



Overview: Siebel Enterprise Application Integration

Version 8.0

December 2006

ORACLE®

Copyright © 2005, 2006, Oracle. All rights reserved.

The Programs (which include both the software and documentation) contain proprietary information; they are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright, patent, and other intellectual and industrial property laws. Reverse engineering, disassembly, or decompilation of the Programs, except to the extent required to obtain interoperability with other independently created software or as specified by law, is prohibited.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. This document is not warranted to be error-free. Except as may be expressly permitted in your license agreement for these Programs, no part of these Programs may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose.

PRODUCT MODULES AND OPTIONS. This guide contains descriptions of modules that are optional and for which you may not have purchased a license. Siebel's Sample Database also includes data related to these optional modules. As a result, your software implementation may differ from descriptions in this guide. To find out more about the modules your organization has purchased, see your corporate purchasing agent or your Siebel sales representative.

If the Programs are delivered to the United States Government or anyone licensing or using the Programs on behalf of the United States Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS. Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the Programs, including documentation and technical data, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement, and, to the extent applicable, the additional rights set forth in FAR 52.227-19, Commercial Computer Software--Restricted Rights (June 1987). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

The Programs are not intended for use in any nuclear, aviation, mass transit, medical, or other inherently dangerous applications. It shall be the licensee's responsibility to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of such applications if the Programs are used for such purposes, and we disclaim liability for any damages caused by such use of the Programs.

Oracle, JD Edwards, PeopleSoft, and Siebel are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

The Programs may provide links to Web sites and access to content, products, and services from third parties. Oracle is not responsible for the availability of, or any content provided on, third-party Web sites. You bear all risks associated with the use of such content. If you choose to purchase any products or services from a third party, the relationship is directly between you and the third party. Oracle is not responsible for: (a) the quality of third-party products or services; or (b) fulfilling any of the terms of the agreement with the third party, including delivery of products or services and warranty obligations related to purchased products or services. Oracle is not responsible for any loss or damage of any sort that you may incur from dealing with any third party.

Contents

Chapter 1: What's New in This Release

Chapter 2: Siebel EAI Overview

About Business Process Integration	7
About Siebel EAI	7
About the Components of Siebel EAI	8
XML and Siebel EAI	9
Siebel EAI Adapters and Connectors	9
Virtual Business Components	10
External Business Components	11
Siebel EAI and Java/J2EE	11
Object Interfaces and Siebel EAI	13
Siebel EAI Message-Based Integration	13
Integration Objects	14
Data Transformation	14
Cross-Application Process Integration and Workflow Control	15
About Siebel Application Services Interfaces	15
About Web Services	17
Web UI Dynamic Developer's Kit	18
Siebel Web Services and Security	18
About Unicode Support and Siebel EAI	19
Inbound and Outbound Unicode Scenarios for XML Messages	21
Using the Transcode Business Service for Non-XML Data	21
About Legacy Encapsulation	22
About Peer-to-Peer Integration	23
About Connecting Applications Using Integration Servers	24
About Other Integration Strategies	25

Chapter 3: Understanding Integration

About the Options for Your Integration	27
About Designing Your Application Integration Project	28

About Usage Models 29
 Using the Real-Time Model in Your Integration 30
 Using the Batch Model in Your Integration 31
About Exchanging Data with Siebel Applications 31

Chapter 4: Siebel EAI Architecture

About the Siebel EAI Architecture 34
About Integration Requirements and Siebel EAI 35
About Business Process Coordination Using Workflows 36
About Transport Mechanisms 37
About High-Volume Data Exchange 37

Chapter 5: Custom EAI Business Scenarios

Determining Your Integration Needs 39
About Common Integration Scenarios 40
About Data Access and Replication 41
 Sending Contact Information to an External System 41
 Providing Access to a Company Catalog 42
 Accessing Data Using Siebel Java Beans 42
 Accessing Customer History Information Without Replication 42
About Data Across the Enterprise 43
 B2B Integration and Siebel EAI 43
 Integrating Legacy Data 44
About Performing Data Transformation 44

Index

1

What's New in This Release

What's New in Overview: Siebel Enterprise Application Integration, Version 8.0

Table 1 lists changes described in this version of the documentation to support Release 8.0 of Oracle's Siebel software.

Table 1. New Product Features in Overview: Siebel Enterprise Application Integration, Version 8.0

Topic	Description
Chapter 2, "Siebel EAI Overview"	Deleted references to Universal Application Network (UAN). For more information, see FAQ 2272 on Siebel SupportWeb.
"About Siebel EAI" on page 7	Deleted references to Microsoft BizTalk and OLE DB. Consolidated duplicate topics.
"About the Components of Siebel EAI" on page 8	Consolidated duplicate topics.
"About Siebel Application Services Interfaces" on page 15	Consolidated duplicate topics.
"About Web Services" on page 17	Consolidated duplicate topics.
"About Connecting Applications Using Integration Servers" on page 24	Updated links for finding validated adapters for Siebel applications.
"About Transport Mechanisms" on page 37	Deleted reference to BizTalk.

2

Siebel EAI Overview

This chapter consists of the following topics:

- [“About Business Process Integration” on page 7](#)
- [“About Siebel EAI” on page 7](#)
- [“About the Components of Siebel EAI” on page 8](#)
- [“About Siebel Application Services Interfaces” on page 15](#)
- [“About Web Services” on page 17](#)
- [“About Unicode Support and Siebel EAI” on page 19](#)
- [“About Legacy Encapsulation” on page 22](#)
- [“About Peer-to-Peer Integration” on page 23](#)
- [“About Connecting Applications Using Integration Servers” on page 24](#)
- [“About Other Integration Strategies” on page 25](#)

About Business Process Integration

Organizations need to integrate business processes across a large number of disparate applications in order to drive revenue growth, increase productivity, and gain visibility into business performance. The velocity of business is accelerating and only by integrating and streamlining their business processes can organizations achieve the agility to respond to fast-changing customer and market demands in real time. The integration imperative is even more pressing as supply and demand chains have become increasingly interdependent, requiring organizations to integrate the extended enterprise to make sure the viability of their entire partner ecosystems.

To meet these requirements, Siebel Systems continues to enhance the integration tool set within Siebel applications—Siebel Enterprise Application Integration (Siebel EAI).

About Siebel EAI

The Siebel Business Platform includes a set of tools, technologies, and prebuilt functional integrations that facilitate application integration. This set of products is referred to as Siebel Enterprise Application Integration (Siebel EAI).

Siebel EAI provides components for integrating Siebel Business Applications with external applications and technologies within your company and is designed to work with third-party solutions such as those from IBM, TIBCO, WebMethods, and others.

Siebel EAI provides bidirectional real-time and batch solutions for integrating Siebel applications with other applications.

Siebel EAI is designed as a set of interfaces that interact with each other and with other components within Siebel application. These interfaces:

- Allow a flexible service-based architecture, built on top of configurable messages using XML and other formats.
- Are compatible with IBM MQSeries; Microsoft MSMQ; Sun Microsystems Java and J2EE; XML, and HTTP, and many other standards.
- Expose internal Siebel Objects to external applications.
- Take advantage of prebuilt adapters and enterprise connectors, and are compatible with third-party adapters and connectors.
- Allow for data transformation.
- Integrate external data through Virtual Business Components (VBCs) and External Business Components (EBCs).
- Provide a graphical business process designer, programmatic interfaces, and a high-volume batch interface.

About the Components of Siebel EAI

Legacy applications that have been deployed for various purposes—including purchasing, accounting, and ERP—contain critical information that needs to coexist and be integrated with Siebel applications. In addition, B2B (business-to-business) interaction necessitates the sharing of customer, order, and account information with partners across the firewall. Siebel EAI provides various components that can be used individually or together with an EAI vendor's toolkit, including XML support in Siebel EAI adapters and connectors, VBCs, and Java Beans.

This section describes the following topics:

- ["XML and Siebel EAI" on page 9](#)
- ["Siebel EAI Adapters and Connectors" on page 9](#)
- ["Virtual Business Components" on page 10](#)
- ["External Business Components" on page 11](#)
- ["Siebel EAI and Java/J2EE" on page 11](#)
- ["Object Interfaces and Siebel EAI" on page 13](#)
- ["Siebel EAI Message-Based Integration" on page 13](#)
- ["Integration Objects" on page 14](#)
- ["Data Transformation" on page 14](#)
- ["Cross-Application Process Integration and Workflow Control" on page 15](#)

XML and Siebel EAI

XML is increasingly the format of choice for data exchange between applications and partner organizations. The metadata definition for XML is in the form of XML Schema Definitions (XSD) or Document Types Definitions (DTDs). Siebel Integration objects can be represented as XSD or DTDs. This allows other applications to capture metadata from Siebel applications and understand how to format an XML document that needs to be sent to Siebel applications. Any data in Siebel applications can be represented as XML and sent over a standard protocol such as HTTP. Siebel EAI provides components that allow bidirectional exchange of XML documents over the firewall using the HTTP protocol. This exchange can be made secure at the transport layer by using the HTTPS protocol.

Siebel EAI supports XML for outgoing and incoming messages. Siebel EAI can be configured to use externally specified XSD or DTDs for defining external integration objects. Siebel Tools includes an Integration Object Wizard for importing and processing external XSD or DTDs and generating these external integration objects that represent the XSD or the DTDs.

All transport adapters, whether built by Siebel Systems or built by partners, can use this business service to dynamically look up the content in an XML document or other format documents and route the message based on this content to a workflow or other business services within a Siebel application.

Siebel EAI Adapters and Connectors

Siebel EAI provides adapters and connectors to help create integrations between Siebel applications and external applications. There are also additional connectors developed by Siebel Business Applications partners. For each supported business process, the connectors include built-in data mapping between the exchanged entities, as well as sequencing and error handling.

Siebel EAI Adapters

An adapter is a Siebel EAI component that provides the low-level interface mechanism to allow one application to talk to another. In Siebel EAI, there are three kinds of adapters: the EAI Siebel Adapter, the EAI UI Data Adapter, and the Transport Adapters.

- The EAI Siebel Adapter business service is specifically designed to interact with the Siebel business object layer. It is a business service in Siebel applications that can take as input an XML document or a property set that conforms to an integration object definition in the Siebel system. EAI Siebel Adapter can then query, insert or update, delete, or synchronize data with the Siebel business object layer.
- The EAI UI Data Adapter business service allows custom-built Web applications access to business components and business objects. This adapter publishes strongly typed APIs to allow data centric object manager operations, such as insert, delete, update, query and execute. The EAI UI Data Adapter allows both queries by example and predefined queries.

- Transport adapters are business services within Siebel EAI that are designed to communicate with other protocols and technologies outside of Siebel applications to transport data from the Siebel system to another system. The HTTP transport in Siebel EAI allows Siebel applications to communicate over the HTTP protocol. The MQSeries adapter allows Siebel applications to communicate with the IBM MQSeries messaging system. The MSMQ adapter allows Siebel applications to communicate with the Microsoft MQ messaging system.

For more information on the EAI Siebel Adapter and the EAI UI Data Adapter business services, see *Integration Platform Technologies: Siebel Enterprise Application Integration*.

For more information on transport adapters, see *Transports and Interfaces: Siebel Enterprise Application Integration*.

Siebel EAI Connectors

Siebel EAI connectors provide low-level connectivity to other back-office applications, such as SAP and Oracle, but also include the business processes used for connecting to the external application. For example, Siebel Connector for SAP R/3 provides connectivity using BAPI and IDOC transport adapters, as well as predefined business processes. Using the connectors, you can exchange customer, order, and product information between Siebel applications and other Oracle applications or SAP.

For more information, see *Siebel Connector for SAP R/3* and *Siebel Connector for Oracle Applications*.

Third-Party Connectors

Your organization may have a need to integrate multiple applications, such as Siebel applications, SAP, Oracle, I2, and so on, with one another. If so, you might need an EAI solution from one of the EAI vendors. These vendors work closely with Siebel Systems to develop adapters to Siebel applications using Siebel EAI. Siebel Systems has a validation program to make sure that these adapters work according to Siebel Systems standards.

Virtual Business Components

Virtual business components (VBCs) are mechanisms in Siebel EAI by which data from an external system can be viewed in Siebel applications without having to replicate that data within the Siebel Database. For example, your users could access detailed credit card information from a mainframe system for a given customer account and see it displayed in the Siebel application interface. In this case, the basic customer account data, such as account number, name, and so on, is stored in Siebel applications while the detailed transaction information on that customer's credit card account is stored in an external database. The detailed transaction information for a given account can be retrieved from the external system on demand without replicating that information in the Siebel applications.

VBCs are configured in Siebel Tools and use business services to access data from an external system. VBCs can use standard transports like MQSeries, HTTP, and MSMQ, along with the XML Gateway Service, to query, insert, and update data.

External Business Components

External Business Components (EBCs) provide a way to access and view relational data sources that reside in a non-Siebel table or view using a Siebel business component. Unlike VBCs that require a business service, EBCs can use Siebel's database connectors to access data.

EBCs are configured in Siebel Tools using a Table object definition in the Siebel Repository and a physical table that resides in the same Siebel database instance or subsystem that contains other Siebel tables.

Siebel EAI and Java/J2EE

Siebel Business Applications provide standards-based technologies that allow you to access J2EE components from Siebel applications and support the creation of Java/J2EE components to access Siebel objects.

For details on J2EE, see *Transports and Interfaces: Siebel Enterprise Application Integration*.

Accessing Java/J2EE Components from Siebel Applications

Siebel Business Applications provide three mechanisms for invoking Java/J2EE components (JSP, Servlets, or EJBs):

- Web Services
- The Outbound HTTP adapter
- Java Business Service

When interacting with J2EE components published as a Web Service, Siebel applications can consume the WSDL (Web Service Description Language) document describing the service and operations, and generate a proxy Business Service allowing the Siebel Business Application to invoke the Web Services just like a local object. When the Business Service is invoked, the Object Manager detects that the Business Service is a proxy to a Web Service and generates the appropriate SOAP (Simple Object Access Protocol) message and dispatches the request using a configured transport such as HTTP.

For more information on Web Services, see ["About Web Services" on page 17](#).

Additionally, Siebel EAI allows you to interact with J2EE components using the Outbound HTTP transport adapter. You can use this when including external content within the Siebel user interface, or when the component you need to interact with does not support a Web Service (SOAP or WSDL) interface.

The Java Business Service allows the sending or receiving of messages through a JMS. The JMS Receiver server component, in fashion identical to the MQ Series Receiver component, allows the asynchronous receipt of messages, except you are using EAI JMS Business Service instead of EAI MQSeries Server Transport.

The Java Business Service is a business service that allows custom business services to be written in Java and accessed from the Siebel code.

The Java Business Service uses the Java Native Interface API provided by Java Virtual Machines (JVMs) and native code interaction. It is a widely used technology used for writing JDBC drivers.

Accessing Siebel Applications from Java/J2EE Components

Java/J2EE components can request information from Siebel applications using a variety of methods including:

- Siebel Java Data Bean
- Siebel Resource Adapter
- Web Services
- Java Message Service

Java Data Bean. The Java Data Bean is a collection of Java classes that allow developers to interact with a variety of Siebel objects such as business objects, business components, and so on. Using this interface, you can develop Java/J2EE components that interact with Siebel applications.

Siebel Resource Adapter. The Siebel Resource Adapter plays a central role in the integration and connectivity between Siebel applications and a Java application server. It serves as the point of contact between application components, application servers and enterprise information systems. A resource adapter, along with the other components, must communicate with each other based on well-defined contracts that are specified by the J2EE Connector Architecture.

Web Services. Web Services are emerging as an important technology for exposing application functionality independent of the underlying technology used to provide that functionality. Release 6.x introduced the notion of Business Services that could be invoked through XML over HTTP and MQSeries. This functionality is now the basis for supporting Web Services. Siebel Business Applications provide support for Simple Object Access Protocol (SOAP) and Web Services Definition Language (WSDL). These two standards provide the basis for Web Services and allow for interoperability between .NET, J2EE, and leading packaged applications such as Siebel Business Applications.

For more information on Web Services, see [“About Web Services” on page 17](#).

Java Message Service. Enterprise messaging is recognized as a tool for building enterprise applications. The Java Message Service (JMS) is a standard Java API for accessing enterprise messaging systems. JMS is part of the Java 2 Platform, Enterprise Edition (J2EE). JMS supports the ability to asynchronously send and receive business data and events. JMS defines a common enterprise messaging API that is designed to be easily and efficiently supported by a wide range of enterprise messaging products. JMS supports both message queuing and publish-subscribe styles of messaging.

JMS is not a separate product. It is the specification of a common API for enterprise messaging. A JMS provider supplied by an enterprise messaging vendor is required to use it. Using the Java Business Service (JBS) feature, the JMS Transport is a feature that allows users to access third-party vendors' JMS providers. A JMS provider is a particular vendor's implementation of the JMS API. For example, IBM implements a JMS provider for their MQSeries product, and BEA implements a JMS provider for their Weblogic product.

For more details on the Java Message Service, see *Transports and Interfaces: Siebel Enterprise Application Integration*.

Object Interfaces and Siebel EAI

Siebel applications support several programmatic interfaces to facilitate communication between external applications and Siebel applications. These interfaces include:

- COM Data Control
- Java Data Bean
- Web Client Automation Server
- Mobile Web Client Automation Server
- COM Data Server

For more information about object interfaces, see *Siebel Object Interfaces Reference*.

Siebel EAI Message-Based Integration

As shown in [Figure 1](#), Siebel EAI supports message-based integration with external applications where Siebel application is the initiator of the request or the receiver. The contents of the messages exchanged are called *integration objects*. Integration objects are defined in Siebel Tools.

There are two types of integration objects:

- **Siebel integration objects.** Data to be loaded into or read from a Siebel Business Object.
- **External integration objects.** Data received from or sent to the external application.

A configuration engine provides configurable data transformation for matching the difference in entities representations, while integration objects represent the information exchanged.

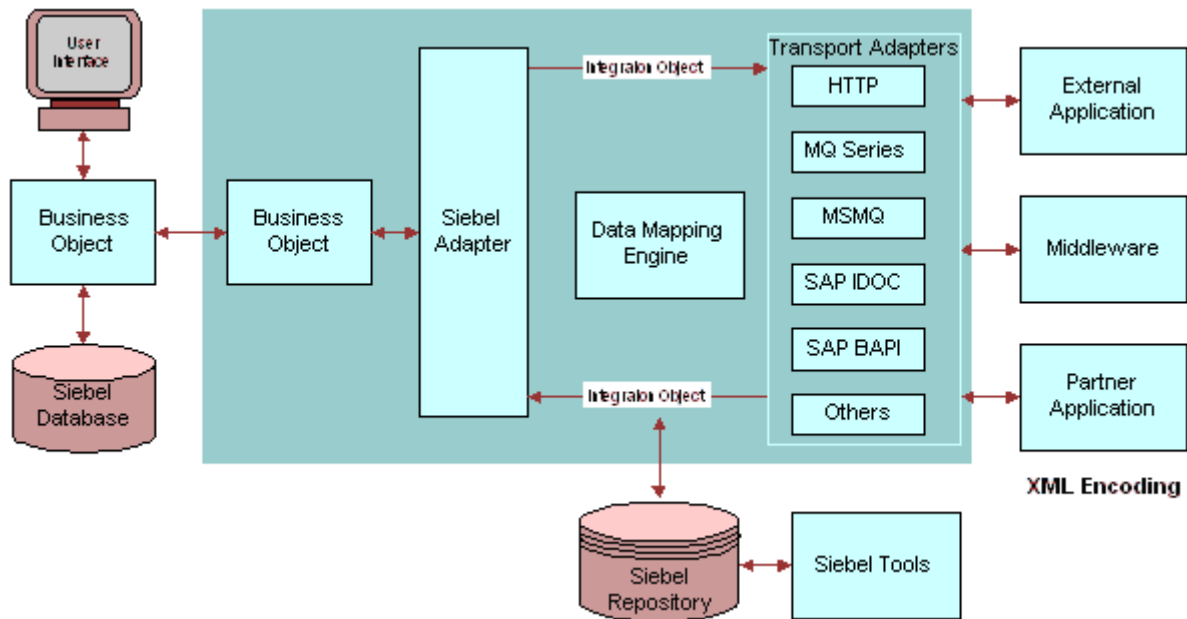


Figure 1. Configurable and Upgradeable Integration

Integration Objects

A Siebel integration object is defined as a subset of a Siebel business object, and the map between an integration object and the business object is maintained in the Siebel Repository. A predefined business service called the EAI Siebel Adapter takes a Siebel integration as input and calls the appropriate business object APIs to operate on the object invoked. External integration objects are based on external metadata. This metadata can be in the form of XML Schema Definitions (XSDs), Document Type Definitions (DTDs), External Data Representations (XDRs), SAP BAPIs and SAP IDOCs, as well as Oracle application metadata. Siebel Tools provides wizards to access this metadata from the external application and create external integration objects based on that metadata within the Siebel repository.

Data Transformation

Siebel EAI provides a declarative data mapper (Siebel Data Mapper) to transform the data expected by Siebel applications (the internal integration object) to the data expected from the external system (the external integration object). A run-time engine called the EAI Data Mapping Engine implements complex domain and structure mapping.

Cross-Application Process Integration and Workflow Control

To achieve process-level integration between Siebel applications and external applications, the Siebel Business platform supports workflow control. You define the integration process using the Workflow Process Designer in Siebel Tools. Workflow processes can be executed in any application object manager as a business service. A business service can be a prebuilt component, such as a transport adapter and the Siebel Data Transformation Engine, or can be custom-developed at design time or run time using one of the Siebel Scripting languages. Workflow processes can also be executed in the Workflow Process Manager server component.

For detailed information on workflow processes, see *Siebel Business Process Framework: Workflow Guide*.

About Siebel Application Services Interfaces

Siebel Application Services Interfaces (ASIs) are prebuilt interfaces to Siebel business processes. These interfaces expose Siebel functionality as services to the application environment, as well as allow Siebel business processes to use services provided by external applications. Siebel ASIs are designed for ease of maintenance and can be configured by an administrator using Web Services Administration screens without requiring in depth knowledge of Siebel EAI.

ASIs provide a release-independent integration interface. The integration interface to the Siebel application remains unchanged with each upgrade to new releases. ASIs are extensible and are accessible as Web Services or through any third-party Enterprise Application Integration (EAI) solution. These benefits mean easier integration between Siebel application and external systems with significant reduction in integration, maintenance, and upgrade costs.

ASIs are published in a standard metadata format, such as Extensible Markup Language Document Type Definitions (XML DTD) and Web Services Descriptive Language/XML Schemas (WSDL/XSD) and are built using the ASI Framework.

There are two types of ASIs, Inbound ASIs and Outbound ASIs. Inbound ASI is used to accept data into Siebel applications from an external system using Siebel workflow processes, Siebel Business Services, and Siebel Data Synchronization services through the Siebel Object Manager as illustrated in Figure 2.

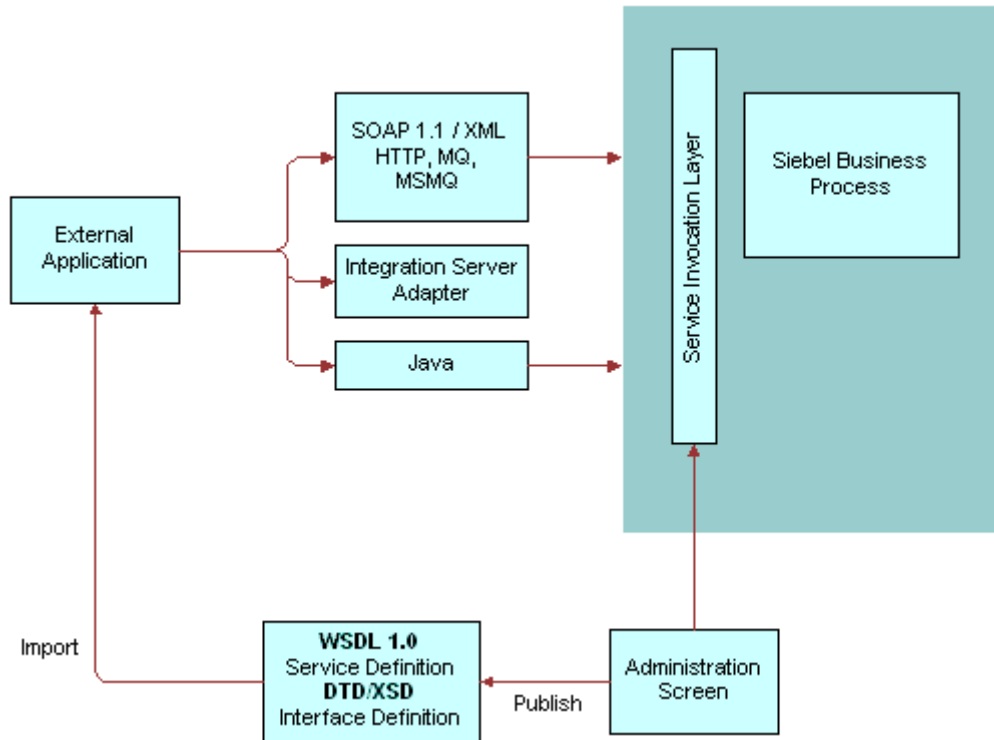


Figure 2. Inbound ASIs

An Outbound ASI is used to send data out based on a UI event or a trigger in your Siebel workflow process as illustrated in Figure 3.

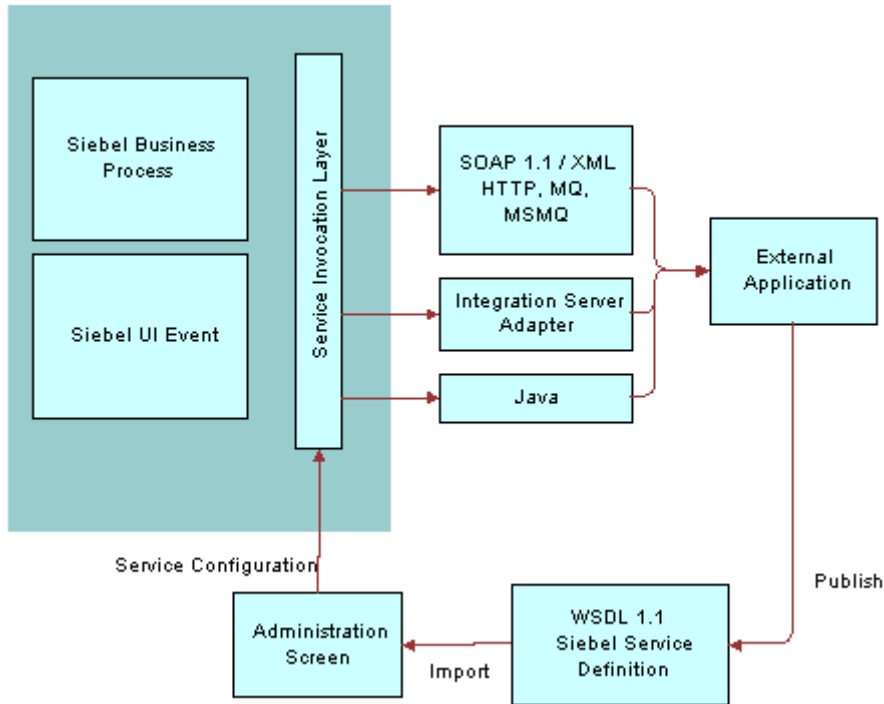


Figure 3. Outbound ASIs

NOTE: For details on ASIs, see *Application Services Interface Reference*.

About Web Services

A Web Service is programmable application logic that is accessible using standard Internet protocols. Web Services combine component-based development and the Internet and can be reused regardless of how the service is implemented. Web Services are accessed through information protocols (such as HTTP) and data formats (such as XML).

Web Services are based on communication protocols that include HTTP, XML, Simple Object Application Protocol (SOAP), and Web Services Description Language (WSDL). A Web Service can be developed on any computer platform and in any development environment as long as it can communicate with other Web Services using these common protocols.

Following are features for use with Web Services:

- **WS-Security Support.** The Web Services (WS) Security specification is a Web Services standard that supports, integrates, and unifies multiple security models and technologies, allowing a variety of systems to interoperate in a platform- and language-independent environment.

- **WS-I Compliance.** By providing the ability to publish a Siebel Web Service as a Document-Literal or Remote Procedure Calls (RPC)-Literal bound Web Service, the Siebel application is conforming to the specification as defined by the Web Services Interoperability Organization's (WS-I) Basic Profile specification. Adherence to this specification makes sure that the Siebel application can interoperate with external Web Service providers.
- **Web Services Cache Refresh.** The Web Services cache is used to store all the global administration information that can be manipulated in the Inbound and Outbound Web Service administration screens. The information contained in these services may need refreshing in order to provide more current or correct functionality. Administrators can directly refresh the memory cache in real time, without stopping and restarting Siebel services.

For more detailed information on Web Services, see *Integration Platform Technologies: Siebel Enterprise Application Integration*.

Web UI Dynamic Developer's Kit

The Web UI Dynamic Developer's Kit (DDK) is based on Web Services and consists of interfaces that allow access to Siebel data and a wizard for generating a quick-start kit. The quick-start kit includes sample JavaServer pages that can be deployed in a Web application and that provide Web developers with sample code demonstrating common data manipulation operations and basic UI rendering. Web developers can modify the sample code instead of developing from scratch, giving them a jump-start on development work.

Key features of the Web UI DDK solution are the following:

- **Web UI DDK Wizard.** This wizard in Siebel Tools is used to generate the development kit that includes sample JavaServer pages and other J2EE artifacts.
- **Sample JavaServer Pages.** These are sample pages generated by the Web UI DDK Wizard and then deployed in a J2EE environment.
- **Data Schema and Metadata Report.** This is a report that Java developers use to understand the Siebel data model.
- **UI Data Adapter.** This Siebel business service exposes APIs for accessing Siebel data.

For detailed information about the UI Data Adapter, see *Integration Platform Technologies: Siebel Enterprise Application Integration*.

For more detailed information on the Web UI DDK and its features, see *Siebel Web UI Dynamic Developer Kit Guide*.

Siebel Web Services and Security

The Siebel Web UI DDK supports Siebel Authentication and Session Management SOAP headers, including Web single-sign on authentication. These security features are part of the Siebel Web Services framework and are summarized below:

- **Siebel Authentication and Session Management SOAP Headers.** Custom Web applications invoke Siebel sessions by sending Web service requests to the Siebel Web Server Extension (SWSE). Requests include Siebel Authentication and Session Management SOAP headers that can include user credentials for logging in and session information for reconnecting to established sessions.
- **Web Single Sign-On.** Siebel Web Services support Web single sign-on deployment scenarios in which third-party applications handle authentication and then pass authentication information to the Siebel application. Once authenticated by the third-party application, users do not have to explicitly log in to the Siebel application.

For more detailed information about Siebel Web services and security, see *Integration Platform Technologies: Siebel Enterprise Application Integration*.

For more information about general security topics, see *Siebel Security Guide*.

About Data Access Control

You can administer access control by associating responsibilities to business services and business service methods. The application is configured so that users, who access a Siebel application from a custom-built Web application user interface, have different levels of access to the Siebel database.

For details about administering access control for business services, see *Siebel Security Guide*.

About Unicode Support and Siebel EAI

Using Unicode, a global company with divisions in different countries with different languages can deploy a single instance of a Siebel application release. However, this provides unique integration challenges since the other systems in their global IT landscape may or may not be Unicode enabled. This issue is addressed by providing two Unicode deployment modes:

- Unicode Siebel Server with Unicode database.
- Unicode Siebel Server with traditional code page database.

For both of these deployment modes, integration of Siebel Business Applications with external applications may involve data replication or data sharing or both. If the code page database is used, then data replication should be allowed only if the code page of the external system matches that of the Siebel Database.

When data is exchanged between applications using different code pages, each character's code point (numeric representation) must be converted from the standard used by the source code page into the equivalent used by the destination code page. This is true when characters are converted from:

- Unicode to a traditional code page, for example UTF-16 to Shift-JS.
- A traditional code page to Unicode, for example Shift-JS to UTF-8.
- One type of Unicode to another type of Unicode, for example UTF-8 to UCS-2.

When there is no equivalent character in the destination code page and the conversion cannot be performed, then a conversion error is generated. Depending on the configuration, the conversion errors may either terminate the current transaction or proceed with the transaction by substituting replacement characters as necessary and ignoring the conversion error.

For details on conversion error handling, see *Transports and Interfaces: Siebel Enterprise Application Integration*.

As illustrated in [Figure 4](#), Siebel EAI supports different adapters such as COM and MQSeries to integrate with external applications. Each of these adapters has to have the ability to:

- Be able to work with the character encoding as specified in the picklist.
- Do conversion to or from the external encoding.
- Handle conversion errors.

NOTE: COM and Java Bean are Unicode protocols. The sender and the recipient are responsible for conversion of the character sets. For example, Siebel application interfaces automatically take care of the conversion so these protocols do not need to do any conversion. Error handling is also handled differently by these protocols compared to the other adapters.

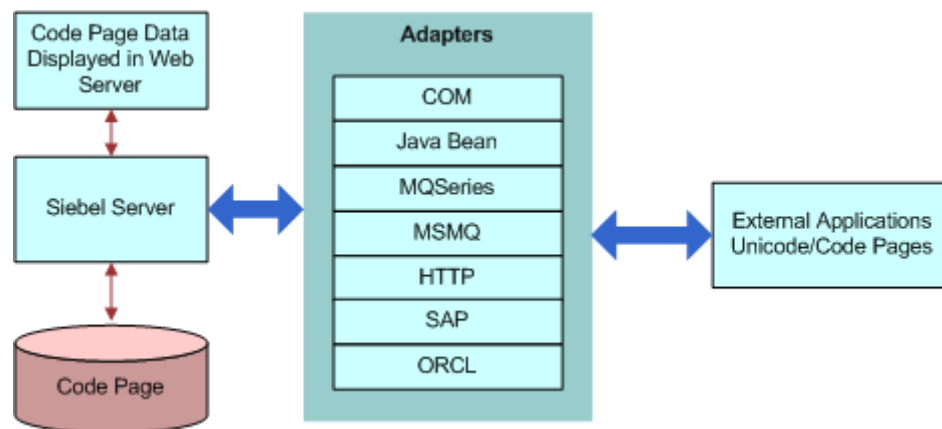


Figure 4. Unicode Conversion

Siebel EAI components and adapters have been enhanced to support code page conversion and to allow you to specify how to handle the conversion errors. The Character Set Conversion for Text Data argument is available in a number of business services such as EAI MQSeries Server Transport, EAI DLL Transport, EAI File Transport, EAI HTTP Transport, EAI MSMQ Transport, EAI XML Converter, EAI JMS Business Service or Transcode. When these business services are invoked from a workflow process the valid set of encodings is controlled by a picklist, but if they are invoked through scripting or similar mechanisms, the character set name is supplied textually. For example, if the Siebel application is communicating with an external system through EAI MQSeries Server Transport, the invocation of the EAI MQSeries Server Transport business service to send or receive a message would also specify the character set the external system sends or expects to receive.

Target character encoding choices are:

- None
- Unicode (UTF-8, UTF-16)
- Traditional Code page

NOTE: For more information on Unicode support and a list of Siebel Character Set Encoding Names, see *Siebel Global Deployment Guide*.

Inbound and Outbound Unicode Scenarios for XML Messages

Because the Siebel application's memory format is Unicode, when sending and receiving XML messages the message has to be converted to the Siebel format or the external system character set. This conversion only needs to be done once for each communication.

The conversion can be done in one of the following ways:

- Using the EAI XML Converter business service in your workflow process.
- Using the mechanism in the transport business service.
- Using the Transcode business service when neither of the above is available.

NOTE: Do not convert your data more than once per communication. Double conversion corrupts your data. If you have an EAI XML Converter in your workflow process, make sure your transport business service is set to ignore conversion.

Using the Transcode Business Service for Non-XML Data

In addition to the improvements to the current Siebel EAI components and adapters, Siebel Business Applications offer the Transcode business service. When non-XML data is exchanged, you can use the Transcode business service in your workflow processes to validate or convert data from one encoding to another. You can do this for both inbound and outbound communication.

Using the Transcode Business Service for Outbound Communication

In an outbound communication between a Unicode Siebel application and a non-Unicode external system, using a Unicode communication channel such as XML, you can use the Validate method of the Transcode business service to take preemptive action to avoid character conversion failures on the external system. You can also use the Validate method to log character conversion errors that occur on the external system, or use the Convert method of the Transcode business service to use a substitute character to avoid errors on the external system.

Using the Transcode Business Service for Inbound Communication

In an inbound communication between a non-Unicode Siebel Database and a Unicode external system, using a Unicode communication channel such as XML, you can use the Validate method to check that the data being sent to the Siebel Database is stored without a character conversion error. If there are character conversion errors, use the Convert method to use substitute characters when appropriate.

NOTE: For details on the Transcode business service and its methods, see the *Siebel Global Deployment Guide*.

About Legacy Encapsulation

In an encapsulated environment, multiple similar legacy applications are rationalized through the construction of high-level APIs that hide underlying complexity. As illustrated in Figure 5, message-oriented middleware such as IBM MQSeries Queue Manager and MQSeries Integrator or Enterprise Java Bean (EJB)-based custom development environments such as IBM WebSphere, help businesses perform transaction decomposition, data mapping, legacy sequencing, and result aggregation. With Siebel EAI, the Siebel Business platform generates and processes the high-level request and response messages expected by the encapsulated environment.

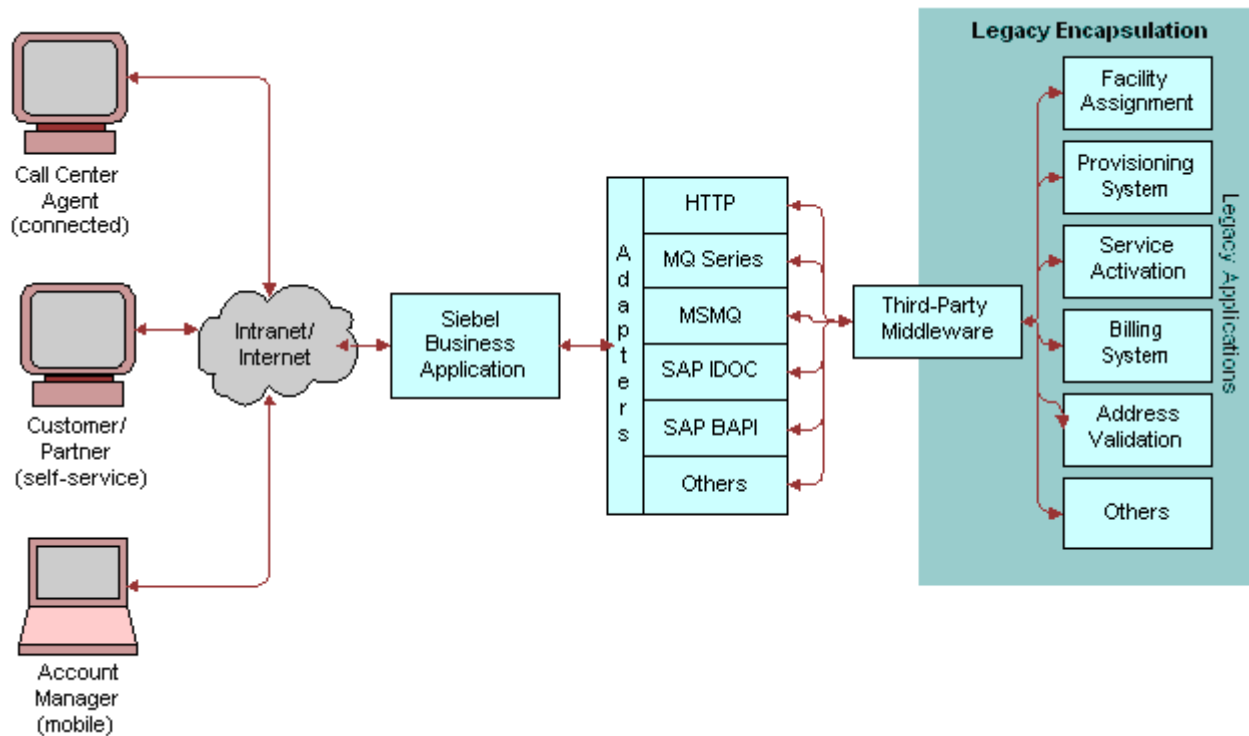


Figure 5. Legacy Encapsulation

About Peer-to-Peer Integration

In peer-to-peer integration, as shown in [Figure 6](#), multiple applications interact with each other through a central software bus. The software bus provides a uniform interface to each application, as well as a number of services. These services range from simple data transformation to publish-and-subscribe schemes. The latter technology allows a set of applications to automatically receive copies of events of interest to them, keeping them in sync with the rest of the organization. Siebel EAI supports the services that the software bus expects from each participating application.

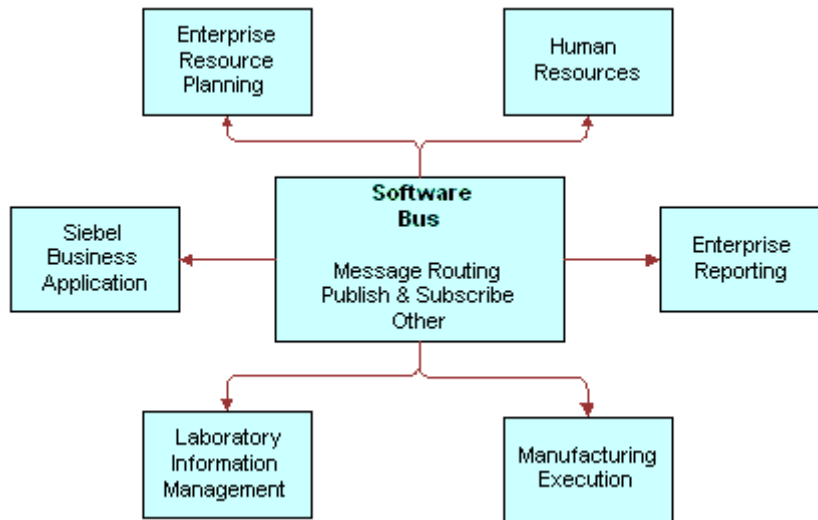


Figure 6. Peer-to-Peer Integration

Many enterprises adopt a software bus strategy for application integration. Master data is partitioned across systems, and the software bus middleware implements a messaging architecture for application data exchange. The messaging functionality ranges from simple message routing, in batch or real time, to a publish-and-subscribe scheme.

NOTE: Siebel EAI supports both batch and real-time integration. Batch integration optimizes interactions for high throughput and therefore uses computing resources more efficiently. By contrast, real-time integration optimizes response time.

About Connecting Applications Using Integration Servers

Another approach to integration is to connect applications using integration servers, which provide an information bus or a hub-and-spoke architecture to tie disparate applications together. The benefit of using integration servers is that they reduce the number of physical connections required to Order N . As illustrated in Figure 7, each application only needs to connect once through an adapter to the integration server. The integration servers also provide a scalable and reliable integration infrastructure, which is successfully implemented in large and critical deployments.

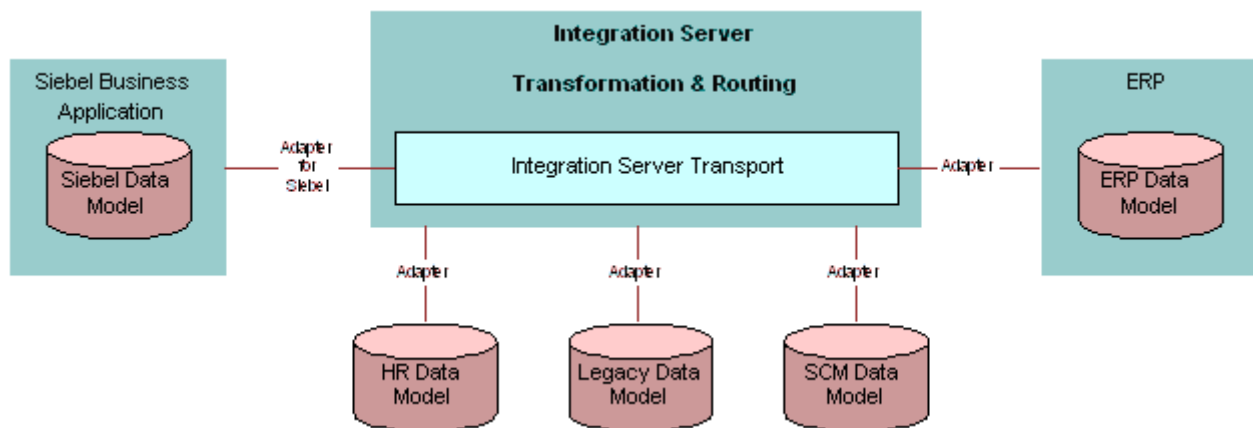


Figure 7. Integration Server

Siebel applications can integrate with an integration server using adapters provided by vendors such as IBM, TIBCO, and WebMethods. These adapters are validated by Siebel Systems.

To find a list of validated adapters for Siebel applications

- 1 Navigate to <http://www.siebel.com/partners/dir/directory.shtm> in your Web browser.
The Siebel Alliances page appears, which has a form to search for partners.
- 2 Enter a partner's name to search by and click Submit or search by making selections from the following partner categories:
 - Consulting Partners
 - Software/Content Partners
 - Platform Partners
 - SMB Partners
- 3 Click on a partner's name to find more details about that partner and its validated adapters.

About Other Integration Strategies

Other integration strategies are applicable as well. These include master data partitioning, data sharing, presentation layer integration (screen scraping), and others.

- **Cross-Application Process Integration.** Successful business demands business processes that span across multiple applications—for example, a quote to cash business process spans across credit check, inventory management, billing, shipping and many other application services. Siebel EAI facilitates such cross-application business processes.
- **Master Data Partitioning.** Master data refers to the set of data required by a community of applications. Master data is partitioned between applications; each piece of data is owned by one and only one application, but the community imposes rules for the creation and update of master data.
- **Data Sharing.** Siebel EAI allows Siebel Business Applications to access external data in real time without having to maintain a local duplicate copy. This is a common requirement in the financial services industry, where the customer information file typically resides on a mainframe.
- **Presentation Layer Integration.** Siebel EAI supports client-side integration in a Web client environment. Client-side integration consists of integrating applications at the user interface level, either by juxtaposing the screens within a portal and coordinating the content, or by *screen scraping* the content of one into the other.
- **Data Mapping and Transformation.** Data shared between applications might not be structured or encoded in the same way. To facilitate integration of nonidentical data structures, Siebel EAI specifies mapping and transformation rules, including filtering and structural changes.
- **Centralized Application Administration.** Siebel Business Applications participate in an application management framework, which facilitates overall administration of all applications.
- **Upgradeable Integration.** The Siebel Business platform is designed so that integration points can be migrated to a new release of Siebel Business Applications and adapted to new releases of external application packages. This capability means organizations can use prior investments in their integrations when upgrading to new versions of Siebel applications or third-party applications.

3

Understanding Integration

Before you begin an integration project, identify your integration needs. Siebel Business Applications need to interact with external systems to provide an integrated view of your information. You may have data in a variety of applications, such as back-office solutions for order fulfillment, manufacturing, or accounting; e-commerce applications such as Web storefronts; and help desk applications.

This chapter consists of the following topics:

- [“About the Options for Your Integration”](#)
- [“About Designing Your Application Integration Project”](#)
- [“About Usage Models” on page 29](#)
- [“About Exchanging Data with Siebel Applications” on page 31](#)

About the Options for Your Integration

There are several ways you can approach your integration, depending on your needs. You can connect Siebel applications and your back office system or e-commerce applications using prebuilt Siebel EAI connectors. You can connect Siebel applications and an external application using Siebel EAI custom-built connectors. You can integrate Siebel applications with middleware solutions. Or you can integrate Siebel applications with external applications using a third-party EAI product.

Siebel Business Applications provide solutions for each element of the EAI planning process.

- Programmatic interfaces, such as COM and Siebel Java Beans provide a tightly coupled integration for real-time data exchange.
- Adapters, such as MQSeries and the HTTP adapters provide a loosely coupled mechanism for real-time integration. This means that one system needs to be aware of implementation details for the other system’s integration mechanisms. A message or a document is exchanged between these systems.
- For high-volume batch integrations, Siebel Business Applications provide you with Siebel Enterprise Integration Manager (EIM). EIM can replicate data between Siebel applications and other systems at the database layer and provides high performance for importing or exporting large volumes of data.

About Designing Your Application Integration Project

There are three questions you must answer as part of the planning phase for any application integration project:

- How do I need to share my organization's data?
- Which access mechanism makes the most sense for my situation?
- Which preexisting connector best applies to my situation?

It is important to identify your need before designing the integration. Your design choice could be based on the following categories:

- **Data Transformation.** Data transformation between two systems is an important part of any integration project. You may want to implement this data transformation using the Siebel-provided data transformation functions or, if your organization has already standardized on a data transformation tool, you might choose to model the integration flow so that it uses the external tool for data transformation.
- **Data Sharing.** If you need to view data from an external source within your Siebel applications and do not need or want to replicate the data, you might use a specific set of objects and tools designed explicitly for this purpose. For example, you might want to view credit card transaction details from within Siebel applications while credit card transaction details are stored on a mainframe system.
- **Real-Time or Batch.** You should consider the type of integration you need: real-time or batch mode. For example, you might want to publish new opportunities created in Siebel applications to an external system as soon as the opportunities occur. Or you may want to exchange information in batch mode when you want to aggregate product catalog information from your suppliers.

Once you have designed your integration needs for each business process flow, you can apply the different components of Siebel EAI to implement a solution. Figure 8 illustrates the different types of applications that Siebel Business Applications can integrate with.

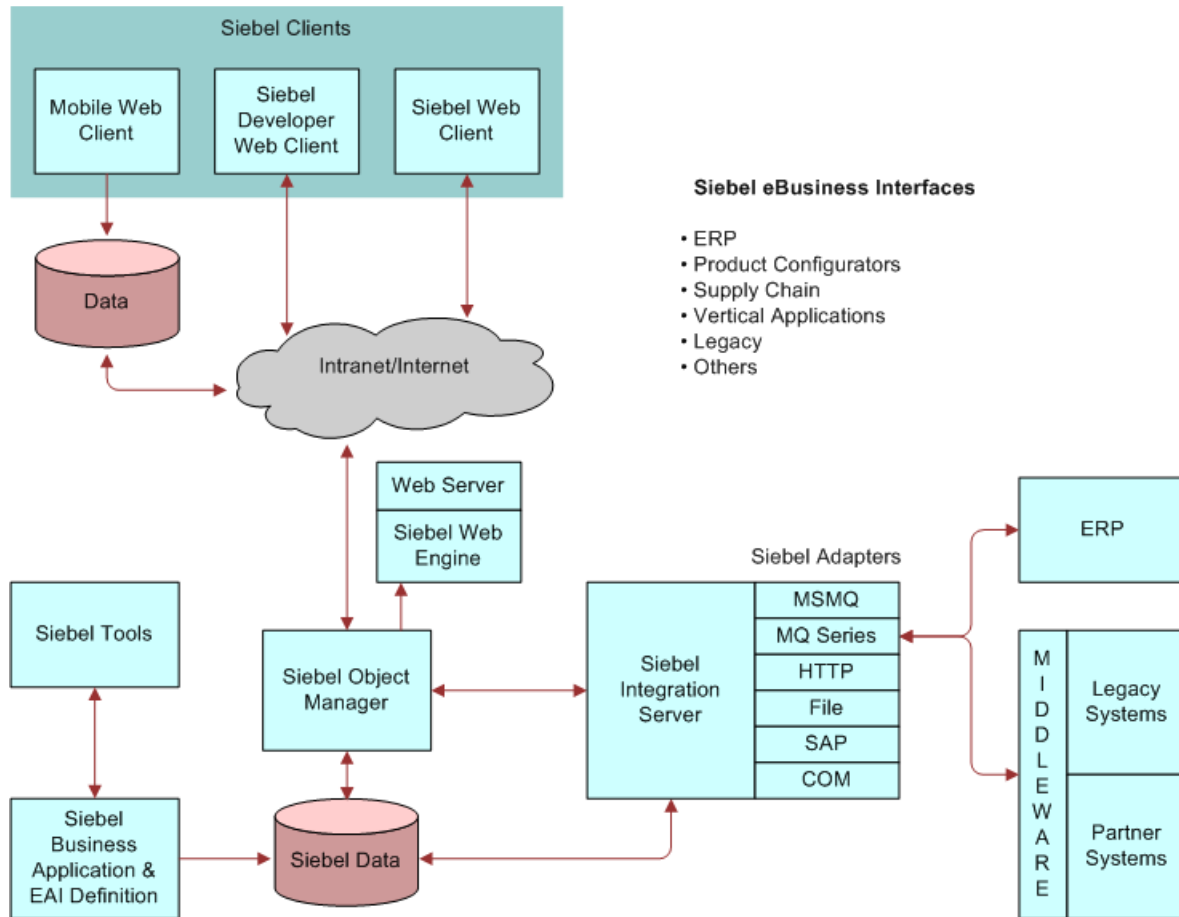


Figure 8. Siebel Business Interfaces

About Usage Models

When looking at your requirements, you should understand how you intend to use Siebel EAI's integration capabilities. You need to determine how the integration will behave and under what circumstances. In other words, in what way are you planning to use this integration? This is referred to as the *usage model*.

As shown in Figure 9, there are two basic usage models, real-time and batch. Batch mode has a single mode, and real-time has two modes, either tightly coupled or loosely coupled. To help you determine the appropriate usage model, ask yourself the following questions:

- How do I plan to integrate Siebel applications with other applications?
- Do I want to integrate in real time or in batch mode?

- If I want to integrate in real time, do I want the solution to be tightly coupled or loosely coupled?

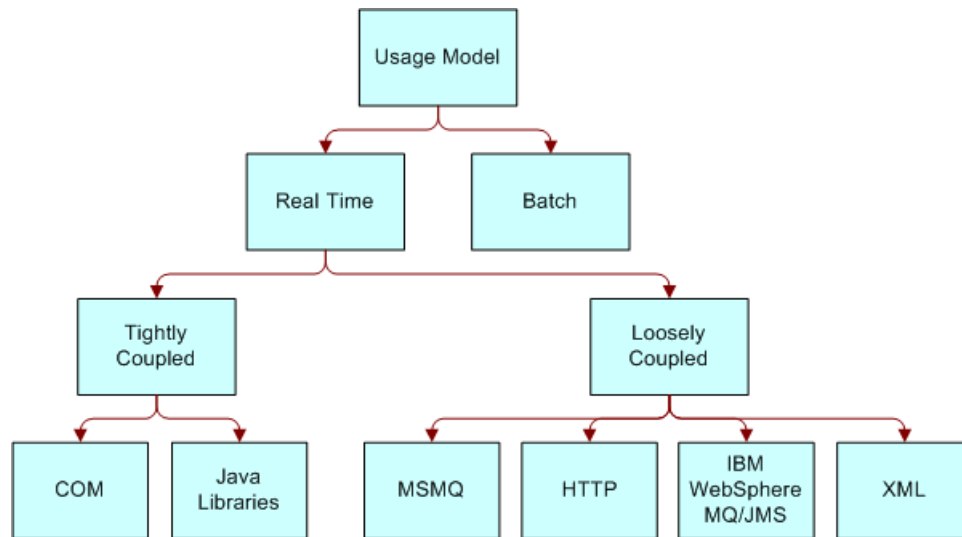


Figure 9. Integration Usage Model

Using the Real-Time Model in Your Integration

With a real-time model, you may choose to deploy Siebel applications as a client or as a service. As a client, Siebel applications access another program or initiate a transaction on an external system. As a service, Siebel applications are accessed by an external program to query, insert, update, or delete data without any initiation from the Siebel application.

Typical scenarios involving Siebel application as a client include:

- Replicating data such as maintaining the SAP order number in the Siebel application, so users can relate Siebel orders to SAP orders.
- Accessing a mainframe system to retrieve customer account information in real time and display it within the Siebel application's interface.
- Sending sales opportunities to trading partners over your corporate firewall without compromising system security.
- Accessing an external system's function to check inventory and then displaying the results in a Siebel application.

Typical scenarios involving Siebel applications as a service include:

- An external Web application accessing a Siebel application to retrieve contact information.
- Call center software extracting service request information from a Siebel application.

Using the Batch Model in Your Integration

Batch processing typically is used for periodic uploads or downloads of large amounts of information into the Siebel Database. Typical batch scenarios involving Siebel applications include:

- Uploading a batch of product catalog and item information into a Siebel application from an external system.
- Downloading a batch of opportunities from Siebel applications to another database.

About Exchanging Data with Siebel Applications

When your business problem is accessing other applications from Siebel applications or accessing Siebel applications from other applications, you need to ask: Exactly what kind of data exchange do I require?

- **Data Sharing.** You may need to share data that exists outside of the Siebel Business Application. For example, you may have several legacy mainframe systems that you need to access with Siebel Call Center in order to provide customers with information on their accounts. You might not want to replicate this information in Siebel Business Applications, but may want to dynamically share this mainframe data as needed and display it within a Siebel application. For data sharing, use Siebel's Virtual Business Components (VBCs).
- **Replication.** Replication of data is often useful. For example, you might want to integrate data in SAP with data in Siebel applications, keeping order status information in both systems. For mass replication of the order status information in Siebel applications, use Siebel Enterprise Integration Manager (EIM).
- **Presentation Layer (UI) Integration.** Integration at the presentation layer is useful in cases where you only need to look at the other application's user interface (UI). The data itself does not need to be retrieved to the Siebel applications. In these cases, you would capture just the screen from the external system using ActiveX Data Controls (ADCs) and represent it natively within a Siebel application. This is the least expensive form of integrating two applications, but it is the least flexible as well.

These Siebel solutions can be either real-time or batch. Figure 10 shows the various access mechanism options and where they fit into the integration process.

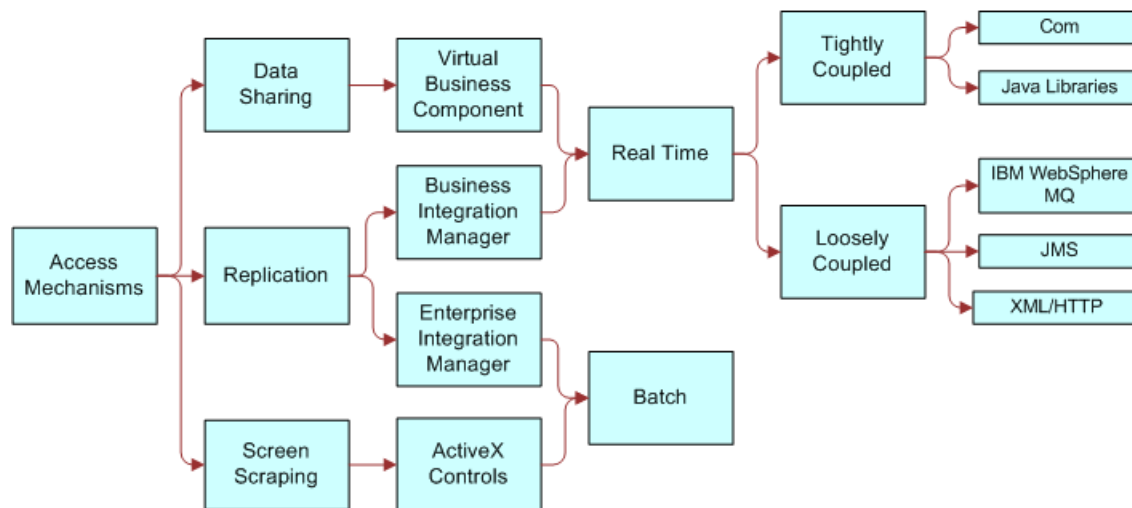


Figure 10. Siebel Solutions

Table 2 reflects the Siebel integration tools that can be used in your integration strategy.

Table 2. Siebel Integration Tools

Integration Strategy	Description	Siebel Integration Tools
Data Replication: Real-time	In real time, synchronize the data in one system with the data in another system.	Integration Workflow Process EAI Connector EAI Dispatch Service Object Interface
Data Replication: Batch	At certain times, synchronize the data in one system with the data in another system.	Enterprise Integration Manager Integration Workflow Process Object Interface
Data Sharing	From System A, access and change System B data. Do not store the data in System A.	Virtual Business Component Object Interface
Screen Scraping	From Application A, access screens from Application B.	ActiveX Controls Object Interface

4

Siebel EAI Architecture

This chapter explains the Siebel EAI architecture. The following topics are included:

- [“About the Siebel EAI Architecture” on page 34](#)
- [“About Integration Requirements and Siebel EAI” on page 35](#)
- [“About Business Process Coordination Using Workflows” on page 36](#)
- [“About Transport Mechanisms” on page 37](#)
- [“About High-Volume Data Exchange” on page 37](#)

About the Siebel EAI Architecture

As it is shown in [Figure 11](#), Siebel EAI architecture consists of the EAI Siebel Adapter, the EAI Data Mapping Engine, and multiple transports that can facilitate data integration with different external systems.

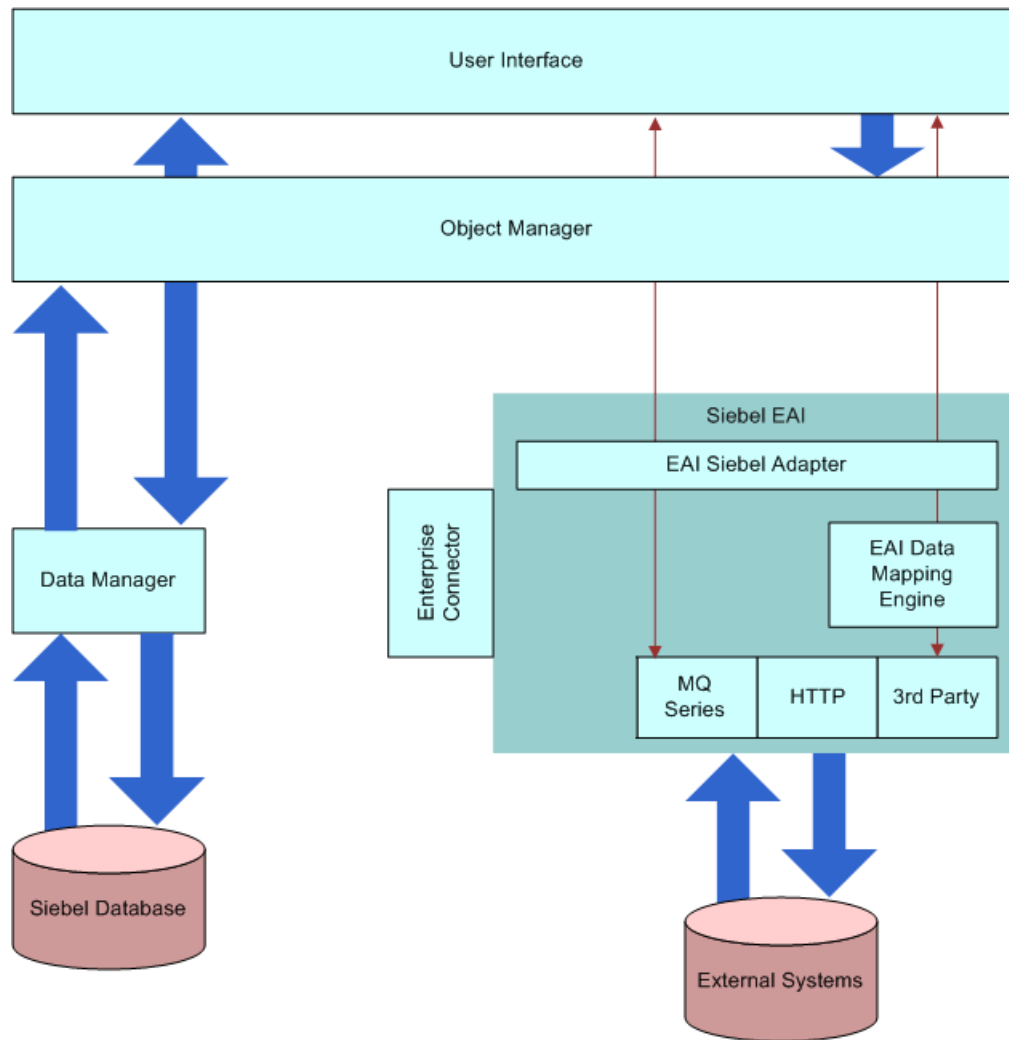


Figure 11. Siebel EAI Architecture

The abstraction of data management functions to an object layer allows for a declarative, as opposed to a procedural model. This improves and simplifies configuration tasks while lowering maintenance costs.

About Integration Requirements and Siebel EAI

Table 3 maps real-world data and application integration requirements with their corresponding Siebel EAI features.

Table 3. Integration Requirements and How They Relate to Siebel EAI Features

Integration Requirements	Siebel EAI Features or Components	Notes
Metadata Definition XML Metadata External application metadata (from SAP applications, Oracle applications, and so on)	<ul style="list-style-type: none"> ■ Integration objects based on Siebel business objects ■ Integration objects based on external metadata ■ Integration Object Wizard 	In any integration project, you need a way to supply data about data, called metadata. Siebel applications provide XML as the common format for representing external application data. In addition, Siebel applications provide an object type in Siebel Tools called the Integration Object, as well as an Integration Object Wizard, which automates the task of creating integration objects.
Data Transformation Declarative data transformation mapping High-performance transformation engine	<ul style="list-style-type: none"> ■ Siebel Data Mapper 	Siebel EAI allows you to define what type of data to be exchanged, and allows you to export an XSD or a DTD for a Siebel integration object.
Business Process Coordination	<ul style="list-style-type: none"> ■ Siebel Workflow Process Designer ■ Workflow Process Manager 	See About Business Process Coordination Using Workflows on page 36 .
Transport Mechanisms (for interfacing with other technologies)	Transport Adapters: <ul style="list-style-type: none"> ■ IBM MQSeries ■ Microsoft MSMQ ■ HTTP ■ Java Message Service 	See About Transport Mechanisms on page 37 .

Table 3. Integration Requirements and How They Relate to Siebel EAI Features

Integration Requirements	Siebel EAI Features or Components	Notes
High Volume Data Exchange	<ul style="list-style-type: none"> ■ EIM (Enterprise Integration Manager) 	See About High-Volume Data Exchange on page 37.
Standards-based Integration Mechanisms	<ul style="list-style-type: none"> ■ XML as a format for data exchange ■ COM and Java-based programmatic interfaces ■ HTTP as a transport protocol 	

About Business Process Coordination Using Workflows

Business process coordination means that every system in an enterprise sees the same up-to-date information. For example, when you need to change a customer address, the change must be reflected in every system that stores the address. The Siebel Workflow Process Designer has been developed to help you define your data integration business process flows.

The application of Siebel's Workflow architecture within Siebel EAI is an important component in the design and deployment of any effective data integration solution. [Figure 12](#) shows how workflows respond to event triggers or time-based triggers to call transports, scripts, or object interfaces to communicate with external applications.

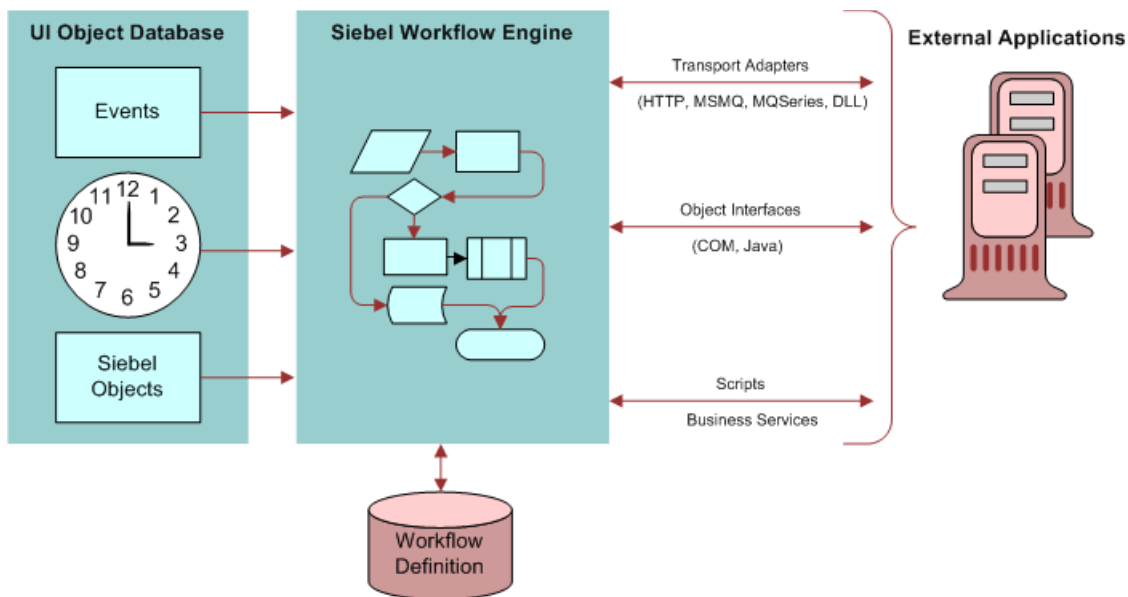


Figure 12. Siebel Workflow Architecture for Siebel EAI

For detailed information on Siebel workflows, see *Siebel Business Process Framework: Workflow Guide*.

About Transport Mechanisms

Many companies have standardized on transport solutions, such as MQSeries from IBM or MSMQ from Microsoft. These transports allow the transportation of messages between systems. Siebel Systems conforms to this standard by allowing you to plug Siebel applications into these environments through the Siebel EAI set of adapters, which include the MQSeries adapter, the MSMQ adapter, and the HTTP adapter.

Transport adapters developed by Siebel Systems and Siebel-certified partners allow diverse applications to communicate with Siebel applications across virtually any operating system, network, and database. Figure 13 shows the various available transports to and from Siebel applications as a new order is processed. Any one or any combination of these transports could be used in an integration solution, depending upon existing applications and other factors. These transports can also be configured as a Siebel Server Receiver to receive messages, as for example, the MQSeries Server Receiver. For more details on using these transports, see *Transports and Interfaces: Siebel Enterprise Application Integration*.

About High-Volume Data Exchange

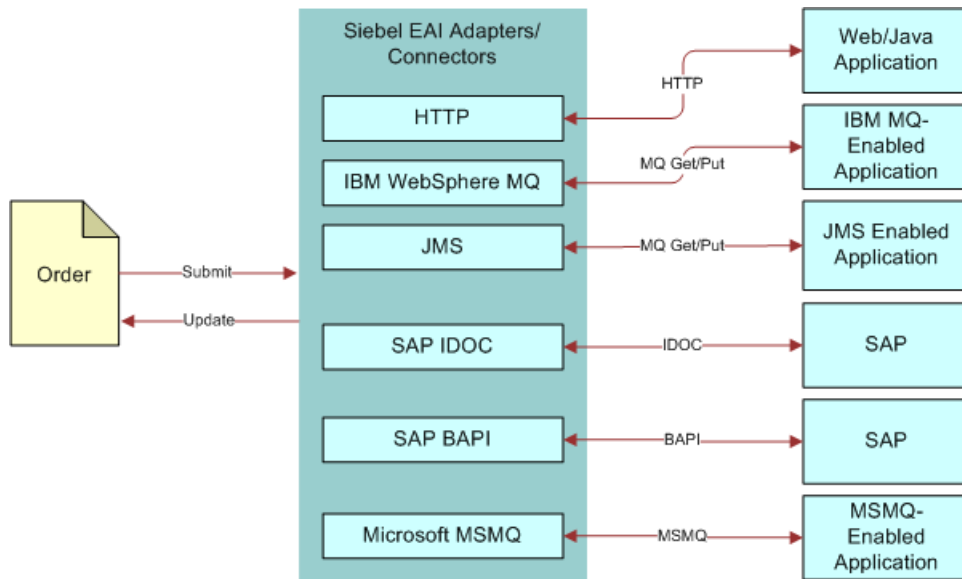


Figure 13. Order Entry Cycle Showing Transport Adapters

Most organizations require a mechanism to schedule high-volume integrations at a specific time, typically at night, to avoid network slowdowns or interference with other applications. Such integrations typically involve the updating of hundreds of millions of customer records. This type of activity requires a batch mechanism capable of processing such large amounts of data in a timely manner. Siebel Enterprise Integration Manager (EIM) allows you to schedule high-volume data exchanges between Siebel applications and other applications whenever needed, at whatever frequency required (twice a day, nightly, weekly, and so on).

About the Enterprise Integration Manager

EIM acts as a high-volume batch interface. With EIM, you can import and export data from the Siebel Database to other corporate databases. Figure 14 shows the EIM batch data process. EIM includes database profiling that improves performance by supporting index specialization in Relational Database Management Systems (RDBMS).

For more information on EIM, see *Siebel Enterprise Integration Manager Administration Guide*.

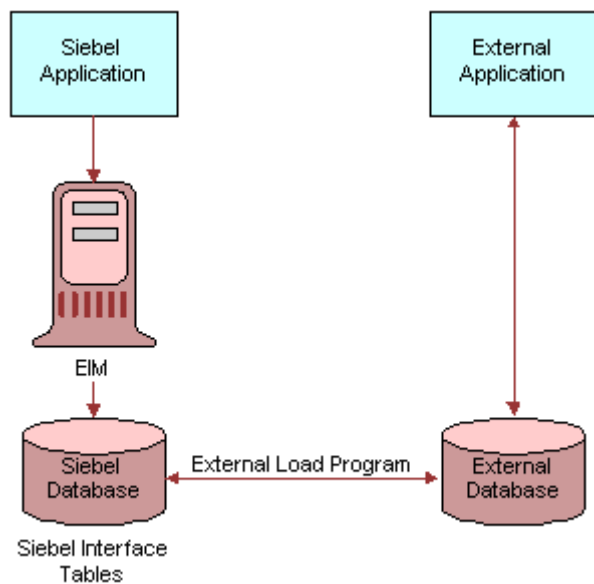


Figure 14. Siebel Enterprise Integration Manager at Work for EAI

5

Custom EAI Business Scenarios

This chapter discusses custom business scenarios for using Siebel EAI for data integration. It includes the following topics:

- [“Determining Your Integration Needs”](#)
- [“About Common Integration Scenarios”](#)
- [“About Data Access and Replication” on page 41](#)
- [“About Data Across the Enterprise” on page 43](#)
- [“About Performing Data Transformation” on page 44](#)

Determining Your Integration Needs

Sometimes the best way to determine your integration needs is to study examples of similar problems faced by other organizations. This chapter presents three scenario categories, each with a number of permutations. The categories are:

- [About Data Access and Replication on page 41](#)
- [About Data Across the Enterprise on page 43](#)
- [About Performing Data Transformation on page 44](#)

The first step in getting started on an integration project is to identify your integration needs. Siebel Business Applications provide solutions for a variety of business needs. They play a key role in the application environment, in which you might use other applications such as back-office solutions for order fulfillment, manufacturing, or accounting; e-commerce applications such as Web storefronts; help desk applications; and mainframe-based applications that store millions of customer transactions.

As customer applications, Siebel Business Applications need to interact with these external systems to provide an integrated view of your information. Siebel EAI provides the tools as well as prebuilt components that you can use to create this integration. There are several ways in which you can approach integration, depending on your needs:

- You can connect Siebel applications and your back office or e-commerce system, using prebuilt EAI connectors.
- You can integrate Siebel applications with middleware solutions.
- You can integrate Siebel applications with external applications, using a third-party EAI product.

About Common Integration Scenarios

The common integration scenarios listed here apply to many of the integration requirements you face. Siebel EAI allows you to accomplish integrations within the scope of these scenarios but is not limited to the scenarios described here.

Outbound Message to a File

In this scenario, you create a Siebel workflow process to create a unique file every time a support representative closes a service request. The file is then sent to a central directory on a network server, where at midnight it is bundled automatically into an Adobe PDF file along with other closed service requests for that day.

Round-Trip Message from a File to Siebel Applications

In this scenario, you want service requests entered from your Web site to be entered as service requests in Siebel Service. You also want to pull an automatically generated service request number from Siebel Service, add it to the service request, and generate a file that is attached to a confirmation email message to the service requestor's email address.

Round-Trip Message from an External Application

In this scenario, whenever your back-office Enterprise Resource Planning (ERP) system recognizes a reorder point on a popular inventory item, it sends a message to your Siebel application to notify salespeople that a popular item is about to be reordered. If the salespeople want to order from the remaining supply at the current price, they can respond to the message.

Round-Trip Message to an External Application with a Different Code Page

In this scenario, your salespeople want to be able to send an order from Siebel applications to your back-office ERP system that is using Japanese 932, while the code page used by the Siebel application is UTF-8. Your salespeople want to receive a confirmation with an order number generated by the ERP system.

Inbound Message from an External Application

In this scenario, whenever product information in your SAP R/3 system is updated, you want the system to send an updated IDOC to your Siebel system and have the product information updated in the Siebel Product business object.

Outbound Message to an External Application

In this scenario, you want to update account information in your SAP R/3 IDOC from Siebel Business Applications.

About Data Access and Replication

Often, it is not sufficient to just bring everything to the same screen. Many business processes require access to particular data in order to accomplish a given task.

This section describes four examples:

- “Sending Contact Information to an External System” on page 41
- “Providing Access to a Company Catalog” on page 42
- “Accessing Data Using Siebel Java Beans” on page 42
- “Accessing Customer History Information Without Replication” on page 42

Sending Contact Information to an External System

Consider a scenario in which you want to send changes to contact information in the Siebel Database to an external mainframe system through the IBM WebSphere MQ mechanism. Your first step is to define and configure the message that needs to be exchanged. In the Siebel Database, the Contact business object contains several business components, and each business component has several fields. You do not want to send all of this information to the external system. You want to choose only a subset of this information.

Use Siebel Tools to define the integration object for the contact message you want to exchange. Siebel Tools provides an Integration Object Builder that walks you through the process of defining your integration object. Once you have defined the message, define what to do with this message. The contact message needs to be sent to the external system using EAI MQSeries Server Transport. You can use the Siebel Workflow Process Manager to model this integration message flow and to send the message. An integration message flow is a sequence of business services that are connected. [Table 4](#) details the requirements for exporting contact information from Siebel applications to a mainframe system.

Table 4. Customer Data Export Integration Requirements

Business Requirement	Technological Requirement	Siebel’s EAI Solution
To access the Siebel Database	Business service to query the integration object	EAI Siebel Adapter
To convert an internal Siebel message format into XML	Siebel message-to-XML conversion technology	EAI XML Converter
To place the converted XML document in the destination queue	IBM MQSeries compatibility	EAI MQSeries Server Transport

Providing Access to a Company Catalog

In this scenario, you need to provide your salespeople access to a product catalog while they are on the road. So, you need to import the catalog information into Siebel applications so that Siebel Remote can take care of the mobile replication. Siebel Enterprise Integration Manager (EIM) imports this information. [Table 5](#) details the requirement for importing a company's catalog for mobile replication.

Table 5. Company Catalog Replication Integration Requirement

Business Requirement	Technological Requirement	Siebel's EAI Solution
To import the company's product catalog	High-volume batch replications	Siebel Enterprise Integration Manager (EIM)

Accessing Data Using Siebel Java Beans

In this scenario, you have a customer service Web application written using JavaServer Pages (JSPs). To pass this data into Siebel applications so that account records can be updated, you invoke the Siebel Object Interface, Siebel Java Data Beans. Now, your Customer Service representatives can use the information from the Web application JSPs while serving the customer.

[Table 6](#) details the requirement for updating the Siebel Database using JSP technology.

Table 6. Siebel Update Integration Requirement

Business Requirement	Technological Requirement	Siebel's EAI Solution
To update Siebel account data from a JSP-enabled application	Programmatic interaction	Siebel J2EE Resource Adapter & Application Services

Accessing Customer History Information Without Replication

In this scenario, your company is part of the financial industry. Financial service institutions are typically cautious in the way they treat customer data, so they want to provide access to data while replicating as few times as possible, if at all. Your customer service agents need to know the transaction history of your customers in order to provide the best service. This information is typically stored in Customer Information files (CIFs) on mainframe computers.

Table 7 details the requirement for integrating customer transaction history information into Siebel applications.

Table 7. Customer Data Access Integration Requirement

Business Requirement	Technological Requirement	Siebel's EAI Solution
To provide access to the Customer Information Files (CIFs) stored on a mainframe and display this information in the customer portal	Access to non-Siebel data without replication	Siebel Virtual Business Components (VBCs)

Siebel Virtual Business Components (VBCs) are built to handle this business need. VBCs abstract external data and present it to Siebel applications as part of the Siebel data model. Siebel business logic and user interface components process Virtual Business Components in the same way that they process Siebel application's regular business components.

About Data Across the Enterprise

Business processes often need to cross departmental or even enterprise boundaries. Because different organizations may be using different applications, you need a way to make sure one application can hand off the execution of a business process to the next application. This section describes two examples: ["B2B Integration and Siebel EAI"](#) and ["Integrating Legacy Data"](#) on page 44.

B2B Integration and Siebel EAI

Business-to-business integration allows partner companies to work together to satisfy customer demands. Since B2B is fundamentally about carrying out transactions over the Internet, the technology must work over the Internet infrastructure. To meet this demand, Siebel EAI provides the EAI HTTP adapter and the Siebel Workflow Process Manager (which is used to model and execute business processes) to handle the task.

Table 8 details the requirement and the solution for a successful B2B integration.

Table 8. B2B Integration Requirement

Business Requirement	Technological Requirement	Siebel's EAI Solution
To create a business-to-business integration to send an order directly to a partner over the Internet for fulfillment	XML messaging over HTTP	<ul style="list-style-type: none"> ■ Siebel Workflow Process Manager ■ EAI HTTP Transport adapter

Integrating Legacy Data

Interfacing with a custom legacy application presents similar challenges to those faced in the B2B scenario. On the Internet, HTTP is the standard protocol for exchanging data. In a mainframe environment, IBM MQSeries is the most common application messaging transport. Therefore, you can replace the HTTP protocol in this solution with MQSeries and still carry out the transaction.

Table 9 details the requirement and the solution for a successful integration of legacy data existing on a mainframe.

Table 9. Legacy Data Integration Requirement

Business Requirement	Technological Requirement	Siebel's EAI Solution
To create an order against a mainframe system	Cross-application business process management	<ul style="list-style-type: none"> ■ Siebel Workflow Process Manager ■ EAI MQSeries Transport

About Performing Data Transformation

When performing data transformation within the Siebel environment, you use two types of integration objects:

- Internal integration objects mirror the data structure of a Siebel business object and contain a subset of business components and fields.
- External integration objects mirror the data structure of an external system.

You define integration objects of both types in Siebel Tools.

The EAI Siebel Adapter performs the same function whether the data transformation is performed inside or outside the Siebel environment. The XML Converter and the transports also perform the same function, whether they are used within the Siebel environment or outside the Siebel environment.

The only difference is that when used outside the Siebel environment, both of these business services perform operations on external integration object instances, as opposed to performing operations on internal integration object instances when used within the Siebel environment.

You define data mapping and data transformation rules using business services. In the prebuilt external adapters, Siebel EAI has provided many mappings and transformation rules.

NOTE: Business services provide a flexible scripting environment for mapping data from the Siebel data structure to an external data structure. Within a business service, Siebel applications provide a set of methods that allow you to perform operations on integration object instances.

Creating Orders to Pass to an External Application

In this scenario, an order is created in Siebel Business Applications and is then passed to an external application. The external application returns the order status and an order number. [Table 10](#) details the requirements for data transformation integration.

Table 10. Data Transformation Integration Requirements

Business Requirement	Technological Requirement	Siebel's EAI Solution
To create a new internal Integration Object	Integration object tool or wizard	Siebel Integration Object Wizard
To create a new external Integration Object	Integration object tool or wizard	Siebel Integration Object Wizard
To create a new business flow	Workflow technology	Siebel Workflow Process Designer

For this task, identify the Siebel business object that you would want to update. For that business object, identify the components and fields that you expect to update. Next, identify the object in the external system with which you want to integrate. Then, determine the mapping between the Siebel business object and the external object.

In Siebel Tools, create a new internal integration object using the Integration Object Wizard. Select only the components and fields that you want to update. Then, in Siebel Tools, create a new external integration object. Then, either in Siebel Tools or in the Siebel client, define the data mapping and data transformation using business services.

In Siebel Tools, define an integration workflow process and add any exception handling to it. Test the workflow process using the process simulator. Finally, add a mechanism for invoking it from Oracle's Siebel applications.

Index

A

adapters

See also Siebel EAI adapters
defined 9

list for Siebel applications, finding 24

Application Services Interfaces (ASIs)

about 15

inbound (diagram) 16

outbound (diagram) 17

architecture

layer diagram 34

Siebel Workflow architecture 36

B

batch integration

high volume transaction processing 38

importing and exporting data 38

mode, about integration planning 28

processing usage model, about and scenarios 31

business processes

coordination using workflows, about and diagram 36

Siebel EAI connectors, about 10

business services

See Siebel EAI adapters 9

C

catalog, data replication scenario 42

client-side integration, about 25

COM protocol, about 20

connectors

Siebel Connector for SAP R/3 10

Siebel connectors, defined 10

third-party, about 10

conversion, double conversion and corrupting data 21

cross-application process integration, about 25

Customer Information Files (CIFs), about information stored in 42

D

data exchange

accessing mechanisms, about 31

business-to-business scenario 43

legacy data integration, scenario 44

presentation layer 31

data mapping

See data transformation

data replication

company catalog replication integration requirement scenario 42

data exchange requirement, about 31

data sharing

business-to-business scenario 43

data exchange requirement, about 31

integration planning, about 28

integration strategy, about 25

legacy data integration, scenario 44

data transformation

data mapping integration, about 25

integration objects, scenario 44

integration, planning 28

order integration, scenario 45

DDK

See Web UI Dynamic Developer's Kit (DDK) 18

declarative model, about 34

design decisions

application integration planning 28

Document Types Definitions (DTDs), metadata XML definition 9

double conversion, about corrupting data 21

E

EAI

See Siebel EAI 7

EAI Siebel Adapter business service, about 9

EAI UI Data Adapter business service, about 9

exporting data

contact information scenario 41

export integration requirements (table) 41

external business components (EBCs), about 11

H

high volume transactions, about the EIM Manager batch interface 38

HTTP transport

transport, about 10

I**IBM MQSeries adapter**

See MQSeries

inbound Unicode scenario, about 21**integration**

- data presentation level 31
- design decisions 28
- requirements and Siebel EAI (table) 35
- scenarios 40

integration objects

defining, scenario 41

integration scenarios

- business-to-business integration 43
- customer history information 42
- data transformation 44
- exporting contact information 41
- inbound message 40
- J2EE Server Pages 42
- legacy data integration 44
- mainframe using MQSeries transport 41
- mobile data replication 42
- order integration 45
- outbound message 40
- outbound message to file 40
- round trip message 40

integration servers, integration strategy 24**integration strategies**

- application administration, about 25
- cross-application process integration, about 25
- data mapping, about 25
- data sharing, about 25
- integration servers, about and diagram 24
- legacy encapsulation 22
- master data partitioning, about 25
- peer-to-peer 23
- upgradeable integration, about 25

integration, planning

- access mechanism options (diagram) 32
- access mechanisms, about and requirements 31
- batch processing usage model, about and scenarios 31
- real-time usage model, about and scenarios 30
- usage models, types of 29

J**Java Bean protocol, about 20****Java/J2EE**

accessing from Siebel applications 11

object interfaces, list of 13

Server Pages, scenario 42

Siebel applications, accessing from 12

L**legacy encapsulation, about and diagram 22****M****master data partitioning, about 25****mobile replication, scenario 42****MQSeries**

integration scenario 41

transport adapters 10

O**Object Interface scenario 42****outbound Unicode scenario, about 21****P****peer-to-peer encapsulation, about and diagram 23****presentation layer integration, about 25****procedural model, about 34****protocols, role of transport adapters 10****publish-and-subscribe technology 23****R****real-time**

mode, about integration planning 28

usage model, about and scenarios 30

replication of data, about and data exchange requirement 31**S****SAP R/3, Siebel connector for 10****screen scraping, integration scenario 31****Siebel adapters**

See adapters

Siebel Application Services Interface (ASIs).

See Application Services Interface (ASIs)

Siebel applications

accessing mechanism options (diagram) 32

accessing mechanisms, about and

requirements 31

Siebel connectors

See connectors

Siebel EAI

about 7

adapters and connectors 9

- architecture diagram 34
 - declarative data mapper, about using 14
 - external business components 11
 - inbound and outbound Unicode scenarios 21
 - Java/J2EE components, accessing 11
 - message-based integration, about and diagram 13
 - Siebel applications, accessing from Java/J2EE component 12
 - Siebel integration object, about and mapped using Siebel Repository 14
 - Transcode business service, outbound and inbound 21
 - Unicode conversion, about and diagram 20
 - virtual business components 10
 - workflow control, about using for integration between applications 15
 - XML and Siebel EAI, about and support of 9
 - Siebel EAI adapters**
 - EAI Siebel Adapter business service 9
 - EAI UI Data Adapter business service 9
 - transport adapters 10
 - Siebel Enterprise Integration Manager (EIM)**
 - batch interface, about acting as 38
 - software bus strategy, about adopting 23**
- T**
- third-party connectors, about 10**
 - transactions, high volume 38**
 - Transcode business service, outbound and inbound communication 21**
 - transport adapters**
 - about and diagram showing available transports 37
 - business services within Siebel EAI 10
 - MQSeries 10
- U**
- Unicode support**
 - inbound and outbound Unicode scenarios 21
 - Transcode business service, outbound and inbound 21
 - Unicode conversion, about and diagram 20
 - upgradeable integration, about 25**
 - usage models**
 - about and types of 29
 - batch processing usage model, about and scenarios 31
 - real-time usage model, about and scenarios 30
- V**
- virtual business components (VBCs), about 10**
- W**
- Web client, client-side integration, about 25**
 - Web Service, about and communication protocols based on 17**
 - Web UI Dynamic Developer's Kit (DDK)**
 - about and key features 18
 - workflows**
 - business process coordination and workflow architecture 36
- X**
- XML data exchange, role in 9**

