's active business processes are made available as Web Services.

- **Monitoring** - The Enterprise Manager lends Web-based monitoring abilities to eInsight, allowing you to observe and correct business process activity.

- **Connectivity Mapping** - The Connectivity Map maintains the relationships between eInsight and other system components. The Connectivity Map specifies the topology of services that will be invoked, by:

  - Identifying the nature of services that are invoked.

  - Depicting relationships between the components, including the publish/subscribe information for data routing.

  - Defining the partners fulfilling the services that are invoked.

- **Version Control** - This feature maintains a history of business process versions, through a check-in and check-out process.

- **Impact Analysis** - Impact Analysis allows you to view how changes to one component or business process will impact other components or business processes of a Project or all Projects in the Repository.

- **Import and Export of Business Process Models** - The ability to import and export business process models makes it possible to recreate the processes on other systems or to reuse processes that may be similar in other areas.

3.3 eInsight Overview

eInsight provides you with a clear view into the internal and external processes of an organization. These processes may be executed by computer systems or employees.

3.3.1. **eInsight Architecture**

eInsight speeds the design and deployment of business processes by providing an open process modeling environment using BPMN (Business Process Modeling Notation) for the graphical notation of a business process and by automatically generating the BPEL4WS code needed to implement the business process across all of the participating Web Services (for example, for applications and business partners).

You drag and drop components into the process model and then specify the additional flow control and business rules that manage what services get called at what time. eInsight supports importing and exporting BPEL4WS to share processes with third party tools.

The technologies that carry out eInsight's functions are entirely based on industry standards. In Figure 9, you can see how the different components interact.

Figure 9   eInsight Architecture
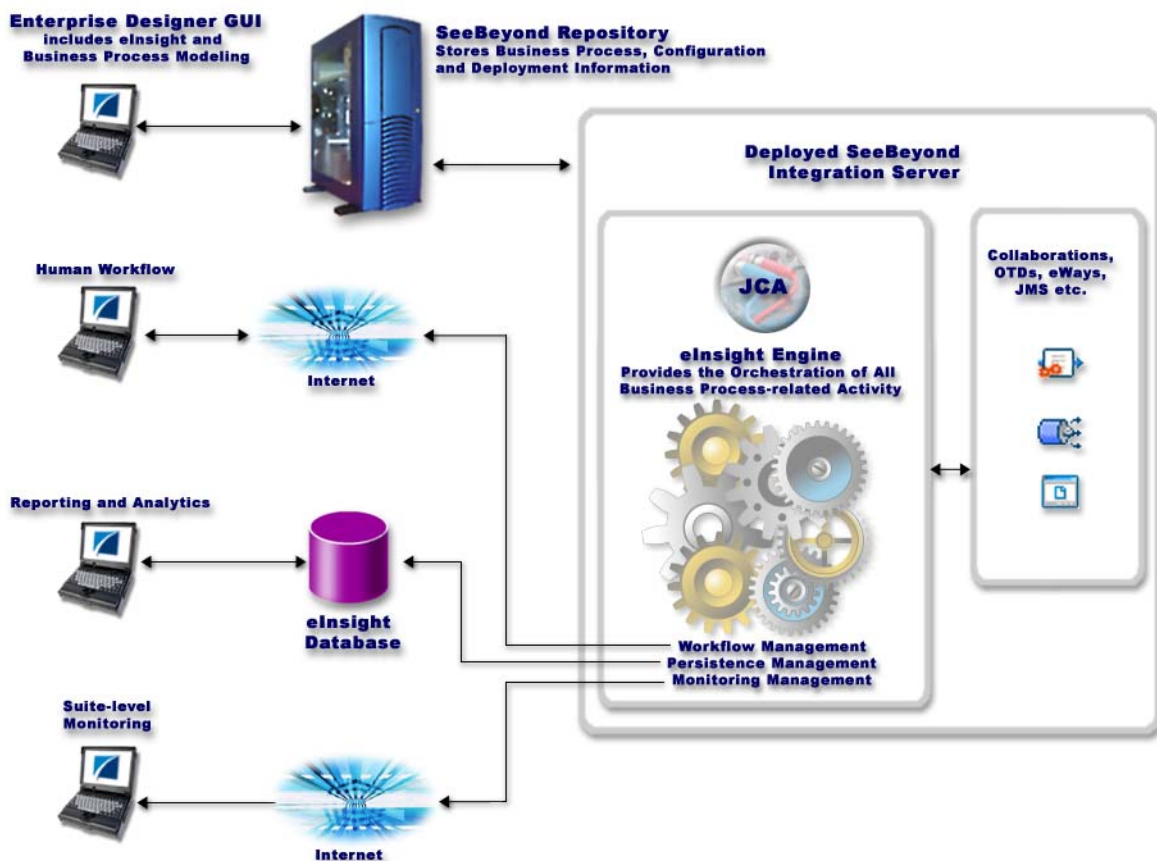
eInsight's architecture uses the following standards:

- Business Process Modeling Notation (**BPMN**), from the Business Process Management Initiative (BPMI) standards body, provides a standard graphical view for Business Process Execution Language for Web Services (**BPEL4WS**).

- Web Services Business Process Execution Language (**BPEL4WS**) is the underlying code generated when creating a business process.

- Web Services Description Language (**WSDL**) is an XML-based language used to define Web Services and describe how to access them. All eInsight business processes are automatically described using generated WSDL.

- J2EE Connector Architecture (**JCA**) provides a mechanism to access external applications and data. The JCA engine is implemented as a standard JCA 1.5 module that plugs into the SeeBeyond Integration Server.

3.4 Process Overview

There are two phases of business process management. The first phase, *design*, is described in the **"Design Phase Overview"**. The design phase begins before you start using eInsight and ends once the business process is deployed.

The second phase is called *runtime*, which is discussed in the section **"Runtime Phase Overview"**. Runtime refers to the tasks that you perform after the business process is deployed.

Figure 9 illustrates the eInsight and ICAN components that work together to provide integration services.

3.4.1. Business Process Modeling and Design

You can use eInsight to streamline operations by creating business logic that helps you reach outward to include customers and trading partners. Using eInsight to implement business process management removes inefficiencies by orchestrating a unified work flow. This flow can include multiple systems/users, therefore extending to customers.

The eInsight graphical user interface (GUI), shown in Figure 10, allows you to model the business processes that your department or even your entire company performs on a regular basis. The tools provided allow for various scenarios and events that may take place in your process.

3.4.2. Business Process Designer

The Business Process Designer serves as the front-end design tool used to create a visualization of your business workflow and increase understanding of the business processes involved. As the business user, you are able to integrate logic into the business process. When used with eInsight, the Enterprise Designer includes the following areas, as shown in Figure 10.

Figure 10   eInsight Business Process Designer

Project Explorer   Business Process Designer
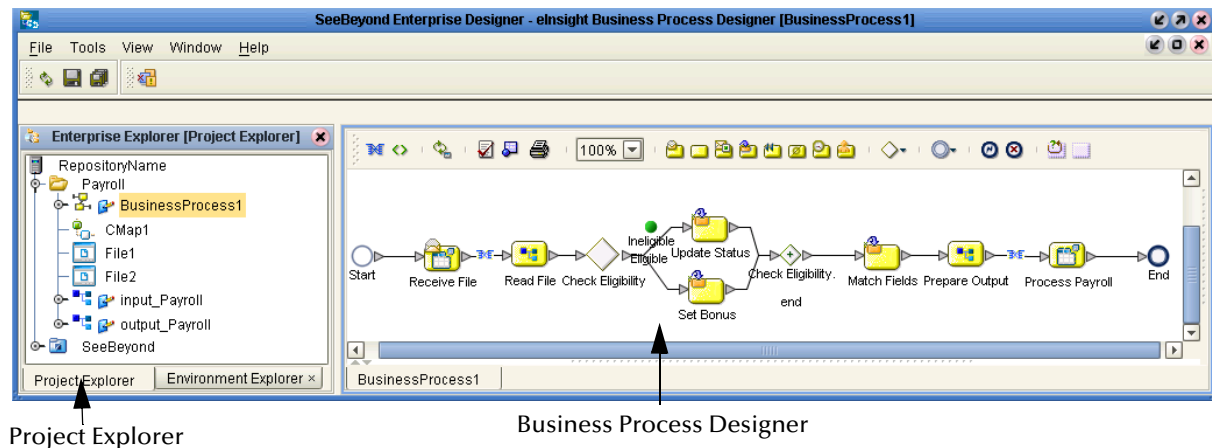
- **Project Explorer** - Displays a hierarchical representation of all the business process models and related ICAN components. This view shows what is currently displayed on the modeling canvas.

- **Business Process Designer** - Used to graphically create the business process model in the form of an activity diagram.

3.4.3. Design Phase Overview

The basic steps that you will perform to design a business process model are as follows:

1 Plan and design a model that represents a business process taking place in your company.

2 Set up a Project and related components necessary to your business process model.

3 Create the new business process model in the Business Process Designer, using activities, links, decision and exception handling logic, and any other modeling elements that express the actual business process.

4 Validate, generate, and save the business process code to the SeeBeyond Repository server, where business process configuration and deployment information is stored.

5 Create a Connectivity Map to configure the relationship between your ICAN Suite components.

6 Select or create an Environment where your business process will run.

7 Select or create a Deployment Profile and activate it to complete the deployment process. The eInsight Engine is now deployed to the Integration Server.

Advanced design phase tasks

There are other tasks you may need to do in the design phase, based upon your configuration, such as:

- Create human workflow tasks using an eVision interface and User Activity element. These custom Web pages can be configured to interact with business processes or

track and view system exceptions or employee assignments. Human workflow are the actual tasks and assignments that a person performs.

◆ eInsight supports the different ways an organization defines its company structure. You may define expression-based task assignments to users, groups, and/or roles.

▪ Configure and maintain persistence and recoverability functions. The instance data is stored in a database that you configured in the design phase. The eInsight Engine writes to this database at runtime.

3.4.4. Runtime Phase Overview

Once all of the design phase tasks are complete and the system is running, you can monitor and manage business process activity and the overall Project with the Enterprise Manager. These tasks are only available if you use a database.

eInsight Engine

The eInsight engine provides process coordination that enables the execution of business processes, activities, and tasks. During the runtime phase, the eInsight engine:

▪ Receives messages that instantiate business process instances.

▪ Writes monitoring, persistence, and recoverability data to a database (if available).

See **Figure 9 on page 26** for a detailed view of the eInsight architecture.

eInsight Database

eInsight components connect to the optional database to provide:

▪ **Persistence** - The eInsight Engine writes instance data to a database to ensure that data is able to persist in the system.

▪ **Recoverability** - Using a database allows you to recover data from the last state of consistency.

▪ **Monitoring** - Instance data is written to a database and then read by the Enterprise Manager to provide current and historical system information.

3.4.5. Business Process Monitoring and Management

Once business processes are up and running, the Enterprise Manager provides visibility into the state of each business process activity. This interface allows organizations to monitor and manage all aspects of the business process.

Figure 11   Business Process Monitor
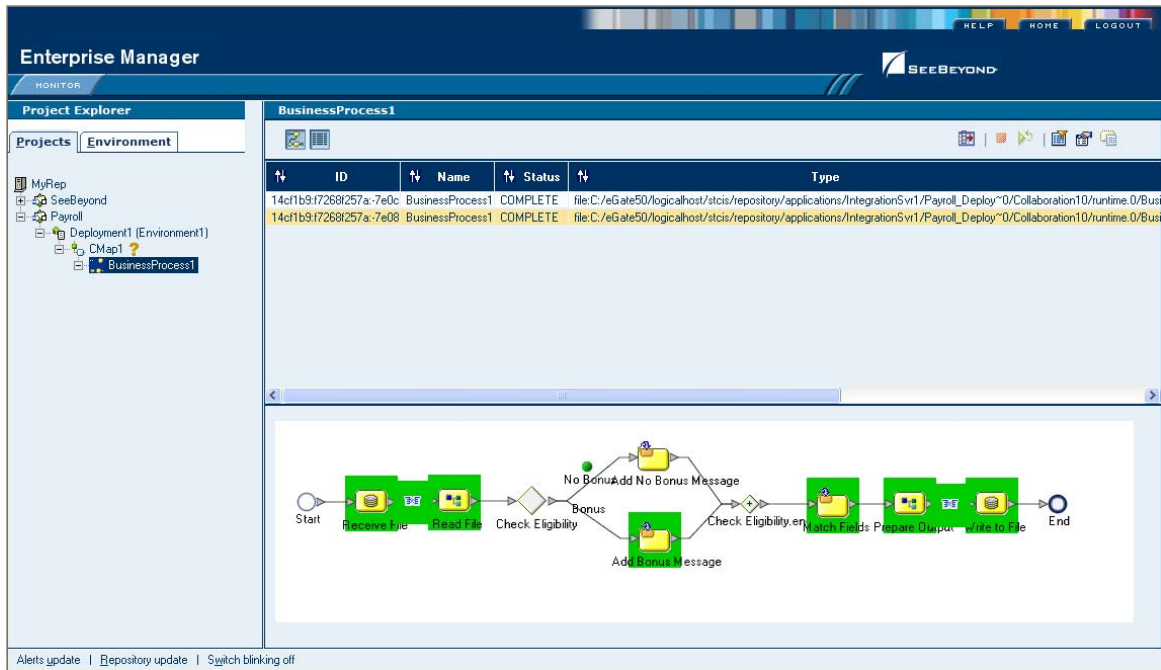
The actions that you can perform on a business process from the Enterprise Manager
are called *Business Process Management* tasks. Some actions that you can take from the
Enterprise Manager include viewing and identifying errors originating from erroneous
data. You can ensure that processes continue to run and work properly with these
powerful monitoring tools.

eInsight Enterprise Service Bus

This chapter describes the features and functionality of the eInsight Enterprise Service Bus (eInsight ESB).

What's in This Chapter

4.1 Summary of Features

eInsight ESB provides the following additional features:

- Streamlines basic integration services needed by small to medium size enterprises in a cost-effective way as eInsight ESB creates business logic that helps organizations integrate and orchestrate Web Services as well as proprietary or legacy applications

- Allows business analysts to model, in a graphical drag and drop environment, the business processes that their department or even entire company performs.

- eInsight ESB is tightly integrated with the ICAN Suite and leverages over 80 packaged SeeBeyond eWay™ Intelligent Adapters that provide out-of-the-box connectivity to a variety of systems, applications, databases, and legacy technologies.

- Provides graphical transformation capabilities to drag-and-drop and visualize the mapping of data transformations between the systems being integrated.

- Uses a distributed and open architecture that enables components to access system resources (memory and processing power) as needed and in conjunction with other components.

- Fulfills security needs such as authentication and authorization access to eInsight ESB functions.

- Stores in the Repository the setup, component, and configuration information for the elements of a Project, including business process and related eInsight ESB components.

- Provides extensive control of deployment profiles, which contain the information necessary to activate a Project business processes and associated components. When a deployment profile is activated, active business processes are made available as Web Services and published to SeeBeyond's UDDI Registry.

- Provides Web-based monitoring abilities to observe and correct business process activity, using Enterprise Manager.

- Provides connectivity mapping - The Connectivity Map maintains the relationships between eInsight and other system components. The Connectivity Map specifies the topology of services that will be invoked by:

  - Identifying the nature of services that are invoked.

  - Depicting relationships between the components, including the publish/ subscribe information for data routing.

  - Defining the partners fulfilling the services that are invoked.

- Provides check-in/check-out version control by maintaining a history of business process versions.

- Provides Impact Analysis, allowing you to view how changes to one component or business process will impact other components or business processes of a Project or all Projects in the Repository.

- Provides for import and export of business process models, making it possible to recreate the processes on other systems or to reuse processes that may be similar in other areas.

4.2   About eInsight ESB

eInsight Enterprise Service Bus (ESB) is a streamlined, distributed integration platform that combines Web Services support, transformation, and content-based routing. eInsight ESB is a rapidly implemented version of eInsight Business Process Manager.

The Enterprise Service Bus edition provides a lower-cost alternative to deploy integrated business processes as composite applications built on a services-oriented architecture. eInsight Enterprise Service Bus is sold standalone and supports SeeBeyond applications and eWay Intelligent Adapters, which can be purchased separately as plug-ins to eInsight ESB to access external applications and data.

4.2.1. ICAN Integration

eInsight ESB is tightly integrated with the SeeBeyond® Integrated Composite Application Network (ICAN) Suite™ and is compatible with the other ICAN suite products (also, see Figure 12), including:

- eVision Studio

- ePortal Composer

- eBAM Studio

- eTL Integrator

- eView Studio

- SeeBeyond eWay™ Intelligent Adapters

Thus, as your organization's integration needs grow, you can leverage and reuse the eInsight ESB integration components that you have already deployed.

Figure 12   eInsight ESB and the ICAN Suite
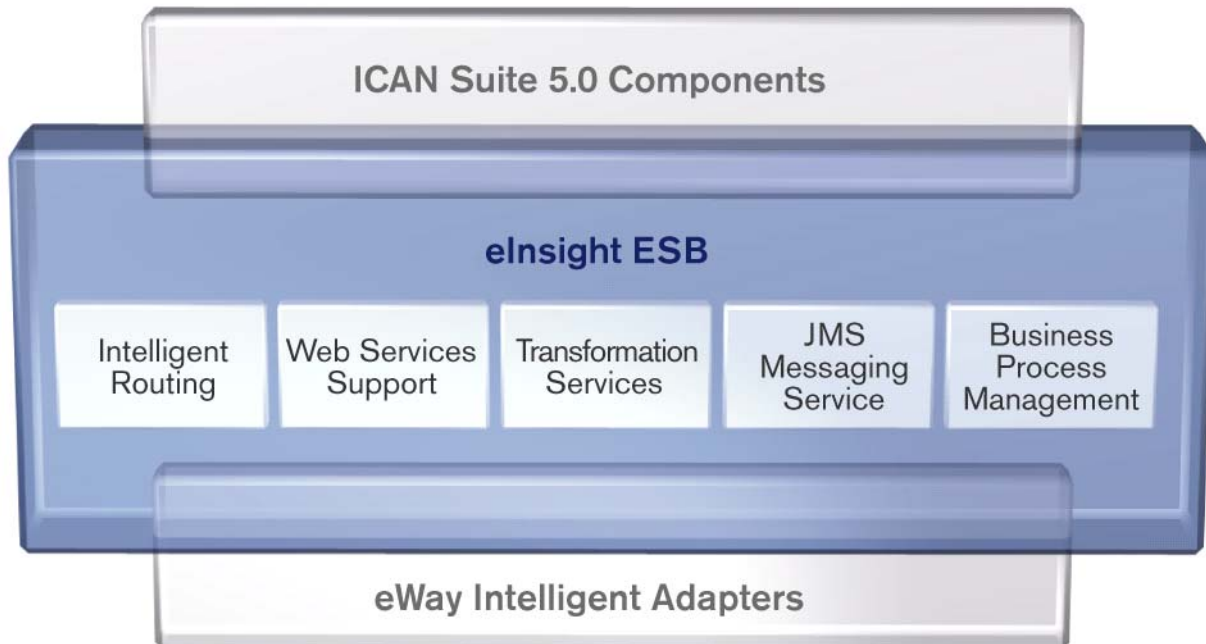
4.2.2. eInsight ESB Architecture

eInsight ESB speeds the design and deployment of business processes by providing an open process modeling environment using BPMN (Business Process Modeling Notation) for the graphical notation of a business process and by automatically generating the BPEL4WS code needed to implement the business process across all of the participating Web Services (e.g. for applications and business partners).

Business analysts drag and drop components into the process model and then specify the additional flow control and business rules that manage what services are called and when. eInsight ESB supports importing and exporting BPEL4WS to share processes with third party tools.

The technologies that carry out eInsight ESB business process functions are entirely based on industry standards. eInsight ESB uses the following standards:

- Business Process Modeling Notation (**BPMN**), from the Business Process Management Initiative (BPMI) standards body, provides a standard graphical view for Business Process Execution Language for Web Services (**BPEL4WS**).

- Web Services Business Process Execution Language (**BPEL4WS**) is the underlying code generated when creating a business process.

- Web Services Description Language (**WSDL**) is an XML-based language used to define Web Services and describe how to access them.

- J2EE Connector Architecture (**JCA**) provides a mechanism to access external applications and data. The JCA engine is implemented as a standard JCA 1.5 module and it plugs into the SeeBeyond Integration Server.

4.3 Process Overview

4.3.1. Business Process Modeling and Design

You can use eInsight ESB to streamline operations by creating business logic that helps you reach outward to include customers and trading partners. Using eInsight ESB to implement business process management removes inefficiencies by orchestrating a unified work flow. This flow can include multiple systems/users, therefore extending to customers.

eInsight ESB allows you to model the business processes that your department or even your entire company performs on a regular basis. The tools provided allow for various scenarios and events that may take place in your process.

4.3.2. Design Phase Overview

The basic steps that you will perform to design a business process model are as follows:

1 Plan and design a model that represents a business process taking place in your company.

2 Set up a Project and related components necessary to your business process model.

3 Create the new business process model in the eInsight Business Process Designer, using activities, links, decision and exception handling logic, and any other elements that express the actual business process.

4 Validate, generate, and save the business process code to the Repository, where business processes configuration and deployment information is stored.

5 Create a Connectivity Map to configure the relationship between your components.

6 Select or create an Environment where your business process will run.

7 Select or create a Deployment Profile and activate it to complete the deployment process.

4.3.3. Runtime Phase Overview

Once all of the design phase tasks are complete and the system is running, you can monitor and manage business process activity and the overall Project with the Enterprise Manager. These tasks are only available if you use a database.

eInsight ESB Engine

The eInsight ESB Engine provides process coordination that enables the execution of business processes, activities, and tasks. During the runtime phase, the eInsight ESB engine:

- Receives messages that instantiate business process instances.
- Writes monitoring, persistence and recoverability data to a database (if available).

See **Figure 12 on page 33** for a detailed view of the eInsight ESB architecture.

eInsight ESB Database

eInsight components connect to the optional database to provide:

- **Persistence** - The eInsight Engine writes instance data to a database to ensure that data is able to persist in the system.
- **Recoverability** - Using a database allows you to recover data from the last state of consistency.
- **Monitoring** - Instance data is written to a database and then read by the Enterprise Manager to provide current and historical system information.

4.3.4. Real-Time Business Process Monitoring and Alerting

Once the automated business processes are up and running, the Enterprise Manager provides a dashboard delivering instant visibility into the state of each business process activity. This allows organizations to monitor all aspects of both internal processes and those involving B2B trading communities. Processes are normally monitored for exceptions using business rules that alert managers as needed. This proactive alerting enables managers to respond instantly to business events such as increased customer demand, inventory shortages, and quality problems.

eVision Studio

This chapter describes eVision Studio (eVision), a graphical design studio for the WYSIWYG creation of integrated Web applications.

What's in This Chapter

5.1   Summary of Features

eVision Studio is a graphical environment that enables business analysts and Web developers to rapidly create interfaces for composite applications, without the need for advanced programming abilities. When drag-and-drop tasks are performed using the eVision tools, the underlying programming code is automatically generated.

eVision provides two main tools: the Page Layout Designer and the Page Flow Designer. Both tools are integrated within Enterprise Designer.

The Page Layout Designer is used to create Web pages. This tool includes a design canvas, GUI component palettes, and a properties window. Among the components in the palettes are check box groups, text boxes, hyperlinks, image maps, progress bars, submit buttons, and charts. The properties window allows the developer to specify a variety of attributes for the GUI components and the overall Web page.

The Page Flow Designer is used to manage the Web pages within a composite application. The Web pages are combined with specialized elements to create a Page Flow. Page Flows can include data mappings from one component to the next. These mappings are configured using a business rule designer.

eVision can be used in a variety of ways as a front end to SeeBeyond ICAN Suite integration solutions. For example:

- In conjunction with eInsight Business Process Manager, eVision allows users to participate in business processes. The users perform workflow tasks with eVision Web pages that are tailored for specific organizational roles.

- eVision can provide the login pages in which users enter their user name and password, which are then authenticated.

5.2   eVision and the ICAN Suite

eVision is part of the SeeBeyond ICAN Suite of products. eVision provides Web application design and deployment features and functions to the ICAN Suite.

eVision simplifies the task of developing Web applications by using patterns and metaphors that are familiar to Web developers, providing graphical abstractions of back-end data, and modeling of user/system interactions.

5.2.1   ICAN Suite Integration

eVision is tightly integrated with the ICAN Suite and runs as a component within the ICAN Suite environment. The Page Layout Designer and Page Flow Designer run as components within Enterprise Designer. eVision Web applications are stored in the SeeBeyond Repository.

Figure 13 shows how eVision is integrated with the ICAN Suite.

Figure 13   eVision Studio Integration With the ICAN Suite
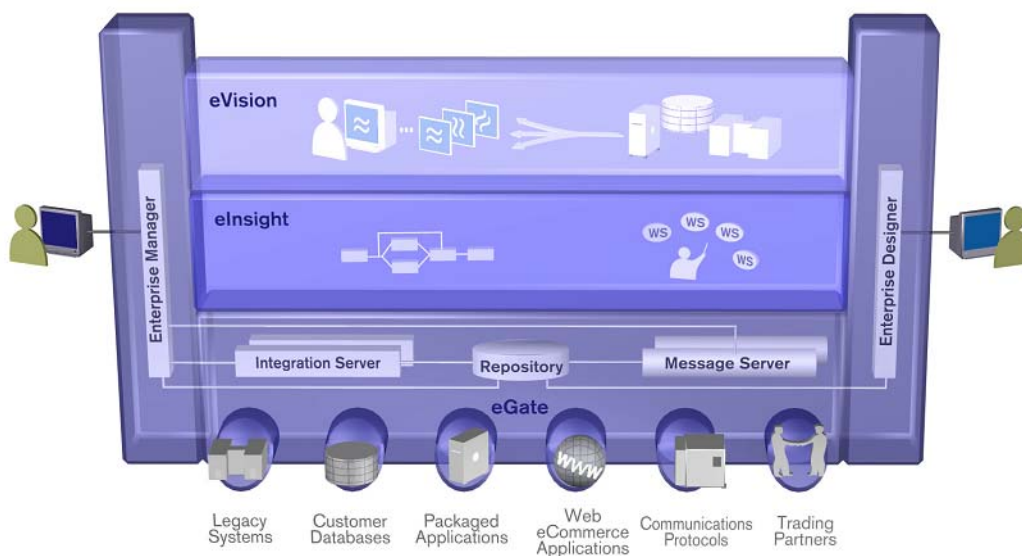
- eVision Studio runs as a component within **Enterprise Designer**.

- Page Layouts, Page Flows, GUI components, and Deployment Profiles are stored in the **SeeBeyond Repository**.

- The Page Flow Engine, which coordinates all process-related activity of a deployed project, runs in a **SeeBeyond Integration Server**.

5.3   eVision Overview

eVision applications receive data from and send data to ICAN Suite components, allowing users to interact with the data at run time. Web applications enable the distribution of integrated business processes across the enterprise and allow real-time user interaction with those processes. eVision allows the organization to present a single, unified view of enterprise data and applications to employees, customers, and partners.

With eVision, the Web developer can create personalized views of business information, allowing Web application users to interact with running business processes in real time, while working with only the information that they need to see. Web applications can be easily structured to allow employees to see what tasks are assigned to them, then to interact with and complete the tasks.

5.3.1 MVC Architecture

eVision applications are based on the Model/View/Controller (MVC) architecture. MVC is a software development paradigm that enhances the task of building software systems, particularly those that generate multiple, synchronized presentations of the same data. For example, MVC is ideal for the development of a graphical statistical presentation application that requires simultaneous rendering of the same data in bar, line, and pie chart formats.

The MVC architecture consists of three types of objects: the Model, the View, and the Controller.

Figure 14   Model/View/Controller (MVC) Architecture

The **Model** object represents the data in a program, which manages behaviors and data within an application. The Model responds to requests for information about its current state (typically requested by the View) and responds to instructions to change its state (typically requested by the Controller).

The **View** object manages the visual display of the Model data; for example, displaying graphics and text to users in a browser.

The **Controller** object enables user interaction with the Model data; for example, mouse and keyboard inputs from the user, which instruct the Model and/or View to perform an action.

eVision's application architecture fully supports the MVC paradigm. In an eVision Web application, user input, modeling of the external world, and visual feedback are managed by MVC objects, where each object is specialized for its task. For example:

- The **Model**, represented by the **Page Flow**, contains the business logic (Object Type Definitions and Collaborations) that interacts with the back-end system applications.

- The **View** contains the JavaServer Pages (JSPs) that are generated with the **Page Layout Designer**.

- The **Controller** is the Web-enabled Page Flow created with the **Page Flow Designer**. The Controller orchestrates the sequence of pages being sent to the browser in response to user actions.

5.4   Process Overview

The steps for the Web application development process are:

- **Page Layout** - Using the **Page Layout Designer** to create Web pages with pre-built GUI components.

- **Page Linking** - Using the **Page Link Wizard** to create links to Web pages and pass parameters from one page to another.

- **Page Flow** - Using the **Page Flow Designer** to connect finished Web pages to create a logical flow for the underlying business process.

- **Binding and Deployment** - Deploying Web applications into an Environment for integration with other ICAN Suite run time components.

5.4.1 User Interface Components

eVision Studio leverages the familiar Enterprise Designer interface to create, manage, deploy, and integrate Web applications. See Figure 15.

Figure 15 Page Layout Designer in Enterprise Designer
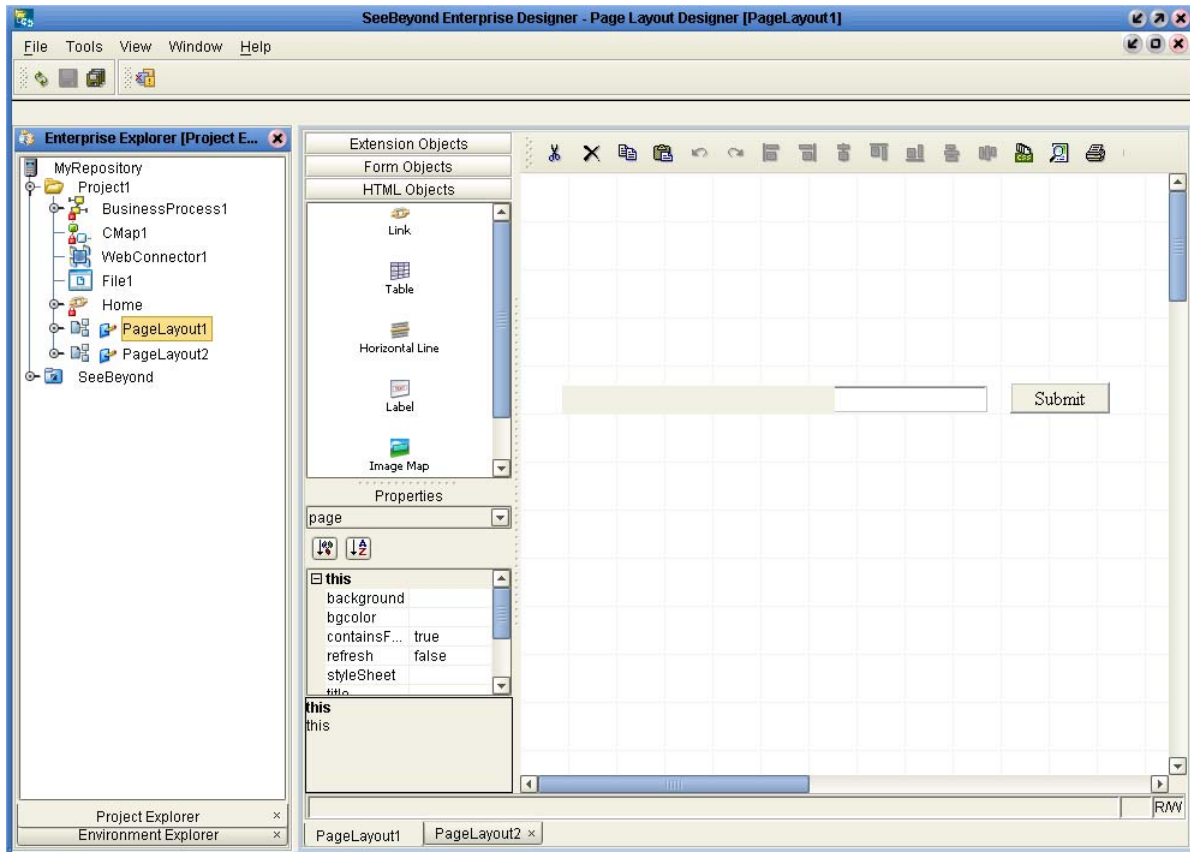
Page Layout Designer

The Page Layout Designer allows the Web designer to specify the details of individual pages in the application using familiar drag-and-drop techniques to place GUI components (Web page design objects) onto a design canvas.

The Page Layout Designer provides a comprehensive collection of pre-built GUI components, and a friendly, WYSIWYG Web page editor.

Page Flow Designer

The Page Flow Designer facilitates the flow of Web-based business activities. The functions include Page Flow modeling, monitoring, and execution, as well as the ability to analyze how data messages flow from page to page.

The Page Flow Designer allows the business analyst to lay out the user workflow of a Web application by modeling the high-level, logical, page-by-page flow that users will follow through the Web application to complete a given task (the *Page Flow*). The business analyst can easily lay out a Web application's page flow using the Page Flow Designer's set of intuitive, graphical modeling tools. At run time, the Page Flow drives the display of eVision Web pages and orchestrates the interactions with back-end systems.

Through the deployment of Web pages that are developed with the Page Layout Designer and Page Flow Designer, the business analyst can enable enterprise-wide, internal and external access to Web applications across an intranet or the Internet.

5.4.2 Page Flow Engine

The **Page Flow Engine** orchestrates the system responses to the execution of Web page component code. At run time, the Page Flow Engine executes page links, receives and processes user input, and, based on human interaction, moves the viewer from page to page until the underlying process is complete.

eXchange Integrator

This chapter describes the basic operation and features of the eXchange Integrator (eXchange) component of the ICAN product suite.

What's in This Chapter

6.1 Summary of Features

eXchange provides an open B2B protocol framework to support standard EDI and B2B business protocols and enveloping protocols. Not only does it support existing standard protocols, with an extensive set of prebuilt business protocol pipelines, it also provides the tools and framework to create and adopt new protocols and to build custom pipelines.

B2B modeling semantics are exposed so that business rules can be added and tailored to address the particular needs of each eBusiness challenge. The tight integration with the rest of the ICAN Suite provides validation, logging, and reporting capabilities, and because each logical step within any business rule is accessible anywhere along the entire business pipeline, the design tools provide complete end-to-end visibility.

The trading partner management facility, eXchange Partner Manager (ePM), is provided via a Web interface. For easy interoperability, trading partners can be configured by importing Collaboration Protocol Agreements (CPAs); or trading partner profiles can be configured manually. Each trading partner profile is identified by a unique ID determined by the enterprise, and delivery channels can be configured for acknowledgments, compression, industry-standard encryption and decryption, and nonrepudiation.

At run time, all steps in the business process, from initial receipt of the message to final delivery to the trading partner, are tracked in real time and also stored in the eXchange database. The Web-based message/package tracker provides tools for retrieving and filtering tracked message and envelope information. Used in conjunction with the other monitoring tools of the ICAN suite, this provides the enterprise with a complete solution for troubleshooting and managing all eBusiness activities.

6.2 eXchange and the ICAN Suite

eXchange is part of the SeeBeyond ICAN Suite of products. eXchange provides a Web-based trading partner configuration and management solution for automating and securely managing business partner relationships for real-time interaction between the enterprise and its partners, suppliers, and customers.

6.2.1 ICAN Integration

eXchange is tightly integrated with the ICAN Suite and runs as a component within the ICAN Suite environment. Figure 16 illustrates how eXchange and other ICAN Suite components work together.

Figure 16   eXchange and the ICAN Suite
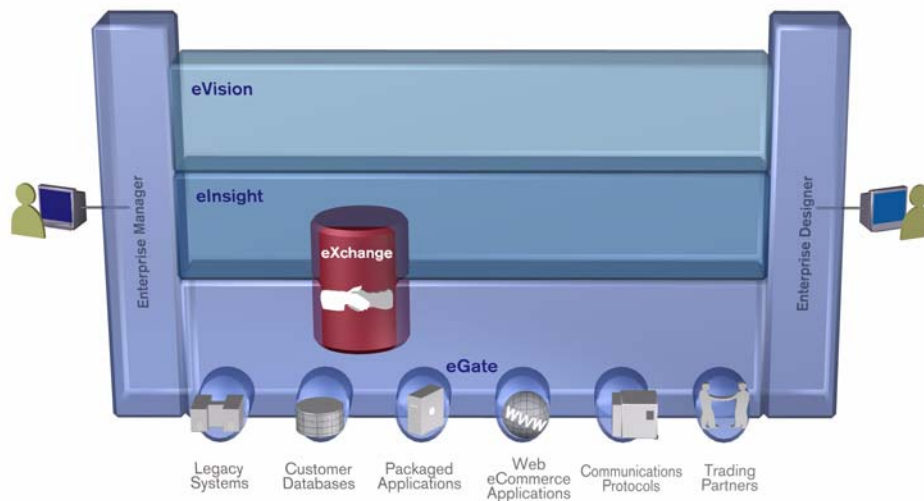
6.3 Architectural Overview

eXchange centers around the concept of a *Delivery Channel Profile* for each trading partner relationship. Delivery Channel Profiles are configured within eXchange for use by runtime components. Each profile specifies which B2B protocol(s) to use, where and how to receive inbound messages from trading partners, how to configure and secure messages in this channel, and how and where to deliver outbound messages to trading partners.

eXchange uses the following key components:

- **B2B Host Designer** — Using the **Enterprise Designer** GUI framework, eXchange provides an editor for setting up B2B environments, called the *B2B Host Designer*. Each B2B Host provides one or more protocol-specific delivery channels that are exposed to the eXchange database via the Repository. Delivery channels provided by the B2B Host can then be accessed by specific trading partners and reused. See **Figure 20 on page 46**.

- **B2B Services and Protocols** — eXchange provides two other special editors: The *eXchange Service Designer*, for modeling the choreography of B2B interactions between the internal system and an external trading partner, as mediated by the B2B Host; and the *eXchange Protocol Designer*, for setting up the message flow logic (*B2B protocol processes*) required to address a specific business challenge, using such activity elements as branching activities, timers, and exception handling. See **Figure 19 on page 45**.

  Prebuilt B2B protocol processes for such industry-standard B2B protocols as AS2 and ebXML are available as separately installable add-on products. eXchange also provides the flexibility of allowing the enterprise to create and configure custom protocol processes. See **Figure 19 on page 45**.

  eXchange also supplies **Channel Manager**, a collection of eXchange Services that provides trading partner–specific integration to the enterprise. Industry-standard transport protocols (FTP, HTTP, HTTPS, SMTP) are supported by Channel Manager and other eWays.

- **Trading Partner Configuration** — eXchange provides a Web-based GUI, eXchange Partner Manager (ePM), for configuring and managing B2B trading partners. Each trading partner has one or more delivery channels that specify the protocols to be used, with corresponding transport mechanisms—encryption parameters such as certificate, signature, and keystore information, acknowledgment-handling preferences, and so forth. See **Figure 21 on page 46**.

- **eXchange Database** — eXchange uses an Oracle database to mediate retrieval of trading partner information and to store run-time information on message tracking.

- **Message Tracking**— eXchange provides a specific MessageTracker application that can be combined with other processes in a project just by dragging it into the Connectivity Map, as well as a Web-based message tracking GUI with powerful filtering and searching capabilities. See **Figure 22 on page 47**.

The interaction of these components is illustrated in Figure 17.

Figure 17 eXchange Architecture
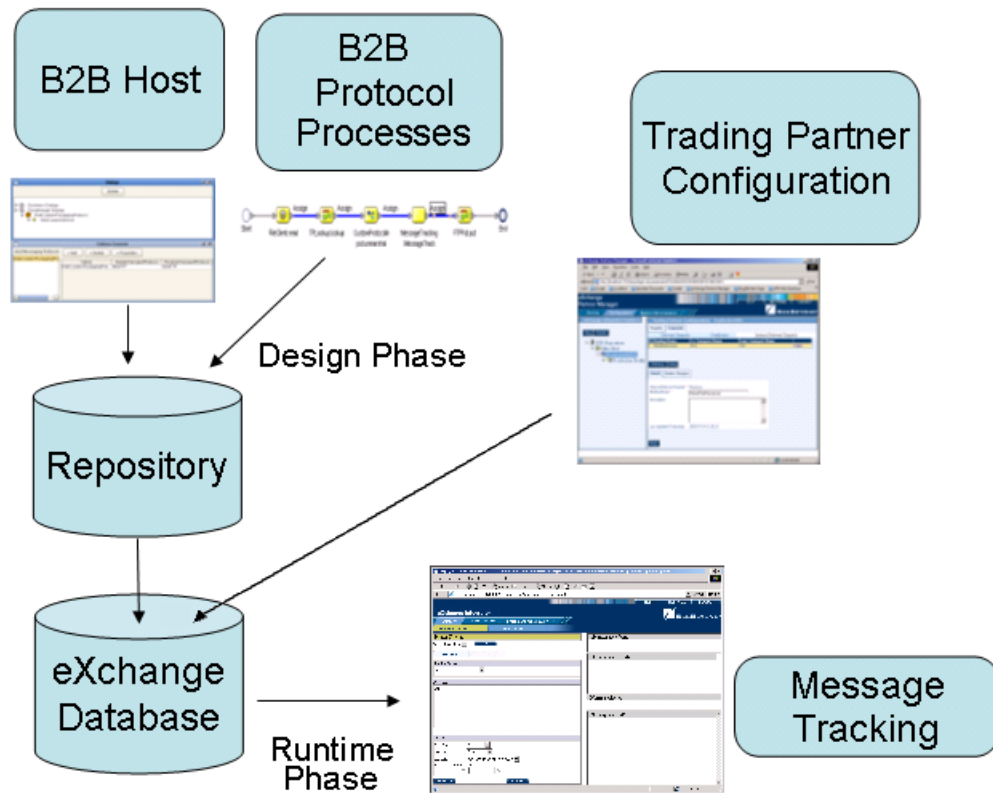
The illustrations in Figure 18, Figure 19, and Figure 20 indicate some of the features provided by the various GUIs.

Figure 18   User-Created B2B Protocol Process
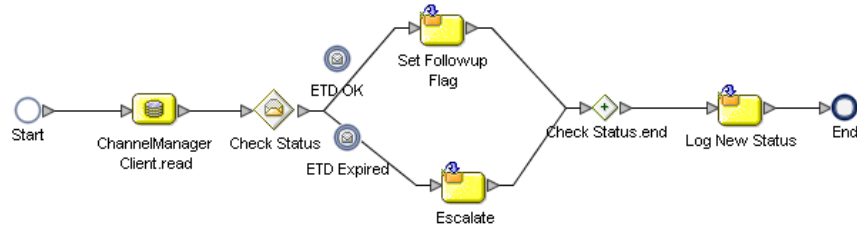
Figure 19   Prebuilt B2B Protocol Process (for AS2 Inbound)
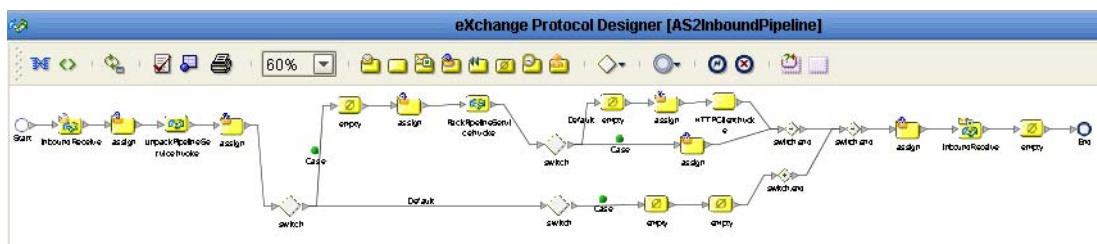
Figure 20   B2B Host Designer in Enterprise Designer
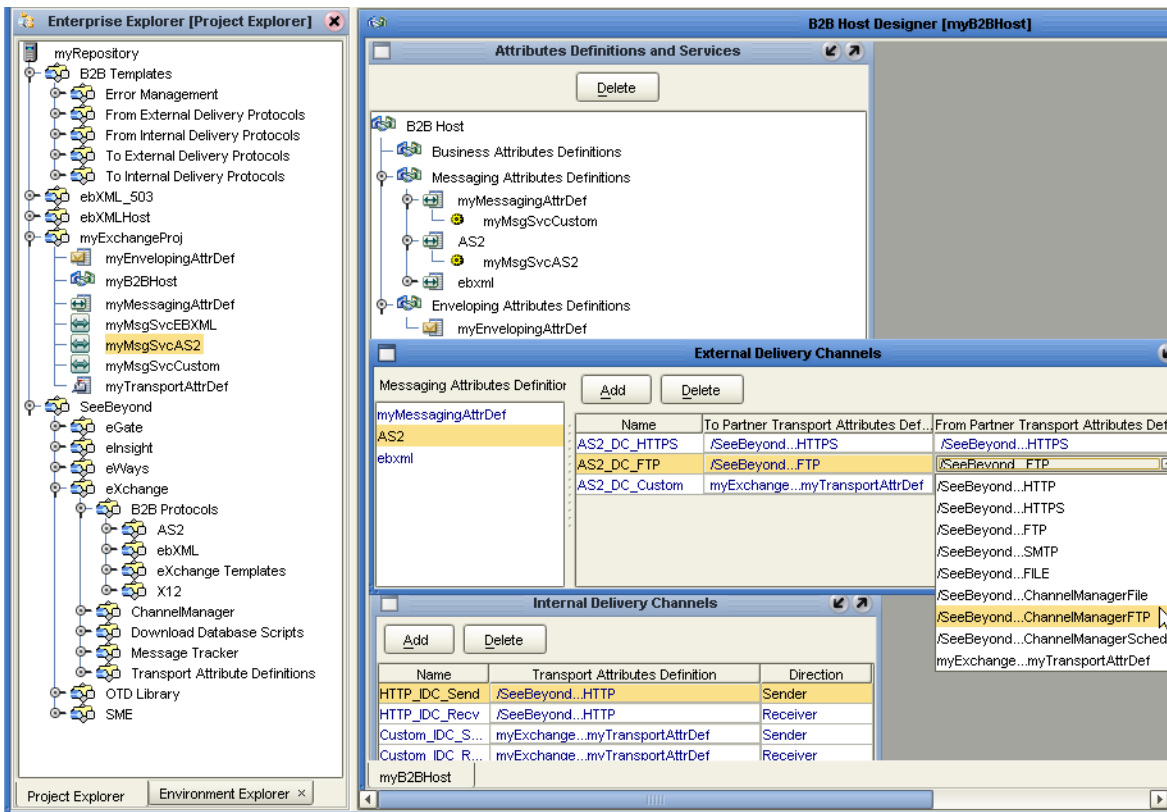
Figure 21   eXchange Trading Partner Configuration
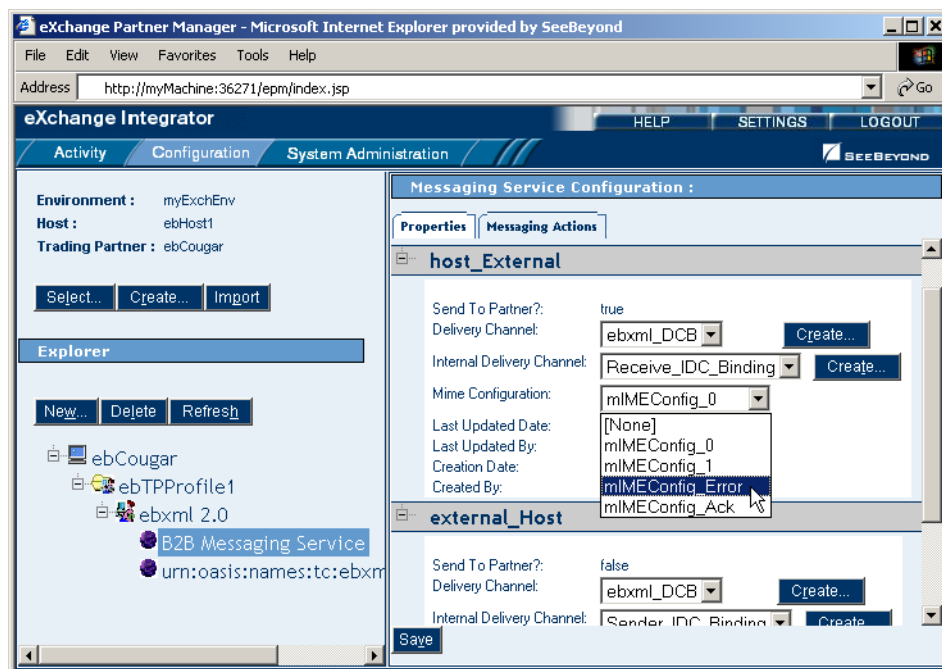
Figure 22   eXchange Message Tracking
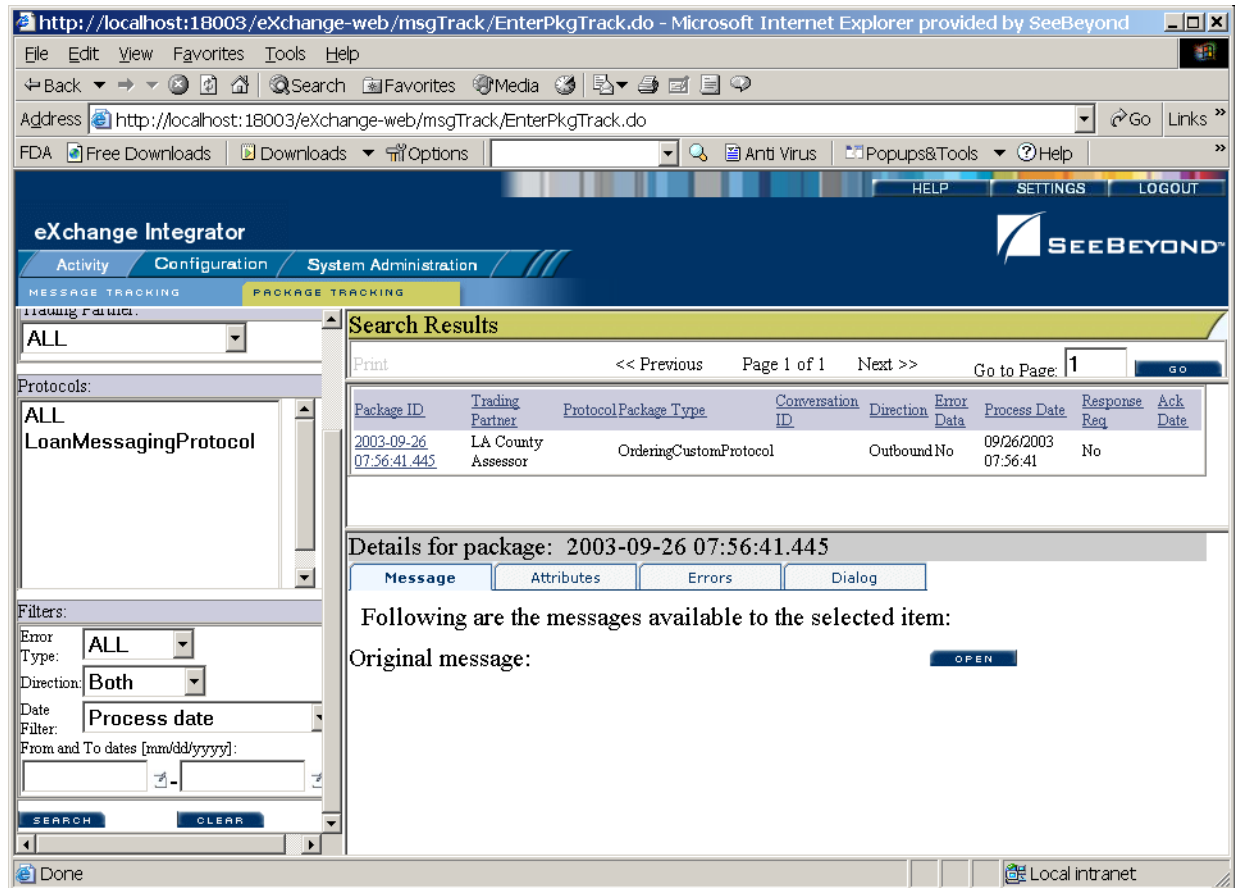
6.4   Process Overview

Using eXchange to create a business solution consists of three phases:

- Design phase within Enterprise Designer

- Design phase within eXchange Partner Manager

- Runtime phase

The purpose of the design phases is to: Create metadata for Delivery Channel Profiles; set up business logic for B2B protocol processes; configure connections with external systems; create and configure trading partners; and associate each trading partner relationship with a Delivery Channel Profile (DCP) configuration. Activating a trading partner exposes its DCP configuration settings to the eXchange database.

At run time, the Logical Host reads the DCP configuration from the database to determine: How to receive and process inbound messages; which business logic to run; and how to process and deliver outbound messages. Results are written to the database, where they can be filtered and viewed by the Message Tracker facility.

These phases are illustrated in Figure 23 and explained in further detail in the following sections:

- **Design Phase: Using Enterprise Designer** on page 48
- **Design Phase: Using eXchange Partner Manager** on page 49
- **Runtime Phase** on page 49

Figure 23 eXchange Architecture
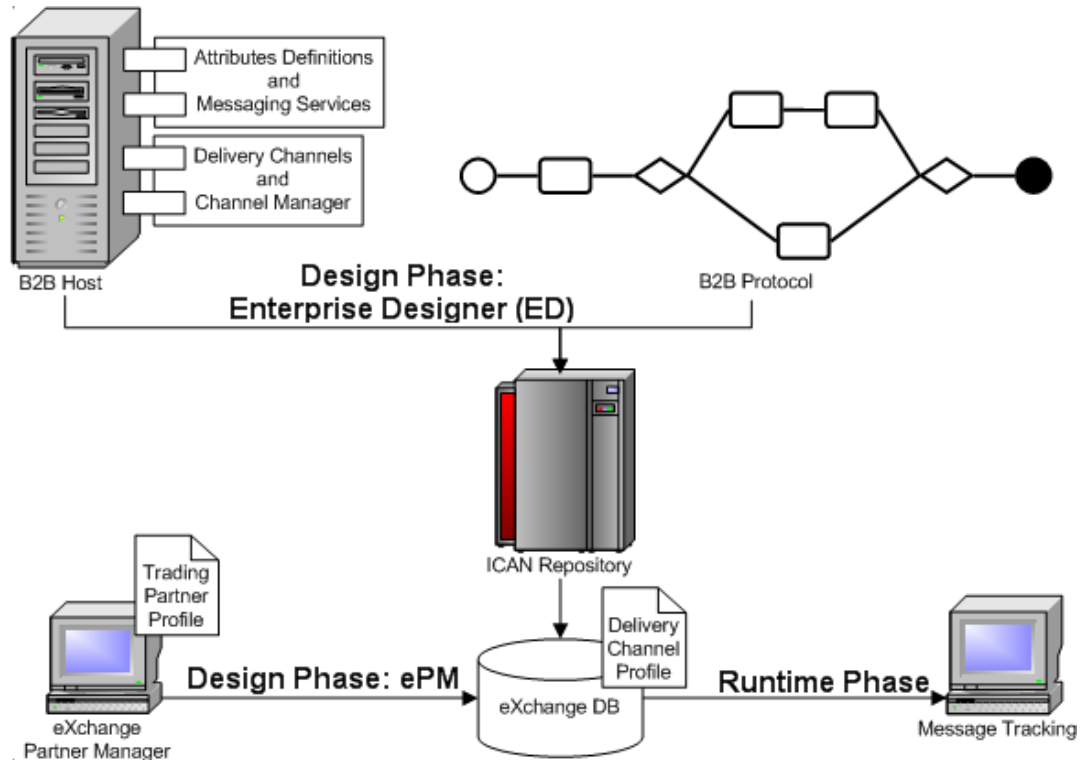
6.4.1 Design Phase: Using Enterprise Designer

Within Enterprise Designer, the B2B Host Designer is used to create B2B Hosts. Each B2B Host is a logical collection of:

- *Messaging services* associated with either standard pre-built B2B protocols (such as AS2 or ebXML) or custom B2B protocols.

- *Attribute definitions* for transport, enveloping (packaging), and messaging.

The B2B Host Designer is used to choose which messaging services are to be used, and to associate each service with one or more sets of transport attribute definitions (such as HTTP, FTP, and many others) to create the metadata for a Delivery Channel Profile—in other words, the *types* of parameters to be supplied for transporting, [de]enveloping, and [receiving]sending messages [from]to trading partners.

After the B2B Host is set up, a Connectivity Map is created to connect its output, and the output of the Message Tracker application, to an Oracle database. Activation causes the DCPs to be stored in the database, and also creates an eXchange Service as an external server in the same Environment that contains the Oracle external. As needed,

the eXchange Service corresponding to the B2B Host is configured with keystores, trust stores, and certificates for authentication and nonrepudiation.

Standard B2B Protocol processes (such as the add-ons for AS2 or ebXML) can be used, and/or the eXchange Protocol Designer can be used to create and configure business logic in custom B2B Protocol processes in the same way the eInsight Business Process Designer is used for Business Processes.

B2B Protocol processes for inbound and/or outbound messages are dragged into a Connectivity Map, where they are represented as services. There, they are connected in the usual fashion with externals (including the eXchange Service for channel management) and with other services. Activation of a corresponding Deployment Profile exposes the map's components for processing by Logical Hosts. As before, it also stores the DCPs into the eXchange database, making them available to eXchange Partner Manager.

6.4.2 Design Phase: Using eXchange Partner Manager

eXchange Partner Manager (ePM) is used to create and configure trading partners and to create trading partner profiles—an association between a particular trading partner and a set of Delivery Channel Profile parameters. For example, if a DCP uses HTTP, then each trading partner profile must be supplied with a value for the URL parameter; or, if a DCP uses FTP, then each trading partner profile must be supplied with values for hostname, target directory, and so forth.

Activating a trading partner stores all of its profiles' DCP configuration settings into the eXchange database.

6.4.3 Runtime Phase

The Logical Host reads the DCP configuration and receives inbound messages from all the channels it references. The DCP parameters for each channel dictate how to handle the inbound message (acknowledgment, decryption, de-enveloping, authentication, …); the business logic of the associated B2B Protocol and Connectivity Map provide further routing and processing; and for an outbound message, the DCP parameters dictate how to handle it (compression, encryption, signature, enveloping, …) and how and where to send it.

eXchange also provides a Message Tracking facility for searching, filtering, and viewing all information written to the eXchange database—errors, acknowledgments, notifications, message attributes, and so forth.

B2B Protocol Manager Composite Applications

This chapter provides an overview of the B2B Protocol Manager Composite Applications.

What's in This Chapter

- **Composite Applications and the ICAN Suite** on page 50
- **About Protocol Manager Composite Applications and eXchange** on page 51
- **Example: X12 Protocol Manager Composite Application** on page 52

7.1   Composite Applications and the ICAN Suite

The ICAN Suite includes several kinds of *composite applications*—applications that are assembled from components provided by several different products, making use of the suite's service-generation, integration, orchestration, and view-generation layers. See Figure 24.

Figure 24 Composite Applications and the ICAN Suite
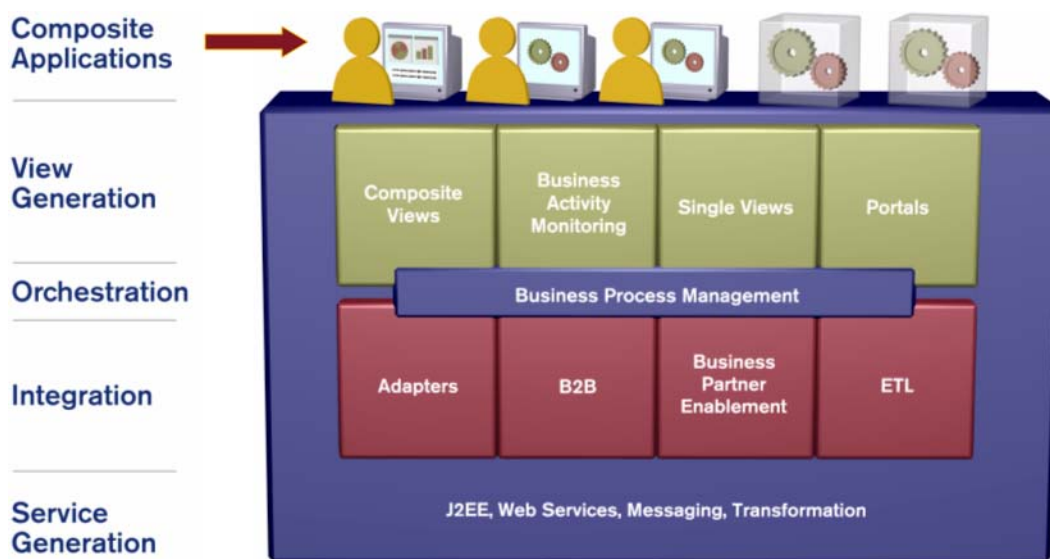
7.2 About Protocol Manager Composite Applications and eXchange

eXchange provides an open framework to support standard B2B business protocols and enveloping protocols. Protocol-specific composite applications are designed to dovetail with the eXchange framework while exposing all components they use (OTDs, BPs, validation rules, and so forth), allowing for further expansion or customization as needed, and achieving end-to-end visibility of all business rules and logic.

In Figure 25, the Project Explorer pane lists the SeeBeyond-supplied BPs for X12, and the canvas shows one of the X12 BPs. All business logic is exposed, and each action is customizable.

Figure 25 SeeBeyond-Supplied BP (for Processing Business Acknowledgments in X12)
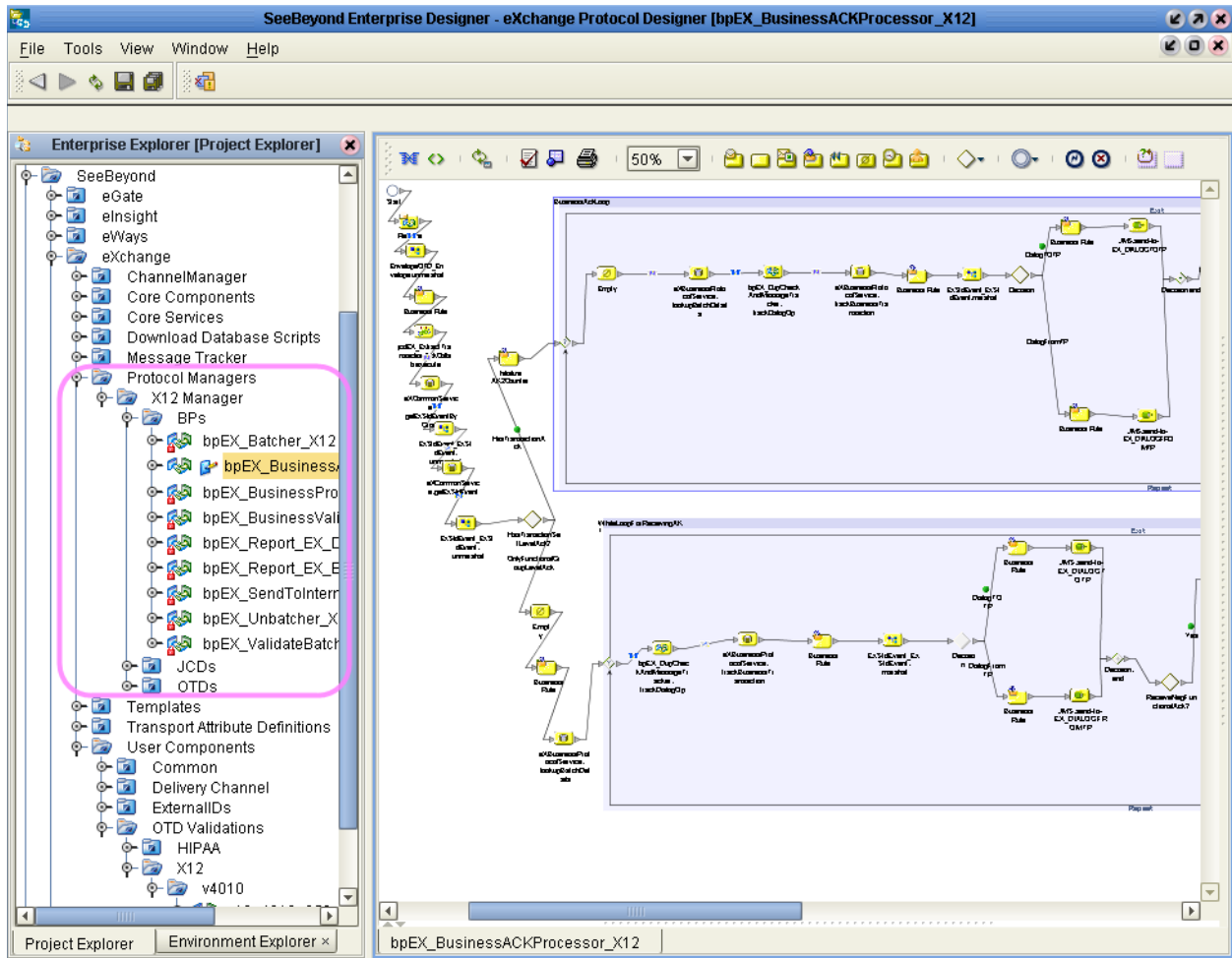
Each protocol manager has the following characteristics:

1 It includes a *messaging service* or *business service*—a choreographed sequence of events incorporating the rules set by the protocol specifications, such as:

- Interchange and acknowledgment processing
- Business message correlation
- Enveloping / de-enveloping
- Batching / splitting
- Event archiving

2 It uses information in the message itself and in the eXchange trading partner profile to prepare the message per the B2B protocol specifications.

3 It works with eXchange common services to prepare, archive, and deliver the message. Common services include:

- Error handling
- Message compression / decompression
- Message tracking
- Trading partner profile database lookup
- PKI security services, such as encryption and signature creation/verification

7.3 Example: X12 Protocol Manager Composite Application

This section provides an overview of how one of our Protocol Managers, the X12 Protocol Manager Composite Application, utilizes eXchange tools to provide a default implementation of X12 message processing.

- In the *View* layer: The eXchange Protocol Designer provides the ability to extend, or customize, the SeeBeyond-supplied X12 protocol implementation, and the eXchange Message Tracker allows searching and viewing of X12 messages.

- In the *Services Orchestration* layer: The eXchange Service Designer is used to choreograph a transaction set; then the X12 protocol implementation prepares and returns the interchange and functional acknowledgments (TA1 and 997) to the trading partner, and also performs message correlation to associate the business response to the request.

- In the *Integration Services* layer: The X12 protocol implementation, in conjunction with the OTD library, validates ISA and GS envelopes from incoming messages, prepares ISA and GS envelopes for outgoing messages, batches together documents to be delivered as a single transaction (ISA), and records the activity in Message Tracking.

eView Studio

This chapter explains the basic operation and features of the SeeBeyond eView Studio (eView) and of the indexing applications created by eView.

What's in This Chapter

- **Summary of Features** on page 53
- **eView and the ICAN Suite** on page 55
- **eView Overview** on page 57
- **Process Overview** on page 58

8.1 Summary of Features

eView provides a flexible framework that allows you to create matching and indexing applications, known as *enterprise-wide master indexes*. A master index uniquely identifies and cross-references the business objects stored in your system databases, creating a single view of all like objects. Business objects can be any type of entity about which you store information, such as customers, members, vendors, businesses, inventory items, and so forth. eView provides your business with a powerful assortment of features that you can use to create master indexes.

8.1.1 eView Studio

The eView tools provide your business with flexibility in designing and creating indexing applications. This flexibility allows you to:

- Rapidly develop a master index for any type of business entity using a wizard to create the framework and using XML documents to configure the attributes of the index.
- Automatically create the primary components of the master index.
- Customize the strategies that determine which field values to populate into the single best record (SBR), which contains the best information about a business object.
- Configure the matching algorithm and logic by specifying the fields to be standardized, the fields to use for matching, and the matching algorithm to use for each field.

- Incorporate a Java API that is customized to the object structure you define. You can call the operations in this API in the Collaboration Definitions or eInsight Business Processes of different Projects.

8.1.2 Master Index

The components of the master index are designed to uniquely identify, match, and maintain information throughout a business enterprise. These components are highly configurable, allowing you to create a custom master index suited to your specific data processing needs. Master indexes created with eView provide the following features.

- **Centralized Information** - The master index maintains a centralized database, enabling the integration of data records throughout the enterprise while allowing local systems to continue operating independently. The index stores copies of local system records and of SBRs, which represent the most accurate and complete data for each object.

- **Configurability** - Before deploying the master index, you define the components and processing capabilities of the system to suit your organization's processing requirements. You can configure the object structure, matching and standardization rules, survivorship rules, queries, EDM appearance, and field validation rules.

- **Cross-referencing** - The master index is a global cross-indexing application that automates record-matching across disparate source systems, simplifying the process of sharing data between systems. The master index uses the local identifiers assigned by your existing systems as a reference for cross-indexing, allowing you to maintain your current systems and practices.

- **Data Cleansing** - The master index uses configurable matching algorithm logic to uniquely identify object records, and to identify duplicate and potential duplicate records. The index provides the functionality to easily merge or resolve duplicates. The index can be configured to automatically merge records that are found to be duplicates of one another.

- **Data Updates** - The master index provides the ability to add, update, deactivate, and delete data in the database tables through messages received from external systems. Records received from external systems are checked for potential duplicates during processing.

- **Identification** - The master index employs configurable probabilistic matching technology, using a matching algorithm to formulate an effective statistical measure of how closely records match. Using a state-of-the-art algorithm in real-time mode and establishing a common method of locating records, the index consistently and precisely identifies objects within an enterprise.

- **Unique Identifier** - Records from various systems are cross-referenced using an enterprise-wide unique identifier (EUID) that the master index assigns to each object record. The index uses the EUID to cross-reference the local IDs assigned to each object by the various computer systems throughout the enterprise.

8.2 eView and the ICAN Suite

eView Studio is a part of the SeeBeyond ICAN Suite of products. eView is highly configurable, allowing you to define the data structure of the information to be indexed and to define the logic that determines how data is updated, standardized, weighted, and matched in the master index database. The master indexes created by eView provide accurate identification of objects throughout your organization, and cross-reference an object's local identifiers using an enterprise-wide unique identification number (EUID) assigned by the master index. The master index also ensures accurate data by identifying potential duplicate records and providing the ability to merge or resolve duplicate records. All information is centralized in one shared index. Maintaining a centralized database for multiple systems enables the indexing application to integrate data throughout the enterprise while allowing local systems to continue operating independently.

8.2.1 ICAN Integration

eView and the indexing applications it creates are tightly integrated with the ICAN Suite and run as components within the ICAN Suite environment. You can create multiple indexing applications in the ICAN Suite environment, each uniquely identifying different types of business objects.

Figure 26 illustrates some of the eView and ICAN components that work together.

Figure 26 eView and the ICAN Suite
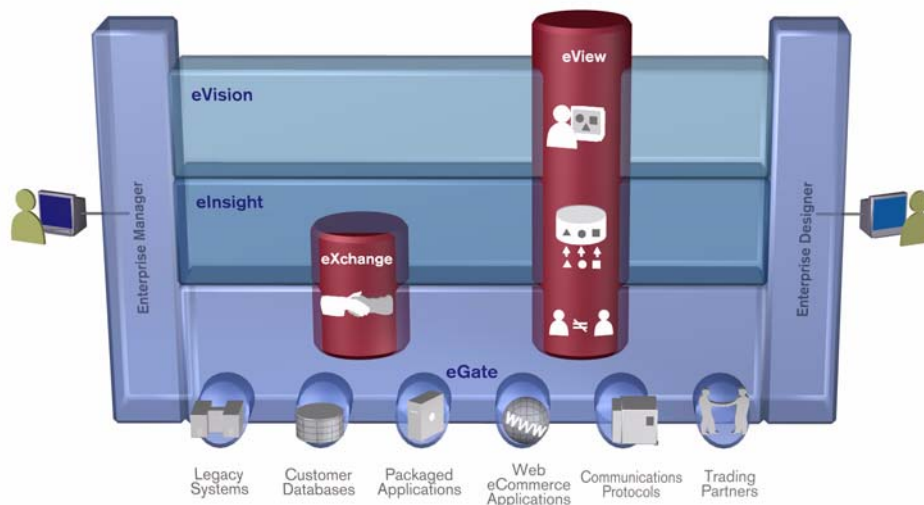
- The eView Wizard, eView editors, and Custom Plug-in module run as components within the **Enterprise Designer**.

- Scripts for creating and customizing the master index databases are stored in, and can be run from, the **Enterprise Designer**.

- eView configuration files, connectivity components, and deployment profiles are stored in the **SeeBeyond Repository**.

- The eView Manager Service, which coordinates the activities of a deployed indexing application, runs within the **SeeBeyond Integration Server.**

- Messages sent to the indexing applications can also be processed through **eInsight** for enhanced business management functions.

- Information in eView databases can be accessed via Web pages created by **eVision**, as well as through the eView database maintenance and monitoring system. **ePortal** unifies the presentation of eView data with information in other ICAN Suite components.

- Web-based process monitoring is provided by the **Enterprise Manager**.

8.2.2. ICAN Suite Services

The ICAN Suite has a wide range of functions that it shares with all of the ICAN Suite products. eView is able to leverage many platform level services, such as:

- **Resource Management** - The ICAN Suite uses a distributed and open architecture that enables components to access system resources (memory and processing power) as needed and in conjunction with other components.

- **Security** - The ICAN Suite provides a security module for eView, to fulfill security needs such as user authentication and authorization of access to eView functions and to the functions of the indexing applications created by eView.

- **Repository storage and access** - The setup, component, and configuration information for the elements of a Project, including database scripts, application files, runtime files, and related eView components, are stored in the Repository.

- **Deployment abilities** - The ICAN Suite provides deployment profiles that contain all the information necessary to activate an eView Project and deploy the indexing applications to the server. When a deployment profile is activated, the eView master indexes are deployed to the server, enabling access to the database.

- **Monitoring** - The Enterprise Manager lends Web-based monitoring abilities to eView, allowing you to observe and correct message transfer and routing activities.

- **Connectivity Mapping** - The Connectivity Map maintains the relationships between the eView master indexes and other system components.

- **Version Control** - This feature allows you to maintain multiple versions of the same components through a check in and check out process.

- **Impact Analysis** - Impact Analysis allows you to view how changes to one component will impact other components of a Project or all Projects in the Repository.

8.3 eView Overview

eView provides the tools that allow you to create and configure an enterprise-wide master index for any type of data. The eView Wizard guides you through the initial setup steps, and special editors are provided so you can further customize the configuration, processing rules, and database structure of the master index. The eView Wizard automatically generates the components you need to implement a master index.

8.3.1. eView Architecture

The components of eView are designed to work within the Enterprise Designer to create and configure the master index, and to define connectivity between external systems and the master index. In Figure 27, you can see how the different components interact.

Figure 27   eView Architecture
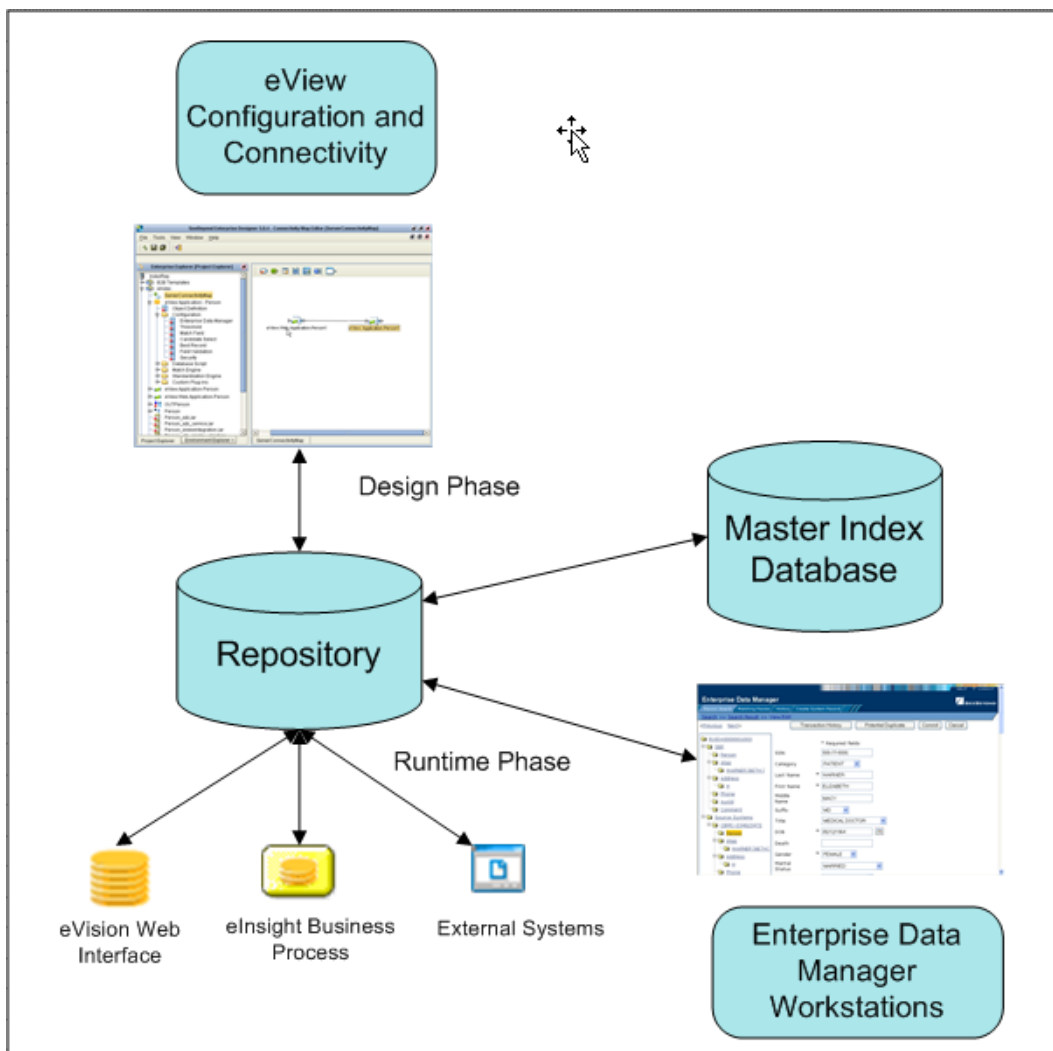
8.4   Process Overview

An eView master index is implemented within a Project in Enterprise Designer. Several eView tools in Enterprise Designer help you create the master index, including the eView Wizard, Project Generator, and eView editors. The Projects associated with eView will also include several standard eGate components, such as OTDs, Collaborations, queues, topics, a deployment profile, and so on. You can use the standard Enterprise Designer editors, such as the OTD or Collaboration editors, to create these components.

8.4.1   Analysis and Design Phase Overview

The process of creating a master index begins with a thorough analysis of the data you plan to store in the index database and to share among the systems connected to the master index. The results of this analysis define the structure of the information stored in the master index database and provide information to help you customize the processing and matching logic for the master index.

From this analysis you can design the object structure, matching and standardization logic, any required custom processing, and the connectivity components for the indexing system. The analysis results can also be used to design the physical components of the system, such as the logical hosts, networking components, and master index database.

8.4.2   Development Phase Overview

The development phase consists of standard tasks for creating an indexing application and advanced tasks for further customizing the applications you create.

Standard Development Tasks

The following steps outline the basic procedure for developing a master index using eView.

1   Create an eGate Project, and create a new eView application within that Project.
2   Using the eView Wizard, define the data and message structure, the operating environment, and external systems sharing data with the master index.
3   Customize the configuration files and generate the application (this creates custom components, such as OTDs and database scripts).
4   Customize the database scripts, and then create the database and load any startup data.
5   Create and define connectivity components, and then create a Connectivity Map to configure the relationship between these components.
6   Define the Environment and configure the Integration Server, database connections, and security.
7   Create the deployment profile and activate it to complete the deployment process. The indexing application is now deployed to the Integration Server.

Advanced Development Tasks

You can perform additional tasks during the development phase to customize your indexing application further.

- **Custom Plug-ins** - Create Java classes to perform custom processing once the matching process is complete, such as performing additional operations before finalizing a transaction or validating certain field values.

- **Database Distribution** - Before running the predefined scripts against the database, create additional tablespaces to distribute the tables of the master index.

- **Match Engine Configuration** - Customize how weighting is performed by modifying the match engine configuration files. For the SeeBeyond Match Engine, these files are included in the eView Project.

- **eInsight Integration** - When you generate the eView Project, a set of operations are created that are specifically tailored to the object structure you defined. Use these operations to integrate the indexing application with an eInsight Business Process.

eView Wizard

The eView Wizard takes you through each step of the master index setup process and, based on the information you specify, creates the XML files that define the configuration of the application.

Figure 28   eView Wizard
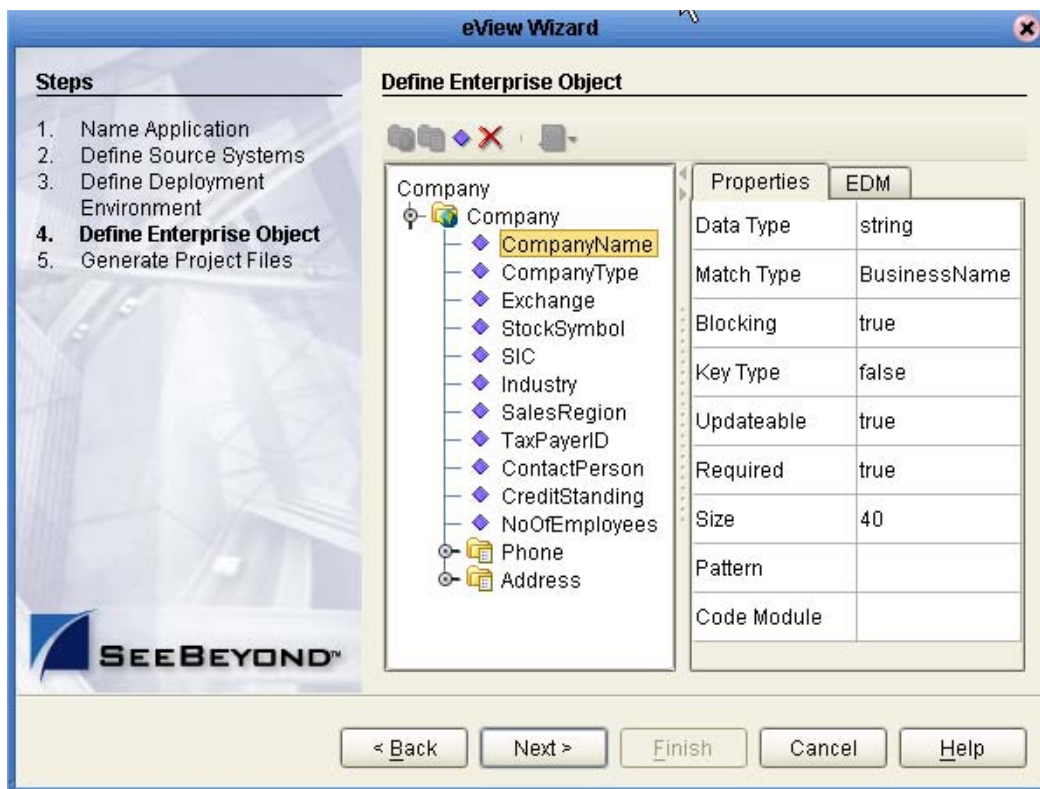
The eView Wizard guides you through the following steps to set up the master index configuration.

1 **Name Application** - Create a unique name for the master index.

2 **Define Source Systems** - List each external system to be connected to the master index.

3 **Define Deployment Environment** - Specify the database platform, match engine type, and date format.

4 **Define Enterprise Object** - Define the data structure of the master index.

5 **Generate Project Files** - Create the framework of the master index.

8.4.3 Runtime Phase Overview

Once all of the analysis, design, and development tasks are complete and the system is running, you can:

- Transform and route data between external systems and the master index application (where the matching process occurs).

- Monitor and manage activities and the overall Project with the Enterprise Manager.

- Monitor and maintain the indexed records in the master index database using the Enterprise Data Manager (a custom Web-based application).

eView Manager Service

The eView Manager Service provides a session bean to all components of the master index, such as the Enterprise Data Manager, Query Builder, Update Manager, and so on. During the runtime phase, the eView Manager Service:

- Provides connectivity to the master index database.

- Specifies the query to use for the match process, along with system parameters controlling the match process.

- Coordinates the activities of the various components of the master index, including queries, updates, object persistence, system parameters, and so forth.

Master Index Database

The components of a master index connect to the database to provide:

- **Persistence** - The Object Persistence service writes instance data to database tables to ensure that data is able to persist in the system.

- **Recoverability** - The eView database allows you to recover data from the last state of consistency.

- **Transaction History** - The database stores a description of the changes that occur for each transaction. This allows you to view a complete history of changes to each record in the database.

Database Monitoring and Maintenance

The Enterprise Data Manager (EDM) is a Web-based interface that allows you to monitor and maintain the data in your master index database. The appearance and search capabilities of the EDM are highly configurable to allow you to view and search for information in the way that best suits your business needs. A sample View/Edit page from the EDM is displayed in Figure 29.

Figure 29   Enterprise Data Manager
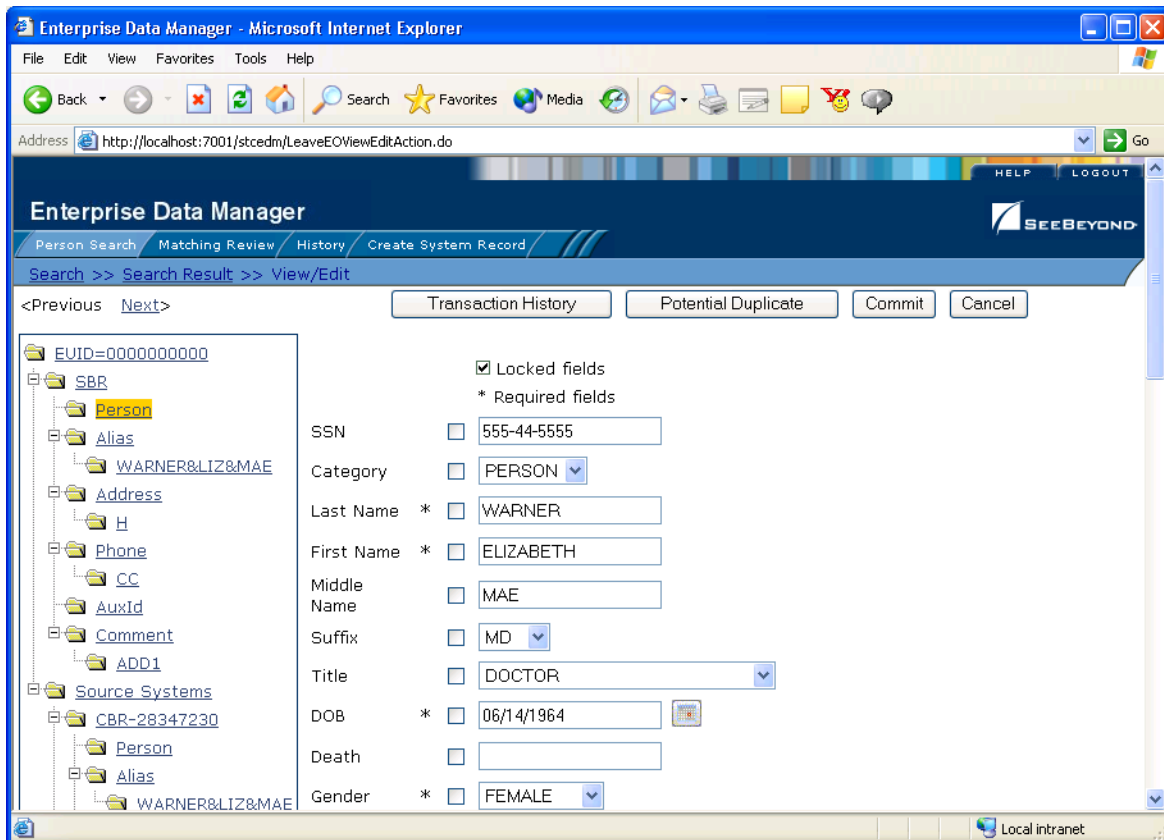
The Enterprise Data Manager allows you to perform these primary functions to monitor and maintain the data in a master index database.

- **Transaction History** - You can view a complete history of each object for both the local system records and the single best record.

- **Data Maintenance** - You can add new records; view, update, deactivate, or reactivate existing records; and compare records for similarities and differences.

- **Search** - You can perform searches against the database for a specific object or a set of objects. For certain searches, the results are assigned a matching weight indicating the probability of a match.

- **Potential Duplicate Detection and Handling** - Using matching algorithm logic, the master index identifies potential duplicate records, and provides the functionality to correct the duplication.

- **Merge and Unmerge** -You can merge records you find to be actual duplicates of one another at either the enterprise-wide unique identifier (EUID) level or the system record level. Merges made in error can easily be unmerged.

eIndex Global Identifier Composite Application

This chapter explains the basic operation and features of the SeeBeyond eIndex™ Global Identifier Composite Application (eIndex).

What's in This Chapter

9.1 Summary of Features

The components of eIndex are designed to uniquely identify, match, and maintain information throughout a healthcare enterprise. These components are highly configurable, allowing you to create a custom master person index suited to your specific data processing needs. eIndex provides your business with a powerful assortment of features, including:

- **Integration Server Technology** - eIndex can be deployed on the SeeBeyond Integration Server (SIS).

- **Centralized Information** - eIndex maintains a centralized database, enabling the integration of data records throughout the enterprise while allowing local systems to continue operating independently. eIndex stores copies of local system records and of single best records (SBRs), which represent the most accurate and complete data for each person.

- **Cross-referencing** - eIndex is a global cross-indexing application that automates record-matching across disparate source systems, simplifying the process of sharing data between systems. eIndex uses the local identifiers assigned by your existing systems as a reference, allowing you to maintain your current systems and practices.

- **Data Cleansing** - eIndex uses configurable matching algorithm logic to uniquely identify object records, and to identify duplicate and potential duplicate records. eIndex provides the functionality to easily merge or resolve duplicates, and can be

configured to automatically merge records that are found to be matches of one another.

- **Data Updates** - eIndex provides the ability to add, update, deactivate, and delete data in the database tables through messages received from external systems. Records received from external systems are checked for potential duplicates during processing.

- **Flexibility** - eIndex is provided in a highly flexible framework, allowing extensive customizations to the application. Before deploying eIndex you can customize the components and processing capabilities of the system to suit your organization's processing requirements. You can configure the object structure, matching and standardization rules, survivorship rules, queries, EDM appearance, and field validation rules.

- **Identification** - eIndex employs configurable probabilistic matching technology, using a matching algorithm to formulate an effective statistical measure of how closely records match. Using a state-of-the-art algorithm in real-time mode and establishing a common method of locating member records, eIndex consistently and precisely identifies members within an enterprise.

- **Matching Algorithm** - The matching algorithm and logic used by eIndex is highly configurable. eIndex uses the SeeBeyond Match Engine for standardization and matching, but it can also be implemented with other match engines that provide a compatible API. The matching and standardization logic of the match engine is customizable.

- **Unique Identifier** - Members from various systems are cross-referenced using an enterprise-wide unique identifier (EUID) that eIndex assigns to each member record. eIndex uses the EUID to cross-reference the local IDs assigned to each member by the various computer systems throughout the enterprise.

9.2 eIndex and the ICAN Suite

eIndex Global Identifier Composite Application is a part of the SeeBeyond ICAN Suite of products. eIndex is a healthcare-oriented, *enterprise-wide master person index* that creates a single view of person information by maintaining the most current information about the people who participate throughout your organization and by linking information from different locations and computer systems. eIndex provides accurate identification of members throughout your healthcare enterprise, and cross-references a member's local identifiers using an enterprise-wide unique identification number (EUID) assigned by eIndex. eIndex also ensures accurate member data by identifying potential duplicate records and providing the ability to merge or resolve duplicate records. All member information is centralized in one shared index. Maintaining a centralized database for multiple systems enables eIndex to integrate data throughout the enterprise while allowing local systems to continue operating independently.

Figure 30 eIndex and the eGate Integrator
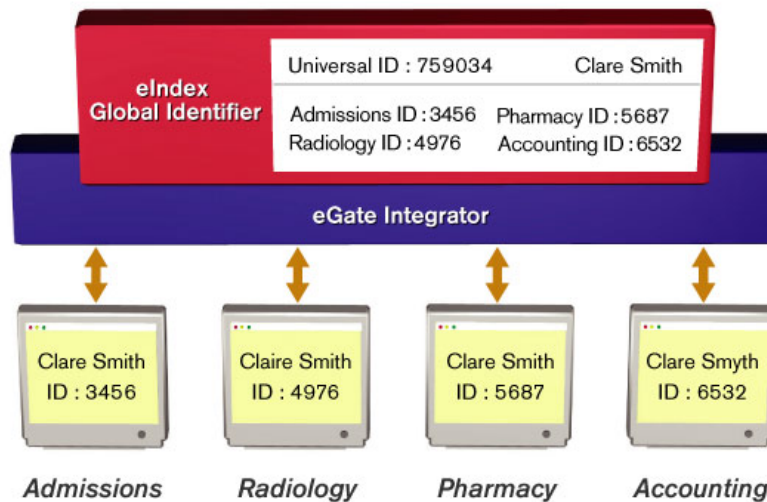
9.2.1 ICAN Integration

eIndex is tightly integrated with the ICAN Suite and exists within a Project of the ICAN Suite environment. The configuration tools of eIndex (the editors and Custom Plug-in module) run within the **Enterprise Designer**. In addition, eIndex is integrated with the ICAN suite in the following ways.

- Scripts for creating and customizing the eIndex database are stored in, and can be run from, the **Enterprise Designer**.

- eIndex configuration files, connectivity components, and deployment profiles are stored in the **SeeBeyond Repository**.

- The eIndex Manager Service, which coordinates the activities of a deployed eIndex application, runs within the **SeeBeyond Integration Server.**

- Messages sent to eIndex can also be processed through **eInsight** for enhanced business management functions.

- Information in the eIndex database can be accessed via Web pages created by **eVision**, as well as through the eIndex database maintenance and monitoring system. **ePortal** unifies the presentation of eIndex data with information in other ICAN Suite components.

- Web-based process monitoring is provided by the **Enterprise Manager**.

9.2.2. ICAN Suite Services

The ICAN Suite has a wide range of functions that it shares with all of the ICAN Suite products. eIndex is able to leverage many platform level services, such as:

- **Resource Management** - The ICAN Suite uses a distributed and open architecture that enables components to access system resources (memory and processing power) as needed and in conjunction with other components.

- **Security** - The ICAN Suite provides a security module for eIndex, to fulfill security needs such as user authentication and authorization of eIndex functions.

- **Repository storage and access** - The setup, component, and configuration information for the elements of a Project, including database scripts, application files, runtime files, and related eIndex components, are stored in the Repository.

- **Deployment abilities** - The ICAN Suite provides deployment profiles that contain all the information necessary to activate an eIndex Project and deploy eIndex to the server (enables access to the database).

- **Monitoring** - The Enterprise Manager lends Web-based monitoring abilities to eIndex, allowing you to observe and correct message transfer and routing activities.

- **Connectivity Mapping** - The Connectivity Map maintains the relationships between eIndex and other system components.

- **Version Control** - Version Control allows you to maintain multiple versions of the same components through a check in and check out process.

- **Impact Analysis** - Impact Analysis allows you to view how changes to one component will impact other components of a Project or all Projects in the Repository.

9.3   eIndex Architecture

The components of eIndex are designed to work within the Enterprise Designer to configure the eIndex application, and to define connectivity between external systems and eIndex. eIndex consists of several components, each acting independently of the others, but working together to provide accurate data maintenance and identification. These components interact with other SeeBeyond ICAN products, such as eGate Integrator and eWays, and through them with the external data processing products and systems used by an organization.

The SeeBeyond ICAN suite provides the flexible architecture required to efficiently enable the seamless sharing of customer information. As data is transferred from one local system to another using eGate, eIndex consults the database and retrieves the appropriate local identifier. This is a transparent process—each computer system within this network can continue using its own local identifiers.

eIndex consists of the following components:

- The Enterprise Data Manager

- An eIndex database

- An eIndex Project

- An eIndex Environment

- Custom Java API methods

In addition, the eIndex system includes one or more client Projects that can use the eIndex custom Java methods in their Collaborations and Business Processes. In Figure 31, you can see how the different eIndex components interact.

Figure 31   eIndex Architecture
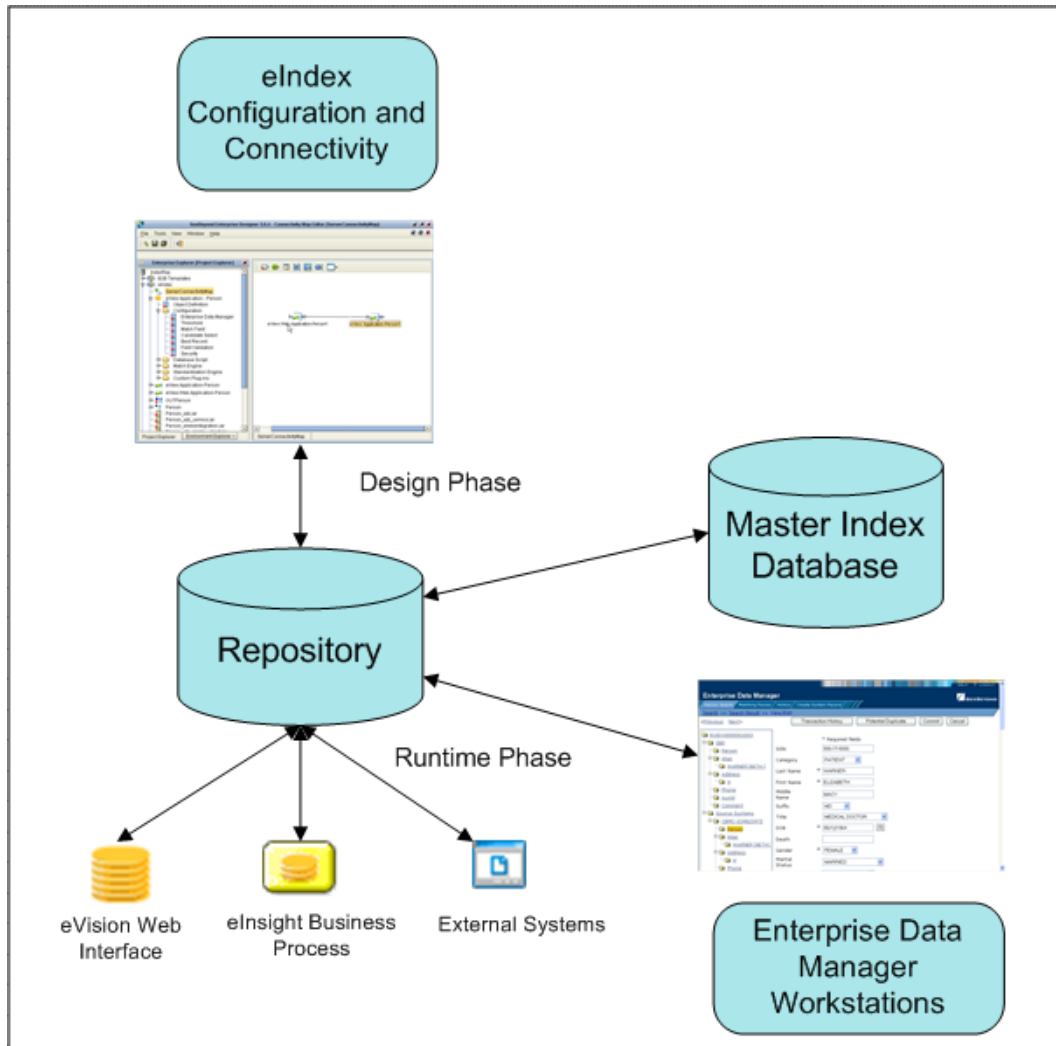
9.4   Process Overview

eIndex is implemented within a Project in Enterprise Designer. Several tools in Enterprise Designer help you configure eIndex, including the Project Generator and eIndex editors. The Projects associated with eIndex will also include several standard eGate components, such as OTDs, Collaborations, queues, topics, a deployment profile, and so on. You can use the standard Enterprise Designer editors, such as the OTD or Collaboration editors, to create these components.

9.4.1 Analysis and Design Phase Overview

The process of configuring eIndex begins with a thorough analysis of the data you plan to store in the eIndex database and to share among the systems connected to eIndex. The results of this analysis define the structure of the information stored in the eIndex database and provide information to help you customize the processing and matching logic.

From this analysis you can customize the object structure, matching and standardization logic, any required custom processing, and the connectivity components. The analysis results can also be used to design the physical components of the system, such as the logical hosts, networking components, and database.

9.4.2 Development Phase Overview

The development phase consists of standard tasks for configuring eIndex and advanced tasks for further customizing the application.

Standard Development Tasks

The following steps outline the basic procedure for configuring eIndex.

1 Customize the configuration files and regenerate the application (this updates custom components, such as OTDs and database scripts).

2 Customize the database scripts, and then create the database and load any startup data.

3 If necessary, create and define connectivity components, and then create a Connectivity Map to configure the relationship between these components.

4 Define the Environment and configure the Integration Server, database connections, and security.

5 Create the deployment profile and activate it to complete the deployment process. The indexing application is now deployed to the Integration Server.

Advanced Development Tasks

You can perform additional tasks during the development phase to customize your indexing application further.

- **Custom Plug-ins** - Create Java classes to perform custom processing once the matching process is complete, such as performing additional operations before finalizing a transaction or validating certain field values.

- **Database Distribution** - Before running the predefined scripts against the database, create additional tablespaces to distribute the tables of the master index.

- **Match Engine Configuration** - Customize how weighting is performed by modifying the match engine configuration files. For the SeeBeyond Match Engine, these files are included in the eIndex Project.

- **eInsight Integration** - When you regenerate the eIndex Project, a set of operations are updated to be specifically tailored to the object structure you defined. Use these operations to integrate the indexing application with an eInsight Business Process.

9.4.3 Runtime Phase Overview

Once all of the analysis, design, and development tasks are complete and the system is running, you can:

- Transform and route data between external systems and eIndex (where the matching process occurs).

- Monitor and manage activities and the overall Project with the Enterprise Manager.

- Monitor and maintain the indexed records in the eIndex database using the Enterprise Data Manager (a Web-based application).

eIndex Manager Service

The eIndex Manager Service provides a session bean to all components of eIndex, such as the Enterprise Data Manager, Query Builder, Update Manager, and so on. During the runtime phase, the eIndex Manager Service:

- Provides connectivity to the eIndex database.

- Specifies the query to use for the match process, along with system parameters controlling the match process.

- Coordinates the activities of the various components of eIndex, including queries, updates, object persistence, system parameters, and so forth.

eIndex Database

The components of eIndex connect to the database to provide:

- **Persistence** - The Object Persistence service writes instance data to database tables to ensure that data is able to persist in the system.

- **Recoverability** - The eIndex database allows you to recover data from the last state of consistency.

- **Transaction History** - The database stores a description of the changes that occur for each transaction. This allows you to view a complete history of changes to each record in the database.

Database Monitoring and Maintenance

The Enterprise Data Manager (EDM) is a Web-based interface that allows you to monitor and maintain the data in your eIndex database. The appearance and search capabilities of the EDM are highly configurable to allow you to view and search for information in the way that best suits your business needs. A sample View/Edit page from the EDM is displayed in Figure 32.

Figure 32   Enterprise Data Manager
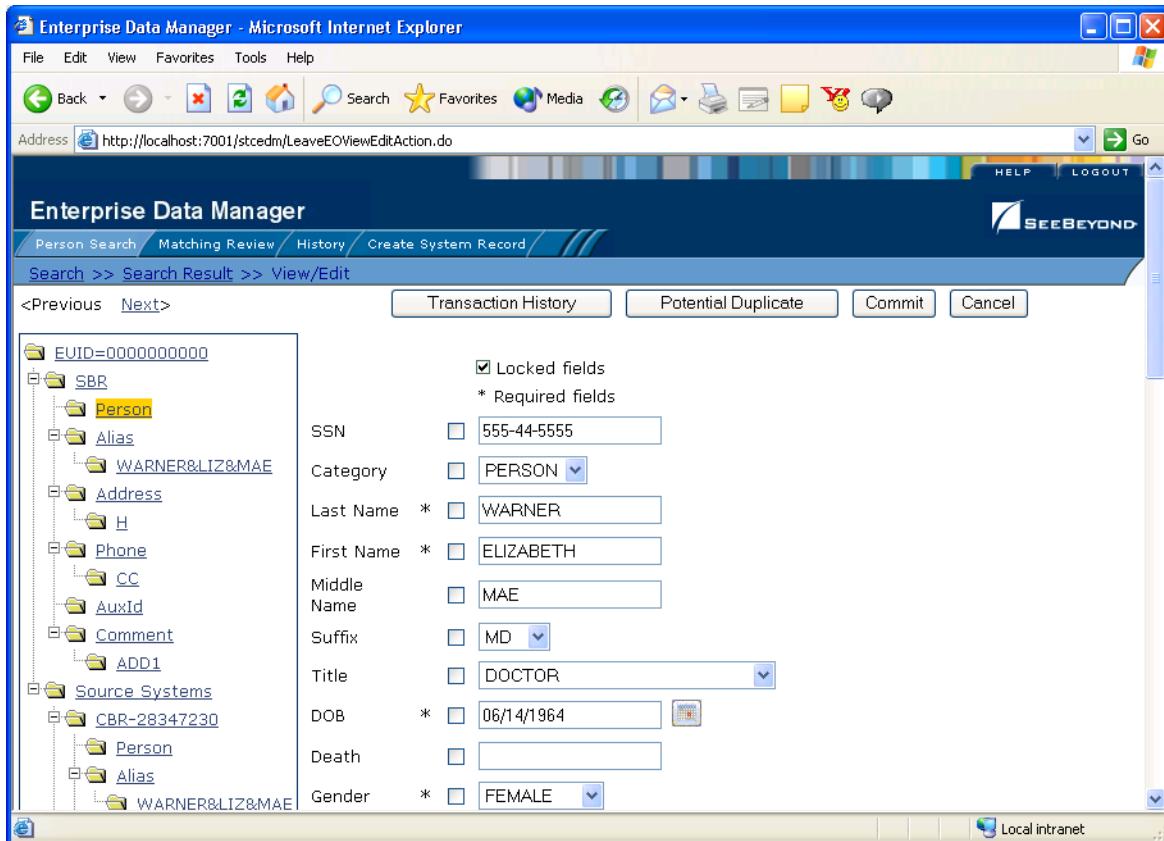
The Enterprise Data Manager allows you to perform these primary functions to monitor and maintain the data in a master index database.

- **Transaction History** - You can view a complete history of each person in the database for both the local system records and the single best record.

- **Data Maintenance** - You can add new records; view, update, deactivate, or reactivate existing records; and compare records for similarities and differences.

- **Search** - You can perform searches against the database for a specific person or a set of person records. For certain searches, the results are assigned a matching weight indicating the probability of a match.

- **Potential Duplicate Detection and Handling** - Using matching algorithm logic, eIndex identifies potential duplicate records and provides the functionality to correct the duplication.

- **Merge and Unmerge** - You can merge records you find to be actual duplicates of one another at either the EUID or system record level. Merges made in error can easily be unmerged.

eTL Integrator

This chapter explains the basic operation and features of SeeBeyond's eTL Integrator.

What's in This Chapter

- **Overview** on page 71
- **Introduction** on page 71
- **eTL Features** on page 73

10.1 Overview

SeeBeyond's eTL Integrator technology is optimized for very large record sets and build data scenarios that are fully integrated with the SeeBeyond ICAN suite (Integrated Composite Application Network Suite) to unify the domains of eAI (eBusiness and Application Integration) and ETL (Extract, Transform, and Load). An eTL Collaboration can be integrated into the enterprise business process or used as a standalone ETL process.

10.2 Introduction

Extraction Transform and Load (ETL) is a data integration technology that extracts data from several heterogeneous data sources, transforms the data, then loads the data in a uniform format into a target data source.

Figure 33 eTL and the ICAN Product Suite
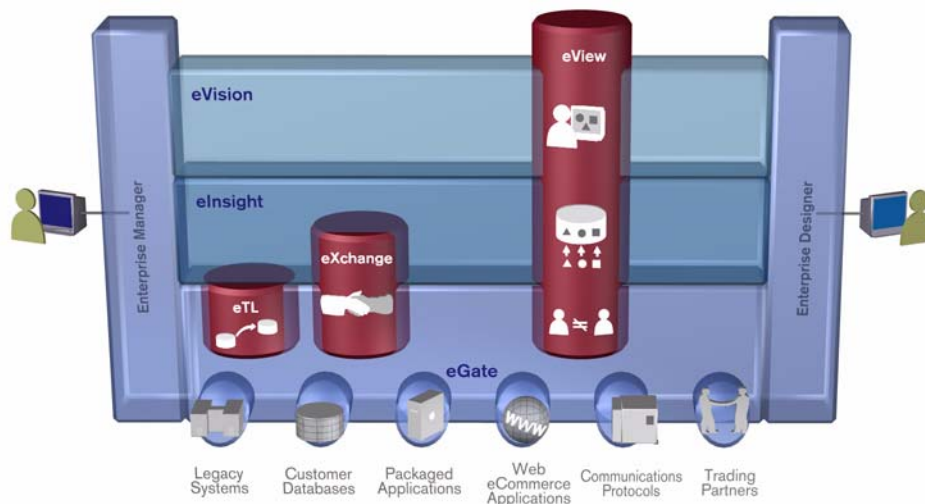

10.2.1. The eTL Integrator Product Description

SeeBeyond's eTL Integrator technology is optimized for very large record sets and build data scenarios that are fully integrated with the SeeBeyond ICAN suite (Integrated Composite Application Network Suite) to unify the domains of eAI (eBusiness and Application Integration), and Enterprise Information Integration (EII). With these unified domains you can build unprecedented solutions using both message based processing (eGate) and dataset based processing (eTL) technologies.

The eTL Integrator product provides excellent performance at runtime for high volume extraction and load of tabular data sets. The eTL Integrator can be integrated into the enterprise business processes or used as a standalone product.

The eTL Integrator product can be used to acquire a temporary subset of data for reports or other purposes, or acquire a more permanent data set for the population of a data mart or data warehouse. The product may also be used for conversion of one database type to another or for the migration of data from one database or platform to another.
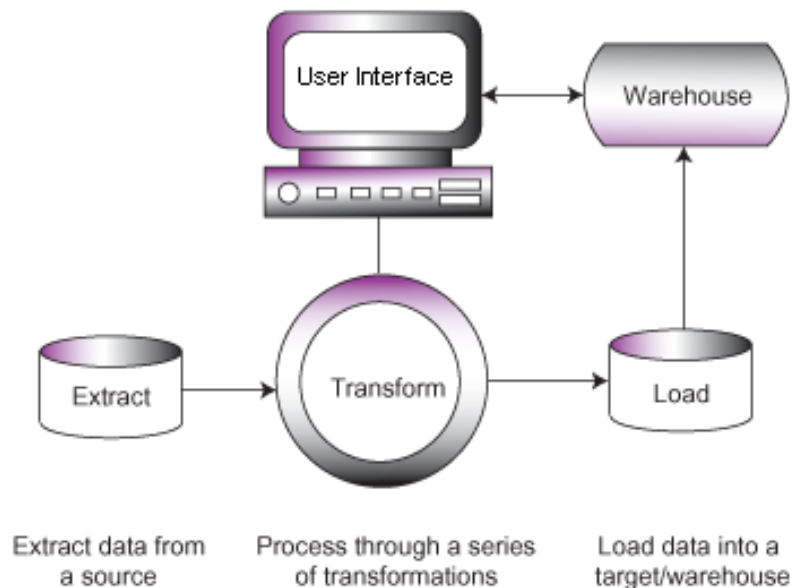
10.2.2. The ETL Process

In an ETL process, data is extracted from data sources. The data is then transformed (or processed), using rules, operators, or filters, into a desired state suitable for loading into a database or data warehouse. See the following figure.

In managing databases, extract, transform, load (ETL) refers to three separate functions combined into a single programming tool.

1 First, the extract function reads data from a specified source database and extracts a desired subset of data.

2 Next, the transform function works with the acquired data – using rules or lookup tables, or creating combinations with other data – to convert it to the desired state.

3 Finally, the load function is used to write the resulting data to a target database, which may or may not have previously existed.

Figure 34   The ETL Process

Extract data from a source   Process through a series of transformations   Load data into a target/warehouse

10.3 eTL Features

Supporting Platforms

eTL Integrator is compatible with the following systems and platforms:

- Oracle 8.1.7 and 9i (9.0.2), SQL Server 2000, DB2 UDB 8.1, Sybase 12.5, and delimited and fixed width flat files
- Multiple sources and multiple destinations
- Standard eGate platform support

Transformation Capability

eTL Integrator enables seamless filtering and data transformation.

- Merge/upsert (updates or inserts as appropriate)

- Drag and drop GUI design features (create joins across disparate data sources)

- Validate Collaborations before performing the ETL processes

- User friendly, state-of-the-art, design tools reduce development time and cost

- GUI based Collaboration editor employs drag and drop design features

  - User friendly Wizards (easy OTD creation)

  - Graphical operators (dragged from a toolbar)

  - Graphical tools (create underlying SQL)

- Tight integration among ICAN Suite business data systems

  - Web Services interface

  - Seamless integration with the ICAN Suite

  - ICAN Suite provides versioning and history

Operators and Transformation Tools and Functionality

Development is simplified with GUI based development tools that are appropriate for SQL Collaborations. Graphical drag and drop modeling tools enable SQL operations in various categories:

- Number

- Date

- Comparison

- Boolean

- SQL operators

- Mathematical operators

- String manipulations

- Conditional data extractions and transformations

- Join

  - Auto-detect primary key relationships between tables, as indicated in OTDs

  - Between tables from disparate data sources that have no relationship

  - Supports inner, left, right, and full outer joins

- Lookups

  - Extensive lists of operators allows for the creation of lookups as part of the eTL process, using joins across tables

- Merge

  - Automatic update if row exists

  - Automatic insert if row doesn't exist

- Test data and test runs

- Runtime variables (configured by the user)

Architecture

Fully integrated with the ICAN Suite to make eTL a more versatile and powerful tool.

- A deployed eTL engine runs as a JCA compliant (J2EE) resource adapter inside the SeeBeyond Integration server.

- The business rules defined by the eTL Collaboration definition are stored in the SeeBeyond Repository.

- At deployment time, the business rules are used to generate the appropriate platform specific SQL.

- eTL Integrator leverages OTDs defined in the Enterprise Designer so you don't have to create OTDs specifically for an eTL Collaboration.

ePortal Composer

This chapter describes the ePortal Composer (ePortal).

What's in This Chapter

11.1 About ePortal

ePortal, SeeBeyond's Web *portal* application, provides a set of Web-based tools that enable the Web developer and/or system Administrator to create Web portals.

A portal is a Web site that serves as a gateway for Web-based services and applications. Portals unify business information and personalize the portal experience based on a user's profile.

11.1.1 ePortal and ICAN Suite

ePortal leverages the SeeBeyond ICAN Suite by enabling enterprise-wide access to business processes from a single point of entry, that is, a portal.

A portal allows the organization to present a single, unified view of enterprise data and applications to employees, customers, and partners. A system Administrator can assign multiple URLs to ePortal *channels* (also called *portlets*). These channels are aggregated within the portal and collectively presented to end-users.

Through any of its Web channels, ePortal can enable you to simultaneously view multiple ICAN eVision Studio applications, other ICAN-generated user interfaces, and specified Web-enabled enterprise content.
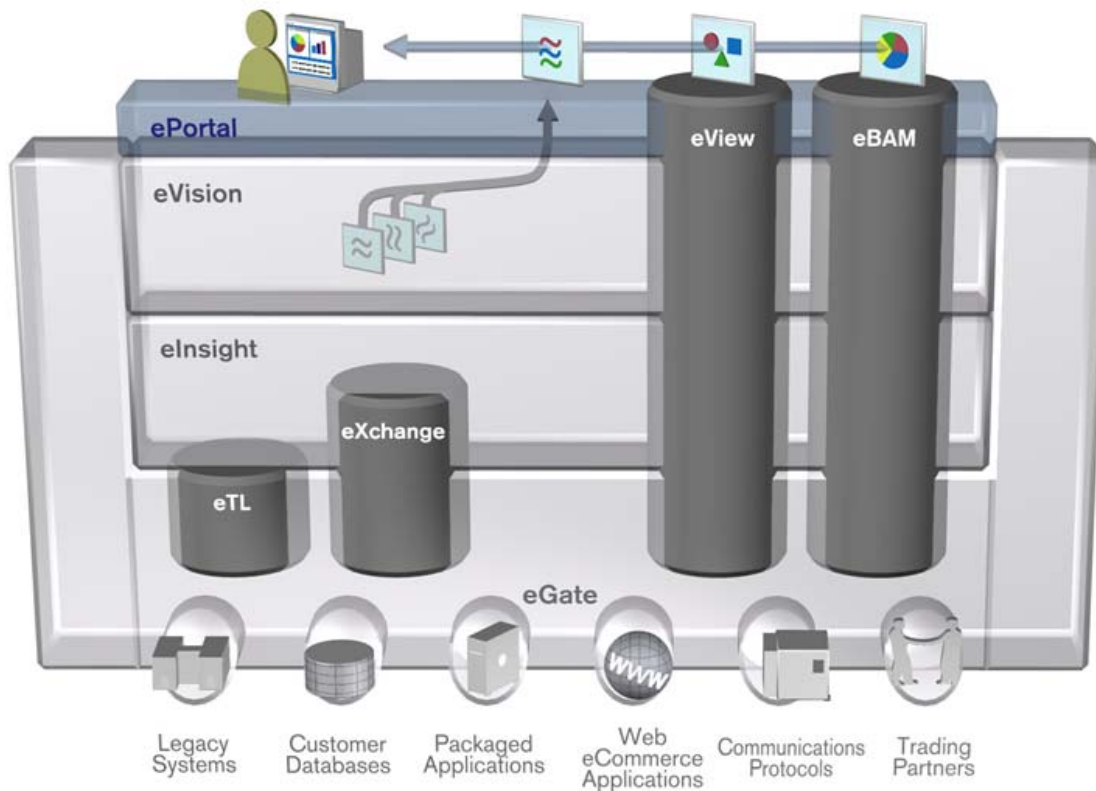
11.1.2 ePortal Operation

Using ePortal in the ICAN Suite, you can:

- Monitor business processes in real time with the Enterprise Manager user interface

- Run eVision Web applications

You are able to do both of these operations within their own individual channels. How ePortal can be integrated within the ICAN Suite is shown in Figure 35.

Figure 35 ePortal Within the ICAN Suite

ePortal operates seamlessly with any of the ICAN Suite applications shown in Figure 35, for example, eVision, but ePortal does not require them. You can use ePortal as a stand-alone application with or without any other ICAN Suite application.

11.2 About Channels

A channel allows you to view and interact with a variety of different kinds of Web content.

11.2.1 Supported Channel Types

ePortal supports a number of different channel types, such as, a specific URL (for example, an eVision page or a custom developed JSP page), Java applets, Web Services Remote Portlets (WSRP), that is, consumer portlets, and Rich Site Summary (RSS) documents. ePortal allows a system Administrator to create, manage, personalize, and publish channels.

11.2.2 Categories and Fragments

Channels are assigned to channel *categories* and/or *fragments*, which are defined as follows:

- **Category**: A collection of channels that logically belong together. For example, channels that have a related purpose or dependency can be assigned to the same category.

- **Fragment**: Displays templates with a combination of predefined channels and layout.

11.3 Using ePortal

ePortal has the following types of users:

- **System Administrator**
- **End-user**

Different kinds of operations are available to either or both types of users. A general summary of the available operations for each user follows:

- System Administrators can create/publish visually appealing Web connections to applications and channels, allowing users to perform real-time interaction with ICAN Suite's run-time assets, or with any other URL or Web application. The system Administrator is also responsible for setting up authentication, categories, and groups.

- End-users can create personalized channels with targeted views of business information. For example, a channel can allow an end-user to log onto a system, review and complete assigned tasks, and monitor the progress of business activities. Through channels, end-users can interact with running business processes, while working only with the information that they need to see.

11.4 Web-based Portal Administration and Management

Under categories, system Administrators can orchestrate channel access using the users and roles defined in the Enterprise Designer or by using Lightweight Directory Access Protocol (LDAP).

ePortal provides Web-based wizards that guide the system Administrator through the process of channel configuration and management. For example, the system Administrator can:

- Create channels to be made accessible in a portal; a channel can access content from any available URL source.

- Assign any channel to a category.

- Create WSRP channels.

- Assign specialized fragments to any group.

ePortal supports any combination of URLs and simplifies the end-user's access to application resources, Web content, workflow, e-mail, content management systems, and packaged applications such as ERP and CRM.

11.5   Authentication in ePortal

ePortal offers a variety of choices to the system Administrator for setting up and maintaining system access.

11.5.1 ACL and LDAP Authentication

ePortal can authenticate users by way of the Enterprise Designer's Access Control List (ACL) authentication features or by using LDAP. Other applications in the same realm may not require a user to log on again.

The Enterprise Designer enables authentication of user IDs and passwords when users log onto ePortal. See the *eGate Integrator User's Guide* for information on how to use the Enterprise Designer.

For information on how to set up authentication using LDAP, see the appropriate LDAP documentation.

11.5.2 Single Sign-on (SSO) Authentication

ePortal supports single sign-on (SSO) authentication for channels hosted by the same ICAN run-time environment. When an end-user logs onto the Integration Server or WebSphere, the channels that also run on that application server are also visible to the end-user.

For eVision for example, end-users are able to see eVision channels, but only if those end-users have been assigned to a role that is allowed to view the channel.

eBAM Studio

This chapter describes the eBAM Studio (eBAM).

What's in This Chapter

- **eBAM Studio** on page 80
- **Using eBAM Studio** on page 82

12.1 eBAM Studio

Business Activity Monitoring (BAM) involves the collection, aggregation, and presentation of business activity data according to specified Key Performance Indicators (KPIs). eBAM Studio provides the tools for generating custom, cross-application, digital dashboards for defining and monitoring KPIs, which summarize the aggregated business data collected through the eInsight or eGate application layers. The eBAM Web interface allows the business analyst to transform data that has been collected over time into meaningful, rich visual presentations.

KPIs provide a context for business processes by turning raw data into useful information, allowing the business analyst to focus on monitoring and analyzing measurable enterprise-wide operations and processes. eBAM Studio provides the business analyst with different views of performance data, enabling the identification of time-critical business trends.

eBAM Studio renders real-time and historical data in familiar visual formats, such as pie and bar charts, for display in digital dashboards. These recognizable, easy-to-read contexts enable the business analyst to quickly translate information into action. Leveraging the ICAN Suite and the Web, eBAM provides visibility to key business information, recasts it in a graphical format, and makes it available across the enterprise.

eBAM Studio leverages ICAN platform services; for example:

- KPIs and other run-time conditions are stored in the Repository.
- Secure, single sign-on functionality at design-time, and run-time resource management are provided by the Integration Server.
- Enterprise Designer is used to design the data collection, graphical display, and notification processes.
- Monitor provides the framework for displaying the charts that eBAM constructs.

▪ The data filtered by eBAM Studio can be queried by eInsight or eVision and then mapped to other components.

How eBAM Studio is integrated with the ICAN Suite is shown in **Figure 36 on page 81**.

Figure 36   eBAM Studio Integration with the ICAN Suite
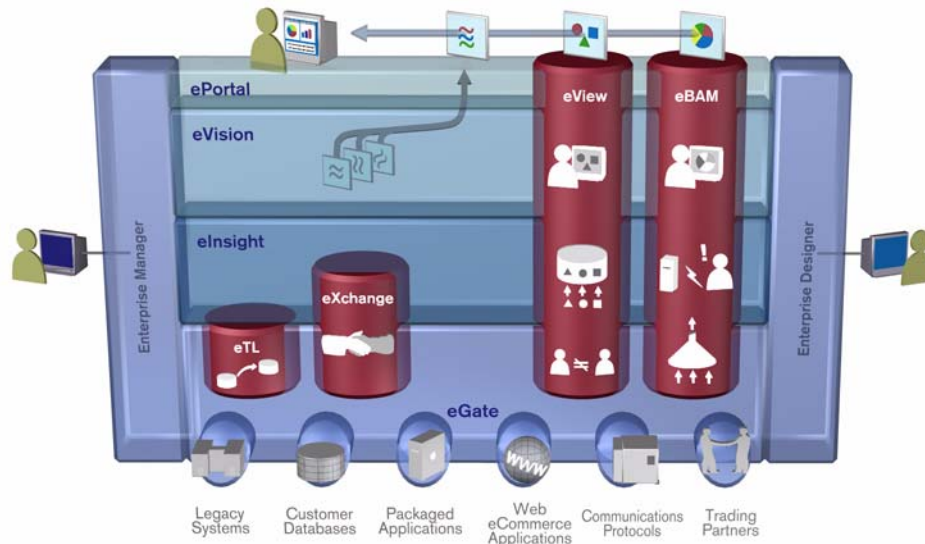
12.1.1 Features and Functions

Users create eBAM Services that contain a key element called a Data Definition. The Data Definition specifies the following:

▪ The raw data to be captured and stored by the eBAM engine .

▪ The datatype to enable mathematical calculations.

▪ The length of time to retain data before flagging it for removal.

▪ The frequency at which expired data is removed.

Based on user-define threshold conditions, eBAM Studio checks for notification conditions at specified intervals. Depending on the business processes that have been set up, eBAM can deliver notifications to assigned destinations, execute stored queries, update data, and/or deliver user-defined views of the data for visual presentation in chart format.

eBAM Studio delivers a real-time view of business activities (for example, monitoring service-level agreements for enforcement), allowing the business analyst to identify problems and trends and resolve them proactively. Digital dashboards can be used as a standalone interface, or can be securely accessed from within a portal.

The business analyst can define the rules for monitoring business activities; eBAM provides the run-time environment for the ongoing execution of the rules, delivering notifications and updates to users via e-mail or the Web.

12.2 Using eBAM Studio

Creating an eBAM application begins when the business analyst specifies the enterprise systems and data to be monitored. The business analyst then defines the polling intervals and thresholds that will trigger automated system responses.

When deployed and activated, the eBAM application listens for specified events. When messages are collected, the eBAM application aggregates and filters the data items of interest, constructs the KPIs, measures them against the business rules and alert thresholds, and delivers the results. See Figure 37.

Figure 37   eBAM Information Delivery
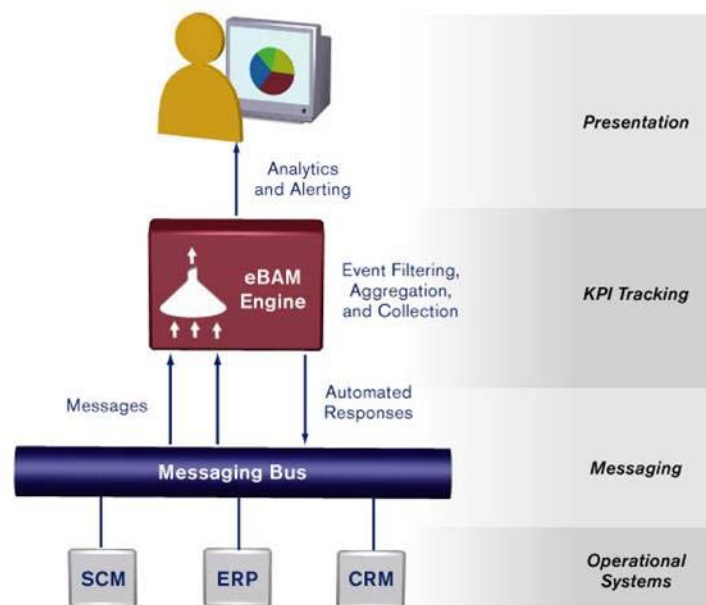
12.2.1 Charts

eBAM provides a wide variety of charts—not just pie charts and bar charts, but also dial-type meters—to display KPI results in a clear, appealing, and understandable graphical context, and to provide real-time, persistent graphical monitoring of business processes over the Web.

12.2.2 Alerts

Notifications are triggered according to preset threshold limits set by the designer. eBAM generates alerts when KPIs fall below, or exceed a specified threshold. The alerts

can be used to send e-mail messages and/or to launch other processes that can be designed to perform an array of notification actions.

12.2.3 Queries

The eBAM graphical palette can be used to construct queries of mid-stream data traffic flowing through an eBAM application. A query is a stored procedure whose results are made available via a Web service; the service is used as a Business Process activity within eBAM (or eInsight, or in an eVision Page Flow), often using run-time input. Queries can reflect values of the data fields that match the defined conditions, as well computed KPI results. Query results can be further manipulated by and mapped to other ICAN components.

Glossary

BI
Business integration (also Business Intelligence).

Collaboration
A logical operation performed between some combination of message destinations and external applications. The operation is defined by a Collaboration Defintion, which can be encoded in either Java or XSLT.

Also see **"Service"** and **"Collaboration Definition"**.

Collaboration Definition
The encoding of business rules, in Java or XSLT format. Typically, the encoding consists of operations on OTDs (see **"OTD"**). Several Collaborations can have the same Collaboration Definition.

Connection
Consists of the configuration information that enables an eWay to connect to an external system.

Connectivity Map
Contains business logic and routing information about the data transmission. A Connectivity Map usually includes one or more Collaborations, topics, queues, and eWays. A Connectivity Map is created under a Project. A Project may have multiple Connectivity Maps.

Constants
A name or value pair that is visible across a Project.

CRM
Customer Relations Management

Data Cleansing
Data must be cleansed of errors in structure and content before it is useful in data warehousing and integration; this means transforming data for accurate and effective use in a database or data management system by cleansing "dirty" or redundant data.

Data Dictionary
Defines the organization of a database and lists all files in the database, the number of records in each file, and the names and types of each field. The data dictionary is often hidden from end users. Although the dictionary doesn't contain actual data, it does contain essential information for managing the database.

Data Integrity
Refers to the accuracy and validity of data. Data integrity can be compromised in many ways, including human error through data entry, or through faulty logic in programming. Computer viruses, software bugs and many other factors can also compromise data integrity.

Data Mapping
In relational databases (RDBMSs) data mapping is the relationship and data flow between source and target objects. Mapping involves structuring the relationship between source and target objects.

Data Mart
A smaller, focused, database designed to help managers make business decisions. (A data warehouse is a larger, enterprise, database(s).)

Data Mining
Used to synthesize or isolate unique data patterns to predict future behaviors or to filter data to select patterns that help discover previously unknown relationships among data. Commonly used by marketers who acquire and distill consumer information.

Data Transformation
Data transformation is necessary after extracting data from legacy data formats, or any format that requires cleansing. Data is transformed for efficient use for Business-to-Business Enterprise Data Integration.

Data Warehouse
A copy or view of enterprise transaction data (sometimes non-transaction data) that is used for reporting. The data is often summarized and always structured for queries and analysis.

Deployment Profile
Contains the information about how the Project components will be deployed in an Environment. A Project can have multiple Deployment Profiles, but only one Deployment Profile can be activated for a Project in any one Environment.

Derived Collaboration
Collaboration that inherits operations from another, according to standard object-oriented practice.

Dimension Table
Dimension tables describe the business entities of an enterprise; also called lookup or reference tables.

Dirty Data
Dirty data contains, but is not limited to, incorrect data including spelling errors, punctuation errors, incorrect data referencing, incomplete, inconsistent, outdated, and redundant data.

Drill Down

To move from summary to more detailed data by "drilling down" to get it. In database terminology this might mean starting with a general category and drilling down to a specific field in a record.

eGate System

See **"Project"**.

Environment

A collection of physical resources and their configurations that are used to host Project components. An Environment contains logical hosts and external systems.

EPR

Enterprise Resource Management

ETL

Extract, Transform, Load. Extract is the process of reading data from a source database and extracting the desired subset of data. Transform is the process of converting the extracted data from its previous form into the desired form. Load is the process of writing the data into a larger database.

eView Manager Service

A component of an eView master index that provides an interface to all components and includes the primary functions of the master index.

eWay

A link between a Collaboration and an external connection including the message server connection (topic or queue) or external application.

External Application

A logical representation in an eGate Project of an external application.

External System

A representation in an eGate Project of an external application system.

Extraction

Data are extracted from a source using software tools. This first step in ETL initially "gets" the data.

Fact Table

A fact table typically contains two types of columns: those containing facts and those that contain foreign keys to dimension tables. Fact tables contain detail facts and/or summary facts.

ICAN Suite

The SeeBeyond Integrated Composite Application Network Suite.

Integration Server
J2EE software platform that houses the business logic container used to run Collaborations and JCA connectors (eWays). Provides transaction services, persistence, and external connectivity.

JMS IQ Manager
JMS-compliant, guaranteed delivery store, forwarding, and queueing service.

Join
Matches records, which are joined by a common field, in two tables in a relational database. Often part of a Select query.

Link
The JMS Connection between a Collaboration and a topic or queue in a JMS-compliant message server.

Linked Message Destination
A reference to a Message Destination defined in another Connectivity Map.

Logical Host
An instance of the eGate runtime Environment that is installed on a machine. A Logical Host contains the software and other installed components that are required at runtime, such as application and message servers.

Management Agent
Uses J2EE technology to manage and monitor an eGate 5.0 deployment that may contain other application servers in addition to the SeeBeyond Integration Server. Defines management interfaces and services designed for distributed environments, focusing on providing functionality for managing networks, systems, and applications.

master index
A database application that stores and cross-references information on specific objects in a business organization, regardless of the computer system from which the information originates. Also called *enterprise-wide master index*.

Matching Service
A component of an eView master index that contains the logic for the matching process.

Message Destination
A general term for a topic or queue. Two or more Projects can share a message destination that has the same name and is deployed on the same message server. A single Project may also have a single message destination referenced in multiple Connectivity Maps.

Metadata
"Data about data." Metadata describes "how," "when," and "who" about structure and format, of a particular set of data. ETL tools are used to generate and maintain a central metadata repository.

Non-normalized Data

Non-normalized data cannot be cross-referenced accurately, if at all, and causes manageability issues. Non-normalized data may be converted to normalized data.

Normalized Data

Normalization is a common database design process used to remove redundant or incorrect organization and data. The design and normalization of the database will create a maintainable data set that can be cross-referenced.

Normalized data is not only easier to analyze but also easier to expand. Normalization involves removing redundancy and correcting incorrect data structure and organization.

OLAP

Online analytical processing.

OTD

An acronym for Object Type Definition. OTDs contain the data structure and rules that define an object. An OTD is used in Java Collaboration Definitions for creating data transformations and interfacing with external systems.

Project

Contains a collection of logical components, configurations, and files that are used to solve business problems. A Project organizes the files and packages and maintains the settings that comprise an eGate system in SeeBeyond's Enterprise Designer.

Query

A request for information from a database. There are three query methods:

Choose – With this easy-to-use method, the database system presents a list of parameters from which you can choose. This method is not as flexible as other methods.

Query by example (QBE) – With this method, the system lets you specify fields and values to define a query.

Query language – With this method, you have the flexibility and power to make requests for information in the form of a stylized query using a query language. This is the most complex and powerful method.

Query Builder

A component of an eView master index that defines how queries are processed.

Queue

A JMS queue is a shareable object that conforms to the *point-to-point* (p2p, or PTP) messaging domain, where one sender delivers a message to exactly one receiver. When the SeeBeyond JMS IQ Manager sends a message to a queue, it ensures it is received once and only once, even though there may be many receivers "listening" to the queue. This is equivalent to the subscriber pooling in other queue implementations. You can reference a queue that exists in another Connectivity Map or Project.

Raw Data

Data that has not been turned into "information," through processing. Although factual and "real," raw data is unorganized.

Relational Database (RDBMS)

Short for Relational Database Management System, most often referred to as RDBMS. Data is stored in related tables. Relational databases can be viewed in many different ways.

In this system a single database can be spread across several tables. (RDBMS differs from flat-file databases where each database is self-contained as a single file or table.)

Repository

Stores and manages the setup, component, and configuration information for eGate Projects. The Repository also provides monitoring services for Projects, which include version control and impact analysis.

Schema Runtime Environment

An add-on in eGate 5.0 that provides the upgrade path for e*Gate 4.x users to upgrade to eGate 5.0. Also known as the SRE.

Service

Contains the information about executing a set of business rules. These business rules can be defined in a Java Collaboration Definition, XSLT Collaboration Definition, Business Process, eTL Definition, or other service. A Service also contains binding information for connecting to JMS Topics, Queues, eWays, and other services.

Single Best Record

Also known as the SBR, this is the best representation of an entity's information in an eView master index. The SBR is populated with information from all source systems based on the survivor strategies defined for each field.

Staging Data

Data that is to be processed before entering the warehouse.

Subproject

An independent Project that is included as part of another Project and listed on the Enterprise Explorer tree beneath the main Project icon.

Table

Refers to data arranged in rows and columns, like a spreadsheet. In relational database management systems, all information is stored in tables.

Topic

A JMS topic is a shareable object that conforms to the *publish-and-subscribe* (pub/sub) messaging domain, where one publisher broadcasts messages to potentially many subscribers. When the SeeBeyond JMS IQ Manager publishes a message on a topic, it ensures that all subscribers receive the message.

Transformation

Data that are extracted from databases are transformed into a desired form, using various tools that cleanse, merge, purge, aggregate, calculate, audit, remove redundancy, standardize, etc.

XSLT

An acronym for Extensible Stylesheet Language Transformations. A file format used in eGate to generate Collaboration Definitions.

e*Gate 4.x Terms in eGate 5.0

Table 1 provides definitions for the terms that are new with eGate release 5.0, as well as equivalent terms from eGate release 4.x.

Table 1   eGate 5.0 Terms

5.0 Term	4.x Equivalent Term
Collaboration	Collaboration
Collaboration Definition	Collaboration Definition
Connection	eWay Connection
Connectivity Map	Closest: Network View of an entire Schema
Deploy	Run the Control Broker
Deployment	<none>
Deployment Profile	Closest: Schema
Enterprise Designer	Enterprise Manager
Enterprise Manager	Enterprise Monitor
Environment	Schema (except only includes physical information, not business logic)
eWay	eWay Connection eWay
eWay Configuration	eWay Connection Configuration
External Application	eWay Connection
External System	eWay Connection
JMS Connection	eWay Connection
Integration Server	<none>
Link	JMS eWay Connection
Linked Message Destination	<none>
Logical Host	Participating Host
Message Destination	Topic or queue
Message Server	JMS IQ Manager
Object Type Definition (OTD)	Event Type Definition (ETD)
Process Manager	Control Broker
Project	Schema (except not including physical layer)
Queue	JMS queue
Repository	Registry
Subproject	Schema

91

Table 1   eGate 5.0 Terms (Continued)

5.0 Term	4.x Equivalent Term
Topic	JMS topic
XSLT	<none>

Index