Siebel CRM

Overview: Siebel Enterprise Application Integration

July 2026

Siebel CRM
Overview: Siebel Enterprise Application Integration

July 2026

F87430-03

Copyright © 2026, Oracle and/or its affiliates.

Author: Sukanya Mishra

Contents

Get Help

1 What's New in This Release 1

What's New in Overview: Siebel Enterprise Application Integration, Siebel CRM 20.1 Update 1
What's New in Overview: Siebel Enterprise Application Integration, Siebel CRM 19.1 Update 1

2 Overview of Siebel EAI 3

Overview of Siebel EAI 3
About Siebel EAI 3
About Siebel EAI Components 4
About Interfaces for Siebel Application Services 9
About Web Services 11
About Siebel EAI and Unicode Support 12
About Siebel EAI and Legacy Systems 15
About Siebel EAI Integration Strategies 16

3 Integration with Siebel EAI 19

Integration with Siebel EAI 19
Overview of Integration with Siebel EAI 19
Integration Options with Siebel EAI 19
Design for an Application Integration Project 20
Real-Time and Batch Mode Usage Models 21
Data Exchange with a Siebel Application 23
Summary of Integration Options with Siebel EAI 23

4 Architecture for Siebel EAI 25

Architecture for Siebel EAI 25
Overview of Architecture for Siebel EAI 25

5 Scenarios for Using Siebel EAI 31

Scenarios for Using Siebel EAI 31

Scenarios for Common Integrations 31
Scenarios for Data that is Accessed, Exported or Replicated 32
Scenarios for Data that Crosses a Boundary 34
Scenarios for Data that is Transformed 35

1 What's New in This Release

What’s New in Overview: Siebel Enterprise Application Integration, Siebel CRM 20.1 Update

No new features have been added to this guide for this release. This guide has been updated to reflect only product name changes.

What’s New in Overview: Siebel Enterprise Application Integration, Siebel CRM 19.1 Update

No new features have been added to this guide for this release. This guide has been updated to reflect only product name changes.

2 Overview of Siebel EAI

Overview of Siebel EAI

This chapter describes an overview of Oracle's Siebel Enterprise Application Integration (Siebel EAI). It includes the following topics:

- *About Siebel EAI*
- *About Siebel EAI Components*
- *About Interfaces for Siebel Application Services*
- *About Web Services*
- *About Siebel EAI and Unicode Support*
- *About Siebel EAI and Legacy Systems*
- *About Siebel EAI Integration Strategies*

About Siebel EAI

In order to drive revenue growth, increase productivity, and gain visibility to business performance, an organization must integrate business processes across a large number of disparate applications. Because business is accelerating, an organization must integrate and streamline business processes in order to achieve the agility that is required to respond to changing customer and market demands. This integration imperative is even more pressing because supply and demand chains are increasingly interdependent, requiring an organization to integrate an extended enterprise to make sure the viability of the entire partner ecosystem.

To meet these requirements, Oracle continues to enhance the integration tool set that exists within Siebel applications.

Siebel Enterprise Application Integration

Siebel Enterprise Application Integration (Siebel EAI) is the set of products on the Siebel Business Platform that includes tools, technologies, and prebuilt functional integrations that facilitate application integration. Some of the features provided by Siebel EAI include:

- Provides components to integrate Siebel Business Applications with external applications and technologies within your company
- Works with third-party solutions, such as solutions from IBM, TIBCO, WebMethods, and so forth
- Provides bidirectional, real time and batch solutions to integrate Siebel applications with other applications
- Provides a set of interfaces that interact with each other and with other components within a Siebel application. Some of the features that these interfaces provide include:
- Provides a flexible, service based architecture that is built on top of configurable messages using XML and other formats
- Provides compatibility with IBM MQSeries, Microsoft MSMQ, Sun Microsystems Java and J2EE, XML, HTTP, and other standards.

- Provides access of internal Siebel Objects to an external application
- Takes advantage of prebuilt adapters and enterprise connectors
- Provides compatibility with third-party adapters and connectors
- Provides data transformation
- Integrates external data through virtual business components and external business components
- Provides a graphical business process designer, programmatic interfaces, and a high volume batch interface

About Siebel EAI Components

This topic describes components in Siebel EAI. It includes the following topics:

- *Siebel EAI and XML*
- *Siebel EAI Adapters and Connectors*
- *Siebel EAI and Business Components*
- *Siebel EAI with Java and J2EE*
- *Siebel EAI and Object Interfaces*
- *Siebel EAI and Message Based Integration*
- *Siebel EAI and Data Transformation*
- *Siebel EAI and Workflow Process Control*

A legacy application, such as purchasing or accounting, can contain critical information that must coexist and integrate with a Siebel application. A business-to-business interaction requires that customer, order, and account information be shared with partners across a firewall. Siebel EAI provides components that can be used individually or together with a toolkit from an EAI vendor, including XML support in Siebel EAI adapters and connectors, a virtual business component, or Java Beans.

Siebel EAI and XML

XML is increasingly the format of choice for data exchange between an application and a partner organization. The metadata definition for XML is in the form of XML Schema Definitions (XSD) or Document Types Definitions (DTD). Because Siebel Integration objects can be represented as XSD, another application can capture metadata from a Siebel application and understand how to format an XML document that must be sent to a Siebel application. Data in a Siebel application 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 an outgoing or incoming message. Siebel EAI can be configured to use an XSD or DTD that is externally specified to define external integration objects. Siebel Tools includes an Integration Object Wizard that can be used to import and process an external XSD or DTD and to generate these external integration objects that represent the XSD or the DTD.

A transport adapter, whether built by Oracle or built by a partner, can use this business service to dynamically look up the content in an XML document or in a document that uses another format, then route the message based on content from the document. The message can be routed to a workflow process or to another business service within a Siebel application.

Siebel EAI Adapters and Connectors

Siebel EAI provides adapters and connectors to help create an integration between a Siebel application and an external application. Other connectors are also available through partners for Siebel Business Applications. For each supported business process, the connectors include data mapping between the exchanged entities, and sequencing and error handling.

Siebel EAI Adapter

An adapter is a Siebel EAI component that provides a low level interface to allow one application to talk to another application. The three types of adapters available in Siebel EAI include: Siebel EAI adapters

- EAI Siebel Adapter. The EAI Siebel Adapter business service is specifically designed to interact with the Siebel business object layer. It is a business service in a Siebel application that can use an XML document as input, or can use a property set that conforms to an integration object definition in the Siebel system. EAI Siebel Adapter can then query, insert, update, delete, or synchronize data with the Siebel business object layer.
- EAI UI Data Adapter. The EAI UI Data Adapter business service allows a Web application that is custom built to access business components and business objects. This adapter publishes strongly typed APIs to allow data centric object manager operations, such as insert, delete, update, query or execute. The EAI UI Data Adapter supports the query by example and the predefined query.
- Transport Adapter. A transport adapter is a business service within Siebel EAI that is designed to communicate with other protocols and technologies that exist outside of a Siebel application. A transport adapter transports data from the Siebel system to another system:
  - The HTTP transport in Siebel EAI allows a Siebel application to communicate over the HTTP protocol.
  - The MQSeries adapter allows a Siebel application to communicate with the IBM MQSeries messaging system.
  - The MSMQ adapter allows a Siebel application 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 Connector

A Siebel EAI connector provides low level connectivity to another back office application, such as SAP or Oracle. It also includes the business processes that are used to connect to the external application. For example, the Siebel Connector for SAP R/3 provides connectivity using BAPI and IDOC transport adapters, and predefined business processes. Using the connectors, you can exchange customer, order, and product information between a Siebel application and another Oracle application or SAP.

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

Third-Party Connector

Your organization might be required to integrate multiple applications with one another, such as Siebel applications, SAP, Oracle, I2, and so forth. You might require an EAI solution from an EAI vendor. These vendors work closely with Oracle to develop adapters for a Siebel application that use Siebel EAI. Oracle uses a program that makes sure that these adapters work according to Oracle standards.

Siebel EAI and Business Components

This topic describes Siebel EAI usage with business components.

Siebel EAI and the Virtual Business Component

A *Virtual Business Component* (VBC) is a mechanism in Siebel EAI by which data from an external system is viewed in a Siebel application without having to replicate that data within the Siebel database.

For example, assume an end user accesses detailed credit card information from a mainframe system for a particular customer account, which is viewed in the Siebel application interface. In this case, the basic customer account data, such as account number, name, and so forth, is stored in a Siebel application while the detailed transaction information on the credit card account for that customer is stored in an external database. The detailed transaction information for a particular account is retrieved from the external system on demand without replicating that information in the Siebel application.

A VBC is configured in Siebel Tools and uses business services to access data from an external system. A VBC can use a standard transport, such as MQSeries, HTTP, and MSMQ, to query, insert, and update data. A VBC also uses the XML Gateway Service.

Siebel EAI and the External Business Component

An *External Business Component* (EBC) is a mechanism in Siebel EAI that provides access to relational data sources that reside in a non Siebel table or view that uses a Siebel business component. Unlike a VBC that requires a business service, an EBC can use a Siebel database connector to access data.

An EBC is configured in Siebel Tools by 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 with Java and J2EE

Siebel Business Applications provide standards based technologies that allow you to access J2EE components from a Siebel application, and supports the creation of Java and J2EE components to access Siebel objects. For more information, see *Transports and Interfaces: Siebel Enterprise Application Integration* .

Access to a Java or J2EE Component from a Siebel Application

Siebel Business Applications provide three mechanisms to start Java and J2EE components:

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

When interacting with a J2EE component that is published as a Web Service, a Siebel application can use the Web Service Description Language (WSDL) document that describes the service and operations, then generate a proxy Business Service that allows the Siebel Business Application to call the Web Service just like it does with a local object. When the Business Service is called:

1. The Object Manager detects that the Business Service is a proxy to a Web Service
2. The appropriate Simple Object Access Protocol (SOAP) message is generated

3. The request is dispatched by using a configured transport, such as HTTP

For more information on Web Services, see *About Web Services*.

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

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

- Allows a message to be sent or received though a JMS. The JMS Receiver server component, which is similar to the MQ Series Receiver component, allows for asynchronous receipt of messages, except you use the EAI JMS Business Service instead of the EAI MQSeries Server Transport.
- Uses the Java Native Interface API that is provided by Java Virtual Machines (JVM) and native code interaction. It is a widely used technology that is used for writing JDBC drivers.

Access to a Siebel Application from a Java or J2EE Component

The ways that a Java and J2EE component can request information from a Siebel application include:

- **Java Data Bean.** The Java Data Bean is a collection of Java classes that allow a developer to interact with a variety of Siebel objects, such as business objects, business components, and so forth. With this interface, you can develop Java and J2EE components that interact with a Siebel application.
- **Siebel Resource Adapter.** The Siebel Resource Adapter plays a central role in the integration and connectivity between a Siebel application and a Java application server. It serves as the point of contact between application components, application servers and enterprise information systems. Resource adapters, along with the other components, must communicate with each other based on well defined contracts that are specified by the J2EE Connector Architecture.
- **Web Service.** Web Services are emerging as an important technology for exposing application functionality that is independent of the underlying technology used to provide that functionality. Release 6.x introduced business services that could be called 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, see *About Web Services*.
- **Java Message Service.** Enterprise messaging is recognized as a tool for building enterprise applications. The Java Message Service (JMS) is a standard Java API that is used to access an enterprise messaging system. 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 supports a wide range of enterprise messaging products. JMS supports both message queuing and publish and subscribe styles of messaging.

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

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

Siebel EAI and Object Interfaces

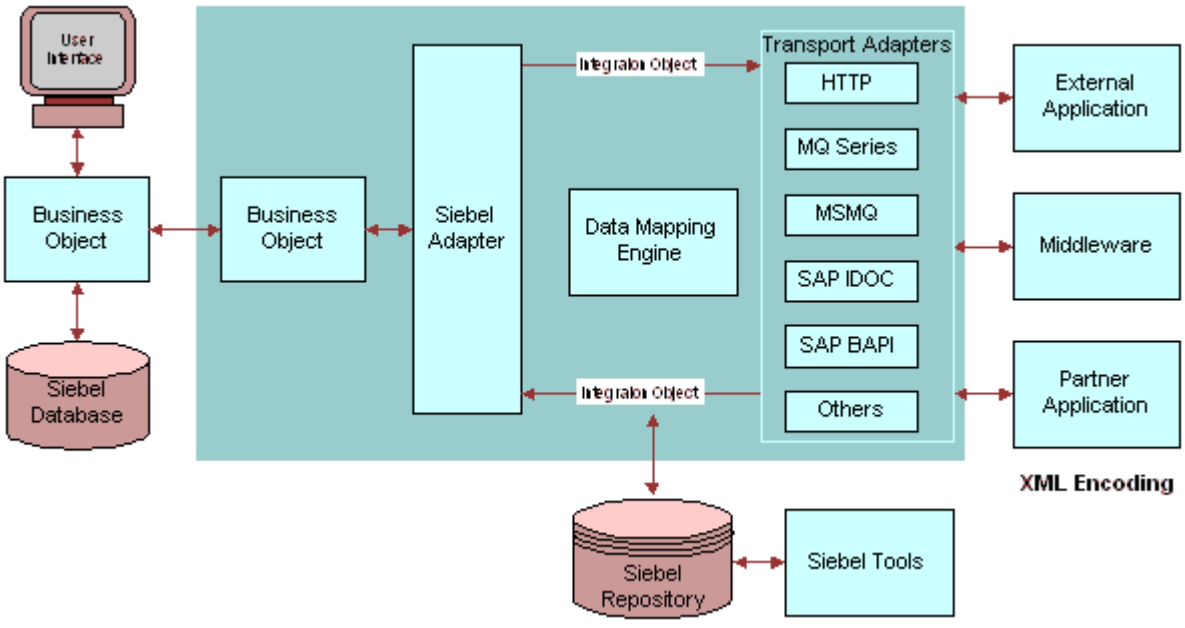
A Siebel application supports several programmatic interfaces to facilitate communication between an external application and a Siebel application. These interfaces include:

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

For more information, see *Siebel Object Interfaces Reference* .

Siebel EAI and Message Based Integration

The following image illustrates how Siebel EAI supports message based integration with an external application, where the Siebel application is the initiator of the request or the receiver.



Siebel EAI and Integration Objects

An *integration object* is the contents of messages that are exchanged, as described in *Siebel EAI and Message Based Integration*. An integration object is defined in Siebel Tools.

The types of integration objects include:

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

A configuration engine provides configurable data transformation to match the difference in the way that entities are represented, while integration objects represent the information exchanged.

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

Siebel EAI and Data Transformation

Siebel EAI transforms the data that is expected by a Siebel application, which is the internal integration object, to the data that is expected from the external system, which is the external integration object. Siebel Data Mapper is a declarative data mapper that is used to perform the transformation. The EAI Data Mapping Engine is a run-time engine that implements complex domain and structure mapping.

Siebel EAI and Workflow Process Control

To achieve integration at the process level between a Siebel application and an external application, the Siebel Business platform supports workflow control. You define the integration process by using the Workflow Process Designer in Siebel Tools. A workflow process is executed in an application object manager as a business service. A business service can be a prebuilt component, such as a transport adapter or the Siebel Data Transformation Engine, or it can be custom developed by using one of the Siebel Scripting languages. A workflow process can also be executed in the Workflow Process Manager server component. For more information, see *Siebel Business Process Framework: Workflow Guide* .

About Interfaces for Siebel Application Services

A *Siebel Application Services Interface* (Siebel ASI) is a prebuilt interface to Siebel business processes. This interface exposes Siebel functionality as services to the Siebel application environment, and allows a Siebel business process to use services provided by an external application. A Siebel ASI provides ease of maintenance and can be configured by an administrator using the Web Services Administration screens which do not require detailed knowledge of Siebel EAI.

Because the integration interface to the Siebel application remains unchanged with each upgrade to a release, the integration interface is independent of a specific release. The interfaces in Siebel ASI are extensible and are accessible as a Web Service or through a third-party Enterprise Application Integration (EAI) solution. These benefits simplify integration between a Siebel application and an external system, resulting in a significant reduction in the costs associated with integration, maintenance, and upgrades.

An ASI is 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). They are built using the ASI Framework.

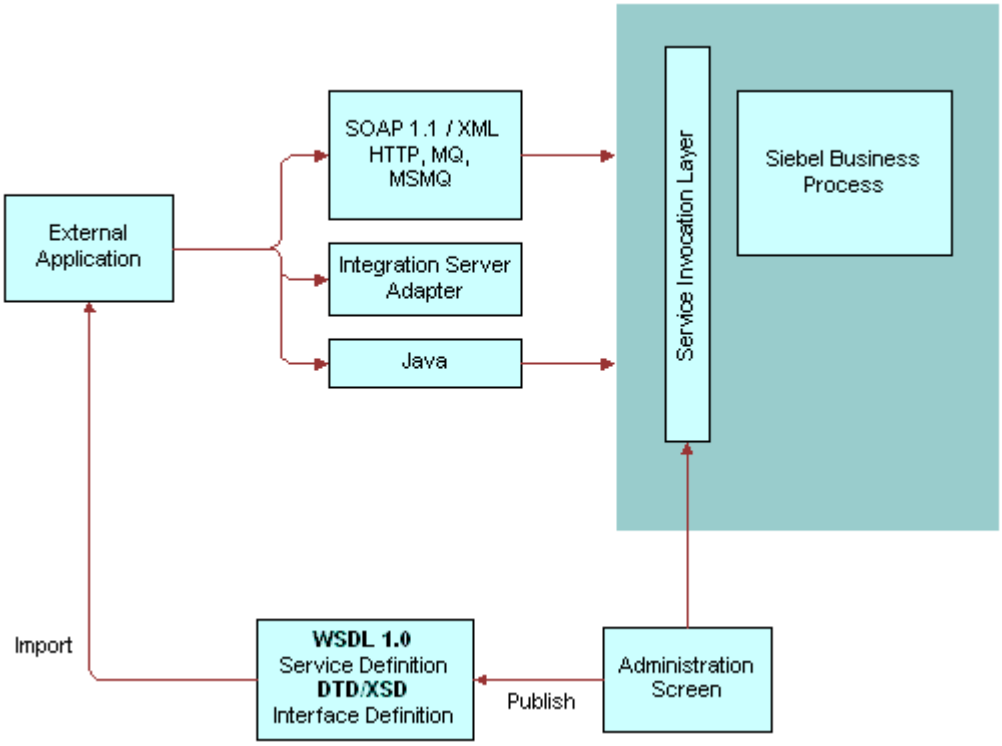
For more information, see *Siebel Application Services Interface Reference* .

Types of ASI Interfaces

An ASI is either an inbound ASI and or an outbound ASI.

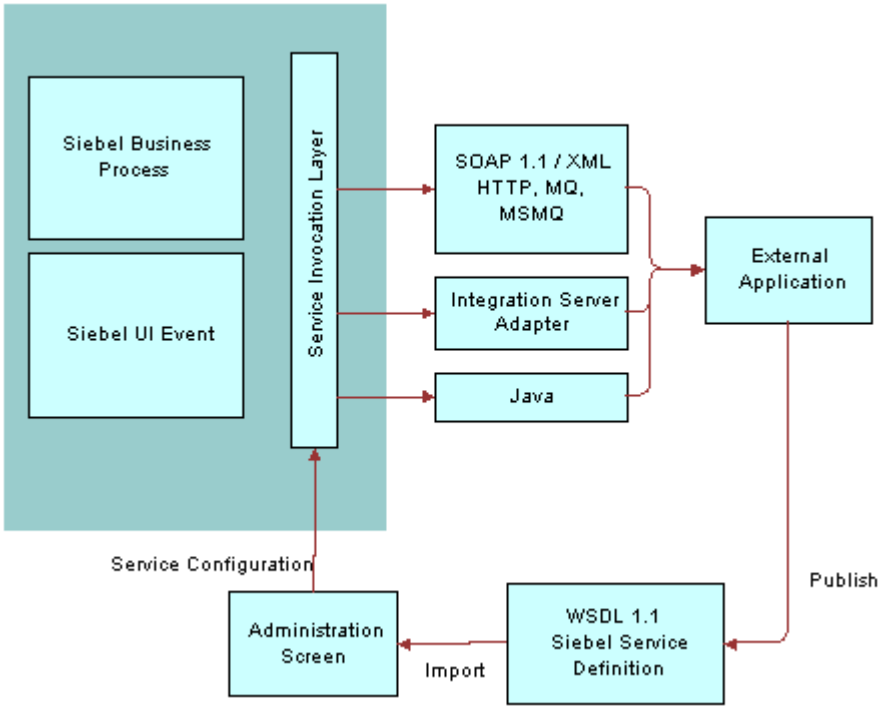
Inbound ASI

The following image illustrates how an inbound ASI is used to accept data into a Siebel application from an external system by using Siebel workflow processes, Siebel business services, and Siebel data synchronization services through the Siebel Object Manager.



Outbound ASI

The following figure illustrates how an Outbound ASI is used to send data that is based on a UI event or a trigger in a Siebel workflow process.



About Web Services

A web service is programmable application logic that is accessed by using a standard Internet protocol, such as HTTP, and data formats, such as XML. Because a web service combines component based development with the Internet, it is reused regardless of how the service is implemented.

A web service is based on communication protocols that include HTTP, XML, Simple Object Application Protocol (SOAP), and Web Services Description Language (WSDL). A web service must be developed on a computer platform and in a development environment that can communicate with another web service that also uses these common protocols.

Features provided with a web service include:

- **WS Security Support.** The Web Services 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 an environment that is independent of a particular platform or language.
- **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 conforms to the specification, as defined by the (WS-I) Basic Profile specification from the Web Services Interoperability Organization. Adherence to this specification makes sure that the Siebel application can interoperate with an external provider of a Web Service.
- **Web Services Cache Refresh.** The Web Services cache stores the global administration information that is manipulated in the Inbound and Outbound Web Service administration screens. The information contained in these services might require a refresh in order to provide more current or correct functionality. An

administrator can directly refresh the memory cache in real time, without stopping and restarting Siebel services.

For more information, see *Integration Platform Technologies: Siebel Enterprise Application Integration* .

Data Access Control

You can administer access control by associating responsibilities with business services and business service methods. The Siebel application is configured so that an end user, who accesses a Siebel application through a custom built user interface for a Web application, is provided different levels of access to the Siebel database. For more information, see *Siebel Security Guide* .

About Siebel EAI and Unicode Support

A multilingual company that includes divisions in different countries who use different languages can use Unicode to deploy a single instance of a Siebel application release. However, this technique presents a unique integration challenge because the other systems in the multilingual IT landscape for the company might or might not allow Unicode. Unicode deployment modes that address this issue include:

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

For these deployment modes, integrating Siebel Business Applications with an external application might involve data replication, data sharing, or both data replication and data sharing. If the code page database is used, and if the code page of the external system matches the code page of the Siebel Database, then data replication is allowed.

Exchanging Data Between Applications that use Different Code Pages

If data is exchanged between applications that use different code pages, then the numeric representation of the code point for each character must be converted from the standard that is used by the source code page, into the equivalent character that is used by the destination code page. This conversion requirement is true when a character is 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.

If there is no equivalent character in the destination code page, and if the conversion cannot be performed, then a conversion error is generated. Depending on the configuration, the conversion error might either terminate the current transaction, or the transaction might proceed by substituting replacement characters, as necessary, and ignoring the conversion error. For more information, see *Transports and Interfaces: Siebel Enterprise Application Integration* .

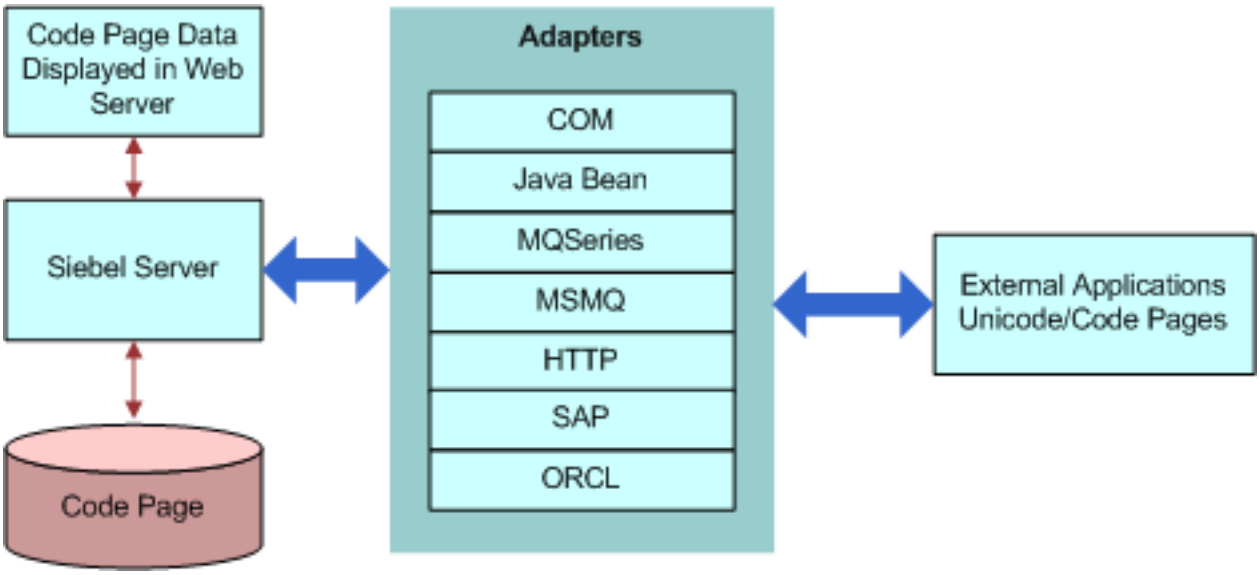
The following image illustrates how Siebel EAI supports the following adapters to integrate with an external application:

- COM
- Java Bean
- MQSeries

- MSMQ
- HTTP
- SAP
- ORCL

Abilities inherent with each of these adapters include:

- Work with the character encoding, as specified in the picklist
- Perform conversion to or from the external encoding
- Handle conversion errors



COM and Java Bean Unicode Protocols

COM and Java Bean are Unicode protocols. The sender and the recipient are responsible for converting the character sets. For example, because interfaces for a Siebel application automatically handle the conversion, these protocols are not required to perform a conversion. Error handling is handled differently by these protocols when compared to the other adapters.

Character Set Encoding

Siebel EAI components and adapters support code page conversion and allow you to specify how to handle a conversion error. The business services in which the Character Set Conversion for the Text Data argument is available include:

- EAI MQSeries Server Transport
- EAI DLL Transport
- EAI File Transport
- EAI HTTP Transport
- EAI MSMQ Transport
- EAI XML Converter

- EAI JMS Business Service
- Transcode

If one of these business services is called from a workflow process, then the valid set of encodings is controlled by a picklist. However, if the business service is called through a script or other similar mechanism, then the character set name is supplied textually. For example, if the Siebel application communicates with an external system through the EAI MQSeries Server Transport business service, then the called of EAI MQSeries Server Transport to send or receive a message also specifies the character set that the external system sends or expects to receive.

Target character encoding choices include:

- None
- UTF-8 Unicode
- UTF-16 Unicode
- Traditional Code page

For more information, see *Siebel Global Deployment Guide* .

Inbound and Outbound Unicode Scenarios for XML Messages

Because the memory format for a Siebel application is Unicode, if an XML message is sent or received, then the message must be converted to the Siebel format or to a character set for the external system. It is only necessary to perform this conversion one time for each communication.

The conversion is performed in one of the following ways:

- Using the EAI XML Converter business service in your workflow process
- Using the transport business service
- Using the Transcode business service when neither of the first two conversion techniques described in this list are available

Note: Do not convert your data more than one time for each communication. Double conversion corrupts your data. If you use an EAI XML Converter in your workflow process, make sure your transport business service is defined to ignore conversion.

Transcode Business Service Usage with Non XML Data

If non XML data is exchanged, then you can use the Transcode business service in a workflow process to validate or convert data from one encoding to another encoding. You can use this technique for inbound and outbound communication.

Transcode Business Service Usage with Outbound Communication

Consider an outbound communication between a Unicode Siebel application and a non Unicode external system that uses a Unicode communication channel, such as XML. In this situation, ways in which you can use methods on the Transcode business service include:

- Use the Validate method to execute preemptive logic in order to avoid a character conversion failure on the external system
- Use the Validate method to log a character conversion error that occurs on the external system
- Use the Convert method to use a substitute character to avoid an error on the external system

Transcode Business Service Usage with Inbound Communication

Consider an inbound communication between a non Unicode Siebel Database and a Unicode external system that uses a Unicode communication channel such as XML. In this situation, ways in which you can use methods on the Transcode business service include:

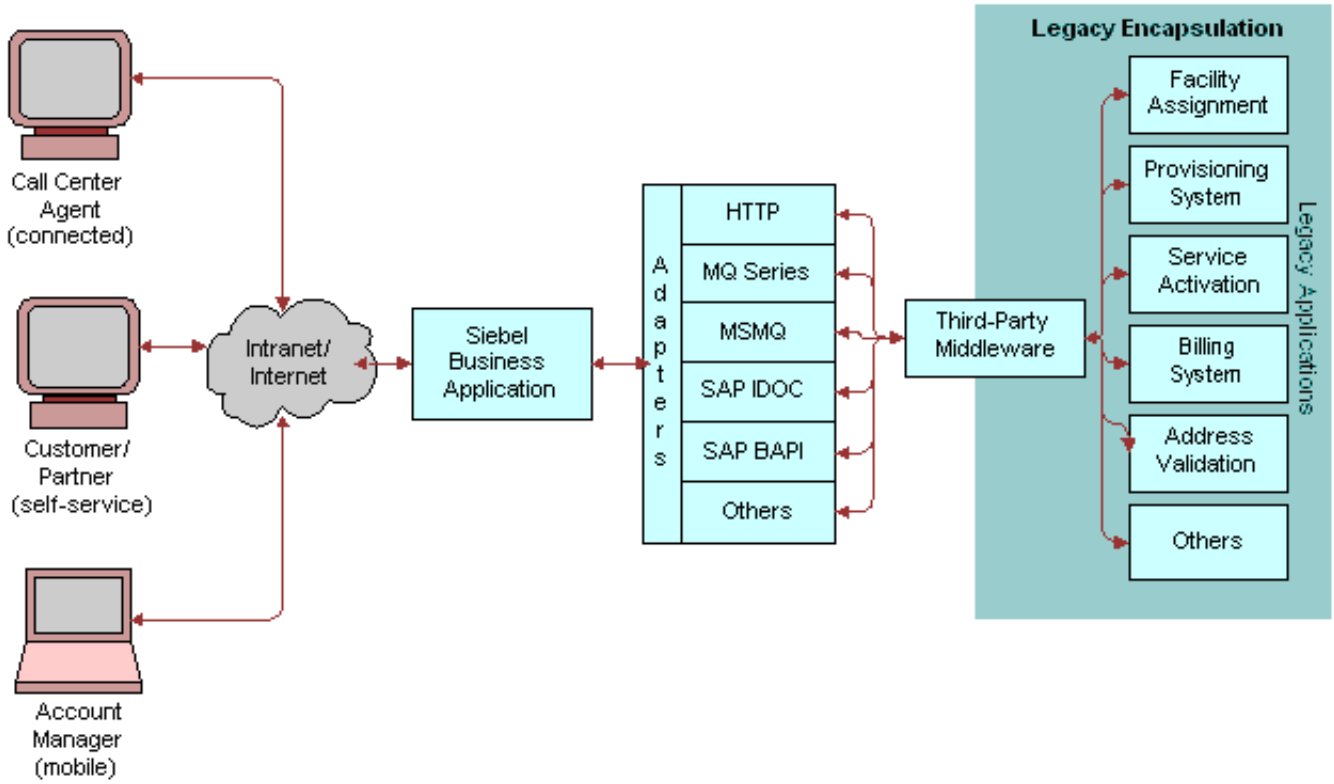
- Use the Validate method to check that the data sent to the Siebel Database is stored without a character conversion error
- If there is a character conversion error, then use the Convert method to use substitute characters, when appropriate

For more information, see the *Siebel Global Deployment Guide* .

About Siebel EAI and Legacy Systems

In an encapsulated environment, multiple similar legacy applications are rationalized through the construction of high level APIs that hide underlying complexity. The following image illustrates how message oriented middleware can help a business 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 that are expected by the encapsulated environment. Examples of message oriented middleware include:

- IBM MQSeries Queue Manager and MQSeries Integrator
- A custom development environment that is based on Enterprise Java Bean, such as IBM WebSphere

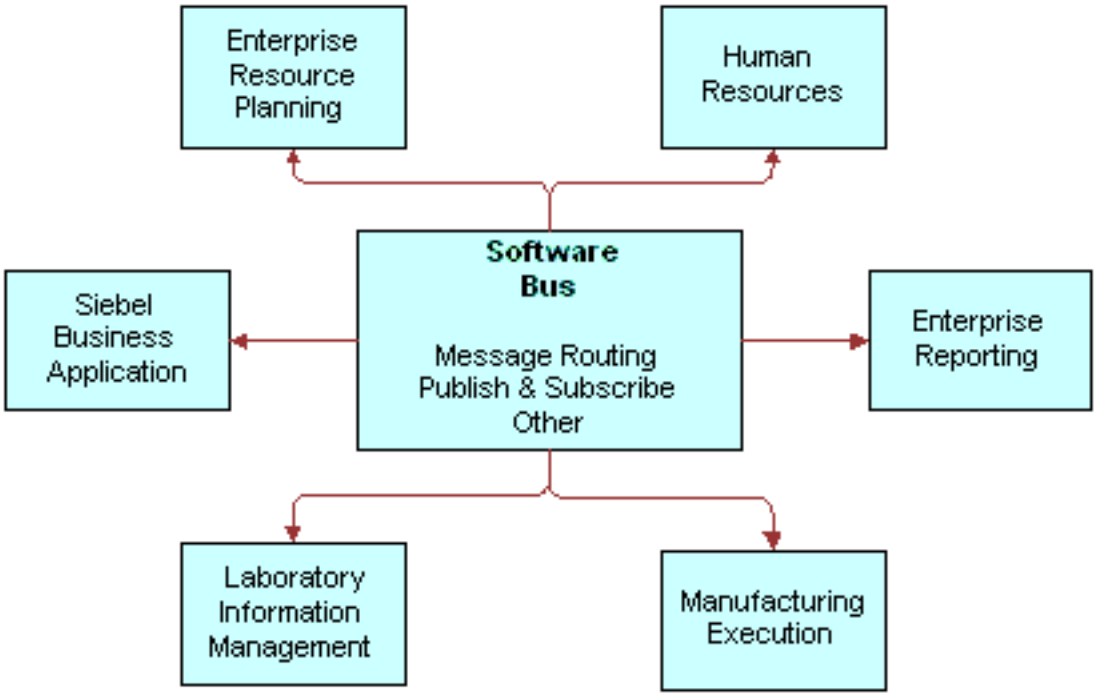


About Siebel EAI Integration Strategies

This topic describes an overview of integration strategies.

Peer-to-Peer Integration

The following image illustrates peer-to-peer integration, where multiple applications (Human Resources, Enterprise Reporting, Manufacturing Execution, Laboratory Information Management, Siebel Business Application, and Enterprise Resource Planning) interact with each other through a central software bus. The software bus provides a uniform interface to each application. The bus also provides a number of services that can range from simple data transformation, to a more complex publish and subscribe scheme that allows a set of applications to automatically receive copies of events that are of interest, thus keeping these applications synchronized with the rest of the organization. Siebel EAI supports the services that the software bus expects from each participating application.



An enterprise can adopt a software bus strategy for application integration. Master data is partitioned across systems, and the software bus middleware implements a messaging architecture to exchange application data. 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. Because batch integration optimizes interactions for high throughput, it uses computing resources more efficiently. By contrast, real-time integration optimizes response time.

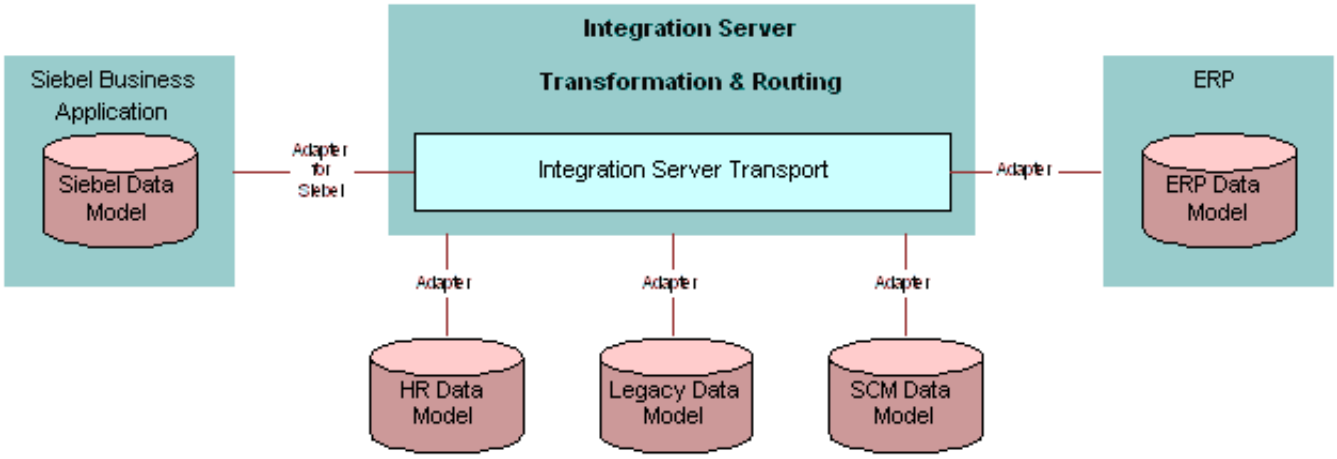
Application Connection with an Integration Server

Integration can be achieved by connecting an application through an integration server that provides an information bus or a hub and spoke architecture to tie disparate applications together. The benefit of using an integration server is that it reduces the number of physical connections.

The following image illustrates how each of the following applications connect only one time, through an adapter, to the integration server:

- Siebel Business Application and Siebel Data Model
- HR Data Model
- Legacy Data Model
- SCM Data Model
- ERP and ERP Data Model

The integration server also provides a reliable integration infrastructure that is successfully implemented in a large and critical deployment.



A Siebel application can integrate with an integration server by using an adapter provided by a partner, such as IBM, TIBCO, or WebMethods. For information about supported integration servers, see Siebel System Requirements and Supported Platforms.

Other Integration Strategies

Some of the other integration strategies that can be used include:

- **Multiple Application Process Integration**. A successful business requires business processes that span multiple applications. For example, a quote to cash business process spans credit check, inventory management, billing, shipping, and other application services. Siebel EAI facilitates multiple application business processes.
- **Master Data Partitioning.** Master data refers to the set of data that is required by a community of applications. Master data is partitioned between applications. Although each piece of data is owned by one application, 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 file for customer information typically resides on a mainframe.
- **Presentation Layer Integration.** Siebel EAI supports client integration in a Web client environment. Client integration includes 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 screen into the content of another screen.
- **Data Mapping and Transformation.** Data that is shared between applications might not be structured or encoded in the same way. To facilitate integration of data structures that are not identical, 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 applications.
- **Upgradeable Integration.** The Siebel Business platform is designed so that integration points are migrated to a new release of Siebel Business Applications and adapted to new releases of external application packages. This capability means an organization can use a prior investment in their integration when upgrading to a new version of a Siebel application, or to a third-party application.

3 Integration with Siebel EAI

Integration with Siebel EAI

This chapter describes integration options with Siebel EAI. It includes the following topics:

- *Overview of Integration with Siebel EAI*

Overview of Integration with Siebel EAI

This topic describes an overview of integration with Siebel EAI. It includes the following topics:

- *Integration Options with Siebel EAI*
- *Design for an Application Integration Project*
- *Real-Time and Batch Mode Usage Models*
- *Data Exchange with a Siebel Application*
- *Summary of Integration Options with Siebel EAI*

Before you begin an integration project, you must identify your integration requirements. Siebel Business Applications must interact with external systems in order to provide an integrated view of your data. Applications in which data resides are varied. For example, they can include:

- A back office solution for order fulfillment, manufacturing, or accounting
- An e-commerce application, such as a Web storefront
- A help desk application

Integration Options with Siebel EAI

Depending on your requirements, the ways that you can approach your integration include:

- Use prebuilt Siebel EAI connectors to integrate a Siebel application with your back office system or your e-commerce application
- Use custom built Siebel EAI connectors to integrate a Siebel application with an external application
- Use a third-party EAI product to integrate a Siebel application with an external application
- Use a middleware solution to integrate a Siebel application

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

- A programmatic interface, such as COM or Siebel Java Beans, provides a tightly coupled integration for real-time data exchange.
- An adapter, such as MQSeries or the HTTP adapters, provides a loosely coupled integration for real-time data exchange. This means that one system must be aware of the implementation details that are defined for the integration of the other system. A message or a document is exchanged between these systems.

- Siebel Enterprise Integration Manager (EIM) provides a solution for a high volume batch integration. For more information, see *About the Enterprise Integration Manager*.

Design for an Application Integration Project

The questions you must answer as part of the planning phase for an application integration project include:

- How is the data that is used in my organization shared?
- Which access mechanism makes the most sense for my situation?
- Which preexisting connector best applies to my situation?

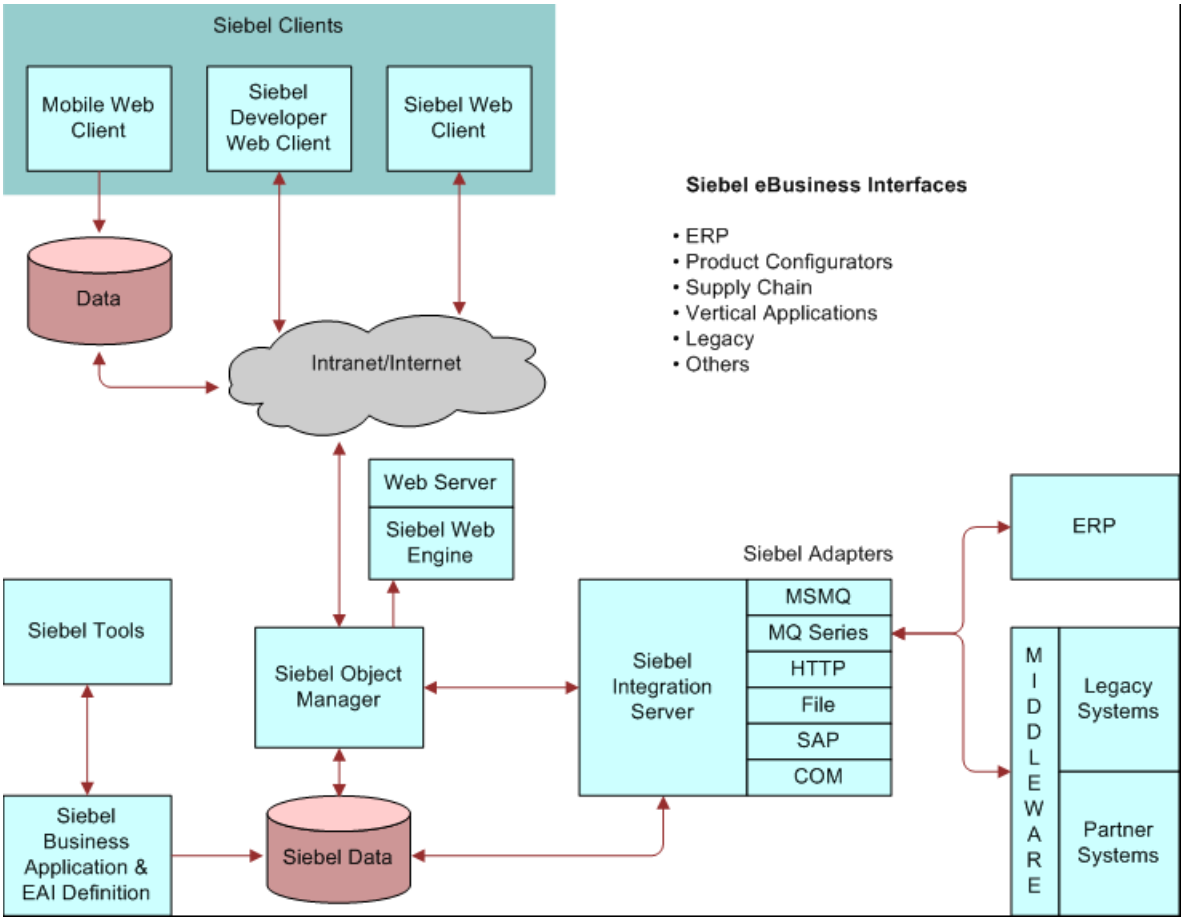
It is important to identify your requirements before you design the integration. Categories on which your design choice can be based include:

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

Siebel Business Interfaces

After you design your integration requirements for each business process flow, you can apply the different components of Siebel EAI to implement a solution. The following image illustrates the different types of applications with which Siebel Business Applications can integrate as follows:

- Siebel eBusiness Interfaces: ERP, Product Configurators, Supply Chain, Vertical Applications, Legacy applications, Other applications.
- Mobile Web Client, Siebel Developer Web Client, Siebel Web Client.
- Web Server, Siebel Web Engine, Siebel Object Manager, Siebel Integration Server, Siebel Adapters (MSMQ, MQ Series, HTTP, File, SAP, COM).
- Siebel Data, Siebel Business Application & EAI Definition, Siebel Tools.

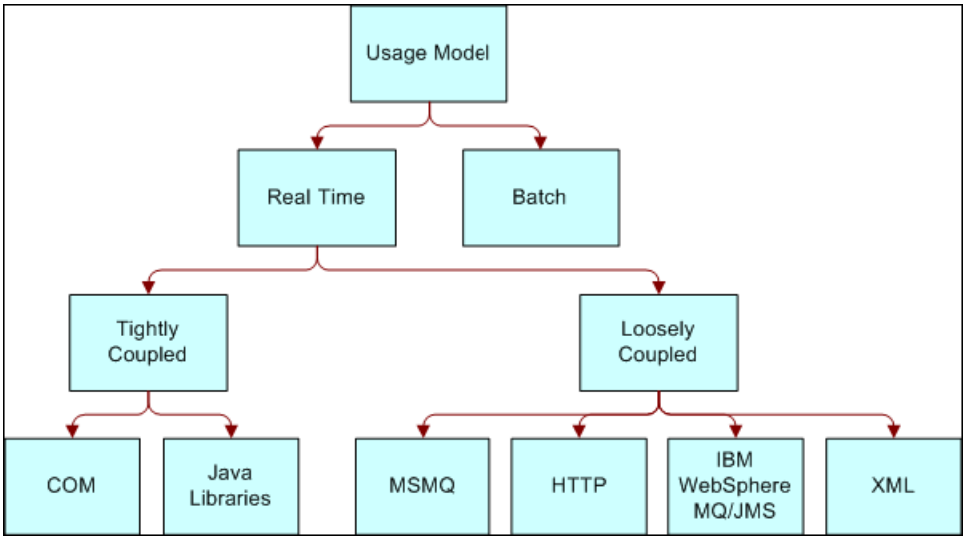


Real-Time and Batch Mode Usage Models

When considering your requirements, you must develop an understanding of how you intend to use the integration capabilities that are provided by Siebel EAI. You must determine how the integration behaves under specific circumstances. How are you planning to use this integration? This model is referred to as the *usage model*.

The following image illustrates real-time and batch usage models. Batch mode uses a single mode, while real-time uses two modes, either tightly coupled or loosely coupled. To help you determine the appropriate usage model, answer the following questions:

- How do I plan to integrate Siebel applications with other applications?
- Do I need to integrate in real time, or in batch mode?
- If I need to integrate in real time, do I need the solution to be tightly coupled or loosely coupled?



Integration with the Real-Time Model

With a real-time model, you might 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 initiation from Siebel applications.

Typical scenarios that involve Siebel applications as a client include:

- Replicate data. For example, to maintain the SAP order number in the Siebel application so that an end user can relate a Siebel order to a SAP order.
- Access a mainframe system to retrieve information for a customer account in real time, then display it within the interface for the Siebel application.
- Send a sales opportunity to a trading partner over your corporate firewall without compromising system security.
- Access a function for an external system to check inventory, then display the results in a Siebel application.

Typical scenarios that involve Siebel applications as a service include:

- An external Web application that accesses a Siebel application in order to retrieve information for a contact.
- Call center software that extracts information for a service request from a Siebel application.

Integration with the Batch Model

Batch processing is typically used to periodically upload or download a large amount of information into or out of the Siebel Database. Typical batch scenarios that involve 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.

Data Exchange with a Siebel Application

If your business requirement is to access another application from within a Siebel application, or to access a Siebel application from within another application, then you must determine exactly the type of data exchange that is required, which can include:

- **Data Sharing.** You might be required to share data that exists outside of the Siebel Business Application. For example, several legacy mainframe systems might exist in your environment that you must access with Siebel Call Center in order to provide a customer with information on their account. Instead of replicating this information across Siebel Business Applications, you can dynamically share it and display it within a single Siebel application. For data sharing, use a virtual business component.
- **Replication.** Replication of data is often useful. For example, you might be required to integrate data in SAP with data in a Siebel application, keeping information about order status in both systems. To mass replicate information about the order status in a Siebel application, use Siebel Enterprise Integration Manager (EIM).

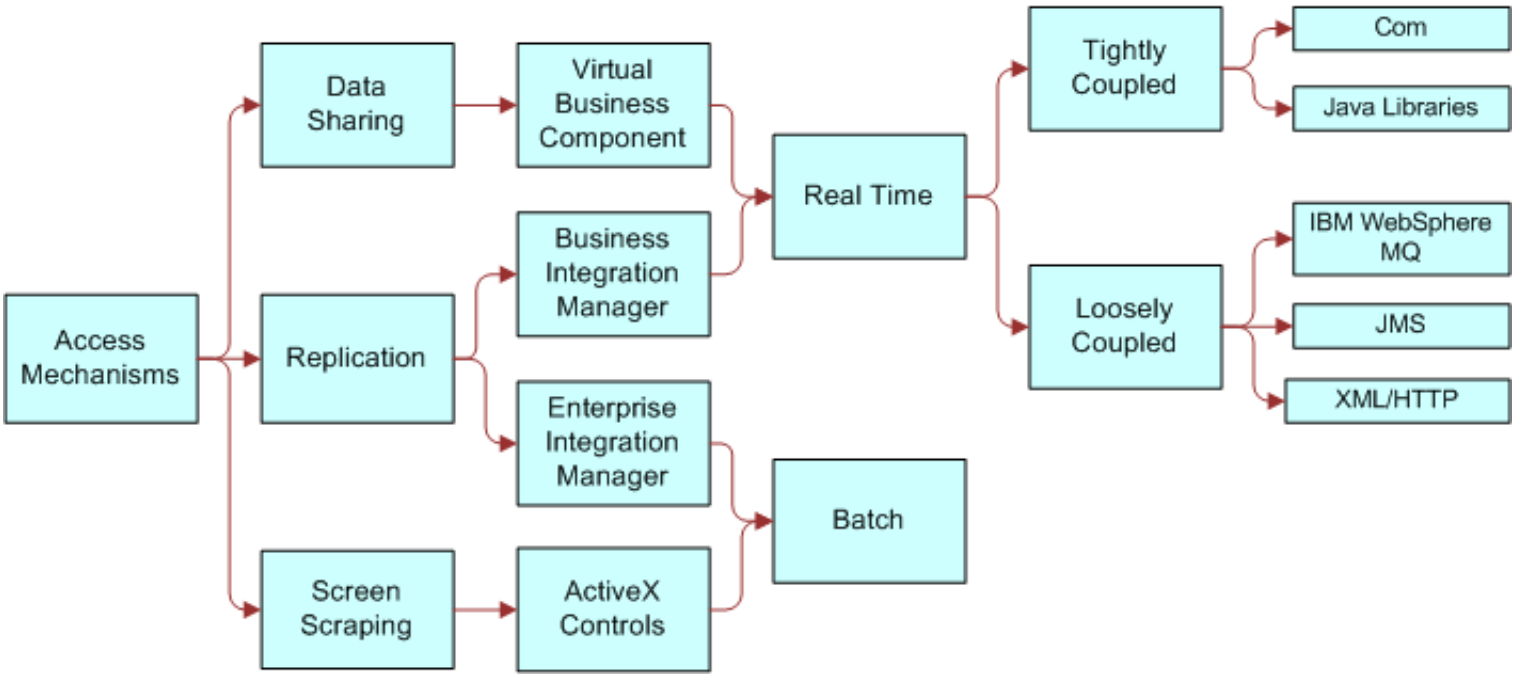
Summary of Integration Options with Siebel EAI

The following table provides a summary of the Siebel integration options that can be used in your integration strategy.

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

The following image illustrates the various access options and where they fit within an integration strategy. These Siebel solutions are either real-time or batch. The access options (or mechanisms) are as follows:

- Data Sharing, which includes the Virtual Business Component integration tool in real-time. The real-time Siebel solution can be Tightly Coupled (COM, Java Libratirs) or Loosely Coupled (IBM WebSphere MQ, JMS, XML/HTTP).
- Replication, which includes the Business Integration Manager integration tool in real-time or the Enterprise Integration Manager integration tool in batch mode. The real-time Siebel solution can be Tightly Coupled (COM, Java Libratirs) or Loosely Coupled (IBM WebSphere MQ, JMS, XML/HTTP).



4 Architecture for Siebel EAI

Architecture for Siebel EAI

This chapter describes the Siebel EAI architecture. It includes the following topics:

- *Overview of Architecture for Siebel EAI*

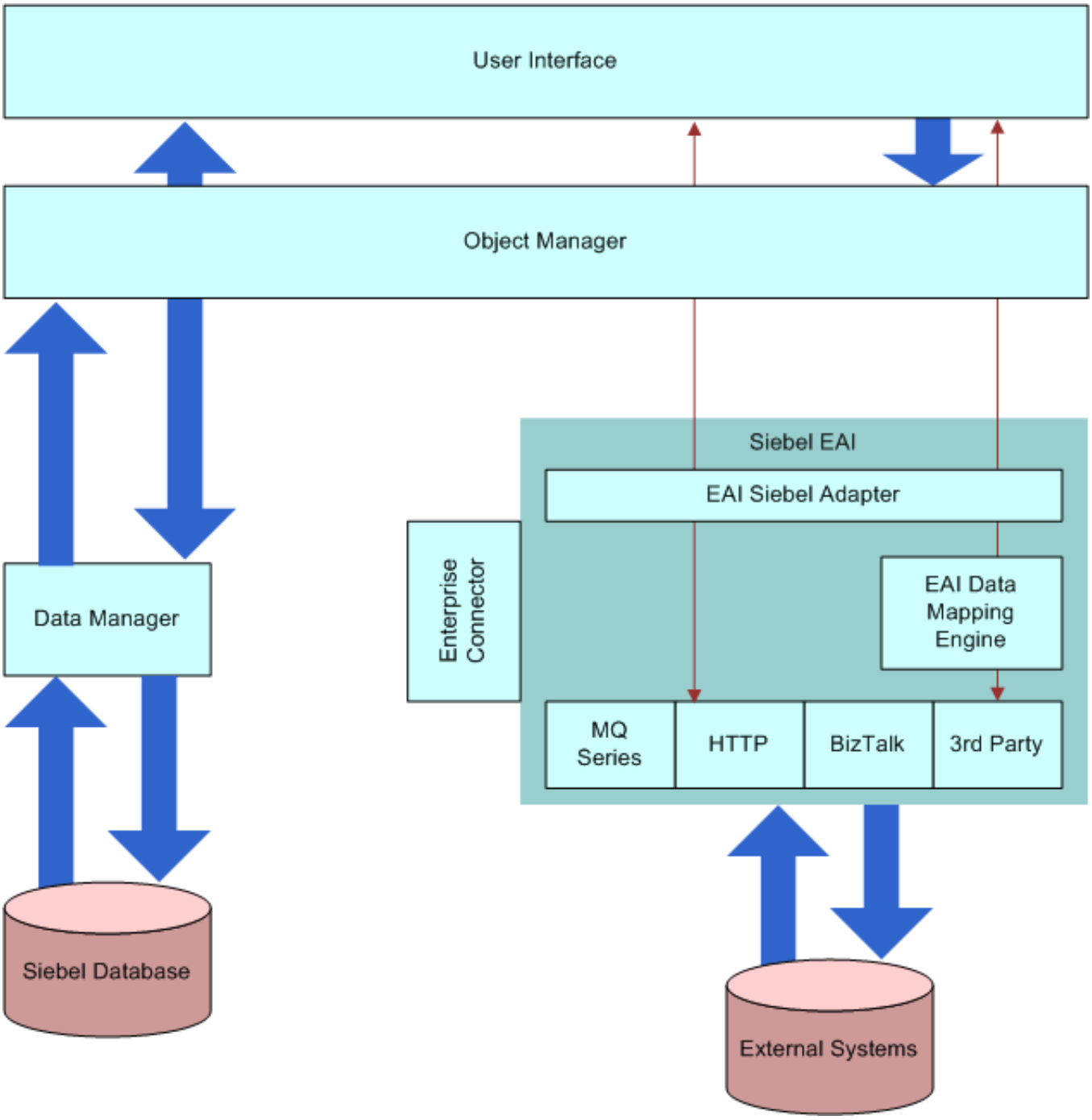
Overview of Architecture for Siebel EAI

This topic describes an overview of architecture with Siebel EAI. It includes the following topics:

- *Siebel EAI Architecture*
- *Integration Requirements for Siebel EAI*
- *Business Process Coordination with Siebel Workflow*
- *Transport Mechanisms*
- *High Volume Data Exchange*

Siebel EAI Architecture

The following image illustrates the Siebel EAI architecture, which includes the EAI Siebel Adapter, the EAI Data Mapping Engine, and multiple transports (MQ Series, HTTP, BizTalk, 3rd Party) that can facilitate data integration with different external systems.



Because data management functions are abstracted to the object layer, declarative configuration can be used instead of a procedural configuration, such as scripting. Declarative configuration improves and simplifies configuration work while lowering the costs associated with maintenance.

Integration Requirements for Siebel EAI

The following table describes data and application integration requirements and their corresponding Siebel EAI features.

Integration Requirement	Siebel EAI Feature or Component	Description
Integration requirements include: • Metadata definition • XML metadata • Metadata for an external application. For example, from an SAP application to an Oracle application, and so forth	Siebel EAI features and components include: • Integration objects based on Siebel business objects • Integration objects based on external metadata • Integration Object Wizard	In an integration project, you must supply metadata. Siebel applications provide XML as the common format for representing external application data. Siebel applications provide an object type in Siebel Tools, the Integration Object, and an Integration Object Wizard, which simplifies the work of creating an integration object.
Integration requirements include: • Data transformation • Data transformation mapping through declarative configuration • High performance transformation engine	Siebel EAI features and components include: • Siebel Data Mapper	Siebel EAI allows you to define the type of data to exchange, and allows you to export an XSD or a DTD for a Siebel integration object.
Integration requirements include: • Business process coordination	Siebel EAI features and components include: • Siebel Workflow Process Designer • Workflow Process Manager	For more information, see *Business Process Coordination with Siebel Workflow*.
Integration requirements include: • Transport mechanisms for interfacing with other technologies	Siebel EAI features and components include transport adapters, such as: • IBM MQSeries • Microsoft MSMQ • HTTP • Java Message Service	For more information, see *Transport Mechanisms*.
Integration requirements include: • High volume data exchange	Siebel EAI features and components include: • Enterprise Integration Manager (EIM)	For more information, see *High Volume Data Exchange*.
Integration requirements include:	Siebel EAI features and components include:	Not applicable.

Integration Requirement	Siebel EAI Feature or Component	Description
• Standards based Integration mechanisms	• XML as a format for data exchange • COM programmatic interfaces • Programmatic interfaces based on Java • HTTP as a transport protocol	

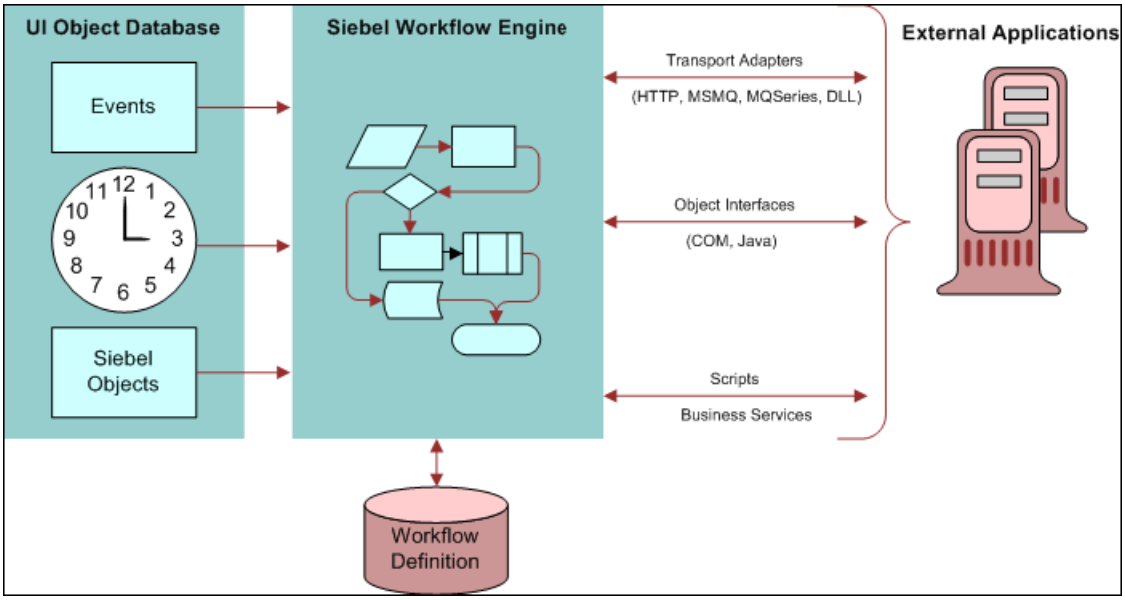
Business Process Coordination with Siebel Workflow

Coordination between multiple business processes requires that the information systems in an enterprise each possess access to the same, up to date information. For example, if a customer address is changed, then the change is reflected in every system that stores the address. The Siebel Workflow Process Designer helps you define the business process flows for your data integration.

The following image illustrates how a workflow process can respond to a trigger that is based on an event or to a trigger that is based on time. The typical steps are as follows:

1. The UI Object Database sends an event or time-based trigger to the Siebel Workflow Engine.
2. The workflow processes the trigger when it receives it and calls one of the following to communicate with an external application:
    - Transport Adapters (HTTP, MSMQ, MQSeries, DLL).
    - Scripts (Business Services).
    - Object Interface (COM, Java).

For more information, see *Siebel Business Process Framework: Workflow Guide* .



Transport Mechanisms

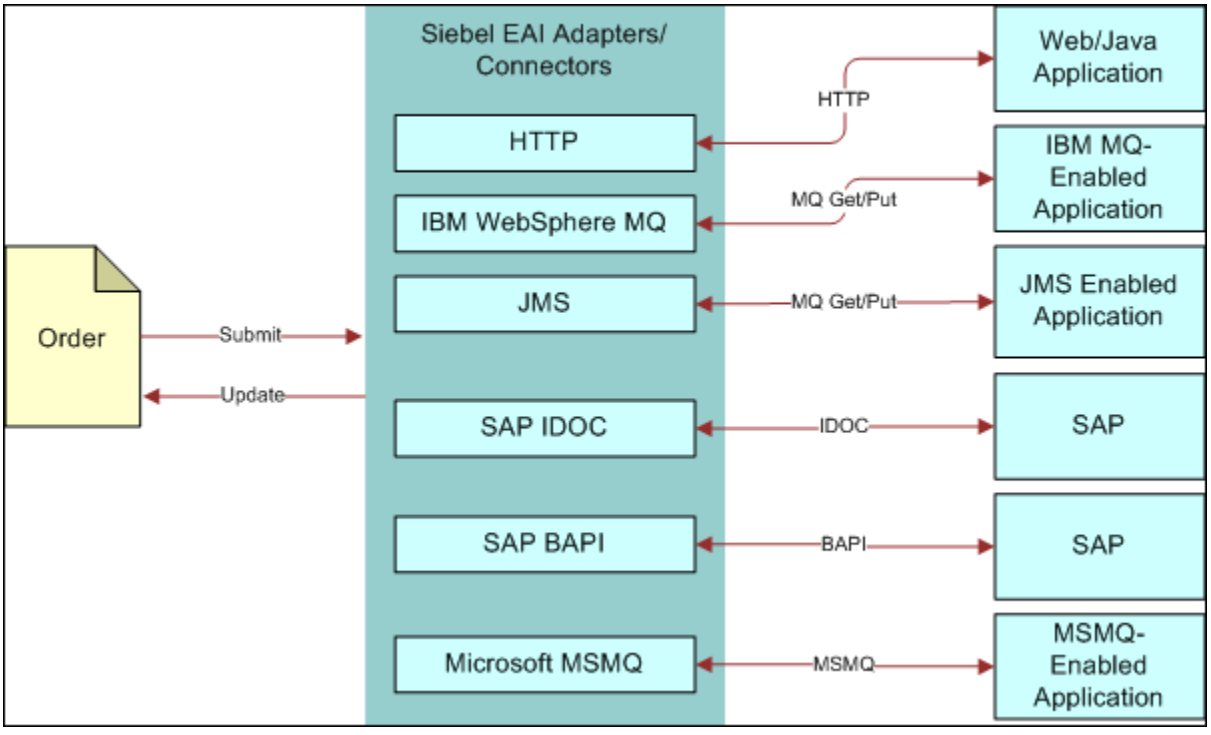
Many companies standardize on a transport solution, such as MQSeries from IBM, or MSMQ from Microsoft. These transports allow the transportation of messages between systems. Oracle conforms to this standard by allowing you to plug a Siebel application into these environments through the Siebel EAI set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

A transport adapter that is developed by Oracle or by a partner that is certified by Oracle, allows diverse applications to communicate with a Siebel application across various operating systems, networks, and databases. The following image illustrates the various transports that are available in order to communicate to and from a Siebel application when a new order is processed. The transports are as follows:

- HTTP (HTTP) transport adapter with Web/Java applications.
- IBM WebSphere MQ (MQ Get/Put) transport adapter with IBM MQ enabled applications.
- JMS (MQ Get/Put) transport adapter with JMS enabled applications.
- SAP IDOC (IDOC) transport adapter with SAP applications.
- SAP BAPI (BAPI) transport adapter with SAP applications.
- Microsoft MSMQ (MSMQ) transport adapter with MSMQ enabled applications.

One or more of these transports can be used in an integration, depending upon requirements from existing applications and other factors. These transports can also be configured as a Siebel Server Receiver to receive a message. For example, the MQSeries Server Receiver.

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



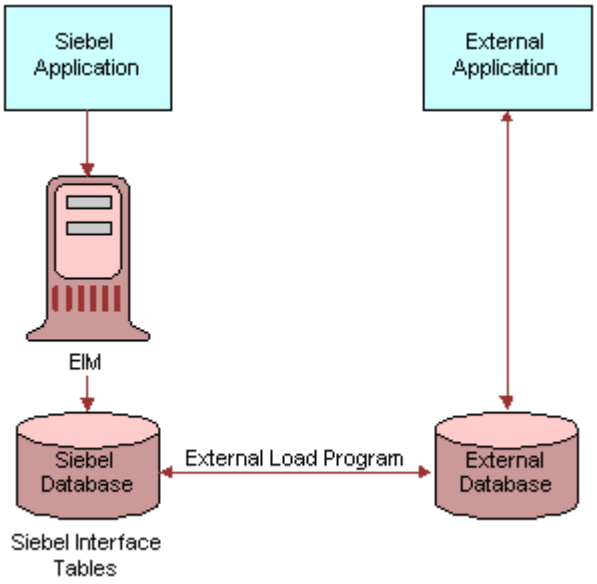
High Volume Data Exchange

In order to avoid a network slowdown or interference with another application, your organization might require a way to schedule a high volume integration at a specific time, typically at night. This type of integration often involves updating hundreds of millions of customer records. This high volume of activity requires a batch mechanism that is capable of processing large amounts of data in a timely fashion. Siebel Enterprise Integration Manager (EIM) allows you to schedule a high volume data exchange between a Siebel application and another application when such an exchange is required, and at the required frequency, such as twice a day, nightly, weekly, and so forth.

About the Enterprise Integration Manager

Siebel Enterprise Integration Manager (EIM) can replicate data between a Siebel application and another system at the database layer. EIM provides high performance to import or export large volumes of data. With EIM, you can import and export data from the Siebel Database to other corporate databases. The following image illustrates EIM data processing for batch. EIM includes database profiling that improves performance by supporting index specialization in Relational Database Management Systems (RDBMS).

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



5 Scenarios for Using Siebel EAI

Scenarios for Using Siebel EAI

This chapter describes several examples of how Siebel EAI is used to achieve data integration. You might use Siebel EAI for data integration differently, depending on your business model. This chapter includes the following topics:

- *Scenarios for Common Integrations*
- *Scenarios for Data that is Accessed, Exported or Replicated*
- *Scenarios for Data that Crosses a Boundary*
- *Scenarios for Data that is Transformed*

Although Siebel EAI allows you to accomplish integration within the scope of these scenarios, it is not limited to only the scenarios described in this topic. For more information, see *Overview of Integration with Siebel EAI*.

Scenarios for Common Integrations

The common integration scenarios described in this topic apply to many of the integration requirements you might encounter.

Outbound Message to a File

In this scenario, you define a Siebel workflow process that creates a unique file when 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, a service request that is entered from within your Web site must also be entered as a service request in Siebel Service. Also, you must pull, from Siebel Service, a service request number that is automatically generated, add it to the service request, generate a file that is attached to a confirmation email message, then send the email to the individual who requested service.

Round Trip Message from an External Application

In this scenario, when 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 a salesperson must order from the remaining supply at the current price, then the salesperson can respond to the message.

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

In this scenario, a salesperson must send an order from a Siebel application to your back office ERP system which is using Japanese 932, while the code page used by the Siebel application is UTF-8. The salesperson must receive a confirmation that includes an order number which is generated by the ERP system.

Inbound Message from an External Application

In this scenario, when product information in your SAP R/3 system is updated, the SAP R/3 system must send an updated IDOC to your Siebel system, then the product information must be updated in the Product business object in Siebel.

Outbound Message to an External Application

In this scenario, account information must be updated in your SAP R/3 IDOC from Siebel Business Applications.

Scenarios for Data that is Accessed, Exported or Replicated

This topic describes scenarios where a business process requires access to specific data in order to accomplish a process goal.

Export Customer Data to an External System

In this scenario, a change to contact information in the Siebel Database must be sent to an external mainframe system through IBM WebSphere MQ. Work you can perform in this situation can include:

1. Define and configure the message that must be exchanged. In the Siebel Database, the Contact business object contains several business components, and each business component includes several fields. It is not desirable to send all of this information to the external system. You must choose only a subset of this information.
2. Use Siebel Tools to define the integration object for the contact message you must exchange. Siebel Tools provides an Integration Object Builder that assists you to define an integration object.
3. Define how the message is handled. The contact message must be sent to the external system by using EAI MQSeries Server Transport. You can use the Siebel Workflow Process Manager to model the message flow for this integration, and to send the message. The flow for an integration message is a sequence of business services that are connected.

The following table describes the requirements to export customer data from a Siebel application to an external system.

Business Requirement	Technological Requirement	Siebel Solution
To access the Siebel Database	Use a business service to query the integration object.	EAI Siebel Adapter
To convert an internal Siebel message format into XML	Use Siebel conversion technology to convert the message to XML.	EAI XML Converter
To place the converted XML document in the destination queue	Use technology that is compatible with IBM MQSeries.	EAI MQSeries Server Transport

Access a Company Catalog through Replication

In this scenario, salespeople must be provided access to a product catalog while they are on the road. Therefore, you import the catalog information into a Siebel application so that Siebel Remote can handle the mobile replication. Siebel Enterprise Integration Manager (EIM) imports this information.

The following table describes the requirements to provide access to a company catalog through replication.

Business Requirement	Technological Requirement	Siebel Solution
To import the product catalog for the company	Use high volume batch replication.	Siebel Enterprise Integration Manager (EIM)

Access Customer History Information Without Replication

In this scenario, a company is in the financial industry. Because a financial service institution is typically cautious when handling customer data, the company must provide access to data while replicating as few times as possible, if at all. However, the customer service agent must access the transaction history for your customer in order to provide the best service. This information is typically stored in Customer Information Files (CIF) on a mainframe computer.

The Siebel Virtual Business Component (VBC) can handle this business requirement. A VBC abstracts external data and presents it to a Siebel application as part of the Siebel data model. Siebel business logic and user interface components process a VBC in the same way that they process a standard business component.

The following table describes the requirements to integrate customer transaction history information into a Siebel application.

Business Requirement	Technological Requirement	Siebel Solution
To provide access to the Customer Information Files (CIF) that are stored on a mainframe computer, and display this information in the customer portal	Use access to non Siebel data without replication.	Siebel Virtual Business Component (VBC)

Access to Java Data

In this scenario, a customer service Web application is written with JavaServer Pages (JSP). To pass this data into a Siebel application so that the account record is updated, you call Siebel Java Data Beans in the Siebel Object Interface. This way, a customer service representative can use the information from the Web application JSP while serving the customer.

The following table describes the requirements to access Java data.

Business Requirement	Technological Requirement	Siebel Solution
To update Siebel account data from an application that uses JSP	Use a programmatic interaction.	Siebel J2EE Resource Adapter and Application Service

Scenarios for Data that Crosses a Boundary

A business process might be required to cross functional, departmental, or enterprise boundaries. Because different organizations might use different applications, you require a way to make sure an application can hand off the execution of a business process to another application.

Access to Business to Business Data

Business-to-business (B2B) integration allows partner companies to work together in order to satisfy a customer demand. Because B2B is fundamentally about carrying out a transaction over the Internet, the technology must work over the infrastructure that is used with the Internet. To meet this requirement, Siebel EAI provides the EAI HTTP adapter and the Siebel Workflow Process Manager.

The following table describes the requirements to implement a B2B transaction.

Business Requirement	Technological Requirement	Siebel Solution
To create a B2B integration in order to send an order directly to a partner over the Internet	Use XML messaging over HTTP.	Siebel solutions include: • Siebel Workflow Process Manager • EAI HTTP Transport adapter

Access to Legacy Data

Interfacing with a custom legacy application presents challenges that are similar to those described in *Access to Business to Business Data*. On the Internet, HTTP is a standard protocol that is used to exchange data. In a mainframe

environment, IBM MQSeries is a common application messaging transport. Therefore, you can replace the HTTP protocol in this solution with MQSeries and still finish the transaction.

The following table describes the requirements to access legacy data that exists on a mainframe.

Business Requirement	Technological Requirement	Siebel Solution
To create an order with a mainframe system	Use business process management that can cross applications.	Siebel solutions include: • Siebel Workflow Process Manager • EAI MQSeries Transport

Scenarios for Data that is Transformed

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

- An Internal integration object mirrors the data structure of a Siebel business object and contains a subset of business components and fields.
- An external integration object mirrors the data structure of an external system.

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, if used outside the Siebel environment, then both of these business services perform an operation on an external integration object instance, as opposed to performing an operation on an internal integration object instance when used within the Siebel environment.

You use business services to define data mapping and data transformation rules. Siebel EAI provides many mappings and transformation rules in the prebuilt external adapters. You define internal and external integration objects in Siebel Tools.

> **Note:** Business services provide a flexible scripting environment to map data from the Siebel data structure to an external data structure. Within a business service, a Siebel application provides a set of methods that allow you to perform an operation on an instance for an integration object.

Pass Data to an External Application

In this scenario, an order is created in a Siebel Business Application, and is then passed to an external application. The external application returns the order status and an order number. Work you can perform in this situation can include:

1. Identify the Siebel business object to update. For that business object, identify the components and fields to update.
2. Identify the object in the external system with which you must integrate.
3. Determine the mapping between the Siebel business object and the external object.
4. In Siebel Tools, use the Integration Object Wizard to create a new internal integration object. Choose only the components and fields that must be updated.
5. In Siebel Tools, create a new external integration object.

6. In either Siebel Tools or in the Siebel client, use business services to define the data mapping and data transformation.
7. In Siebel Tools, define an integration workflow process and add exception handling to it.
8. Test the workflow process using the process simulator.
9. Add a mechanism to call the workflow process from Oracle's Siebel applications.

The following table details the requirements for data transformation integration.

Business Requirement	Technological Requirement	Siebel EAI Solution
To create a new internal Integration Object	Use the integration object tool or wizard.	Siebel Integration Object Wizard
To create a new external Integration Object	Use the integration object tool or wizard.	Siebel Integration Object Wizard
To create a new business flow	Use workflow process technology.	Siebel Workflow Process Designer