

Understanding webMethods and Intelligent Business Operations Products

Version 9.8

April 2015

This document applies to webMethods Product Suite Version 9.8 and to all subsequent releases.

Specifications contained herein are subject to change and these changes will be reported in subsequent release notes or new editions.

Copyright © 19982015 Software AG, Darmstadt, Germany and/or Software AG USA Inc., Reston, VA, USA, and/or its subsidiaries and/or its affiliates and/or their licensors.

The name Software AG and all Software AG product names are either trademarks or registered trademarks of Software AG and/or Software AG USA Inc. and/or its subsidiaries and/or its affiliates and/or their licensors. Other company and product names mentioned herein may be trademarks of their respective owners.

Detailed information on trademarks and patents owned by Software AG and/or its subsidiaries is located at <http://softwareag.com/licenses>.

This software may include portions of third-party products. For third-party copyright notices, license terms, additional rights or restrictions, please refer to "License Texts, Copyright Notices and Disclaimers of Third Party Products". For certain specific third-party license restrictions, please refer to section E of the Legal Notices available under "License Terms and Conditions for Use of Software AG Products / Copyright and Trademark Notices of Software AG Products". These documents are part of the product documentation, located at <http://softwareag.com/licenses> and/or in the root installation directory of the licensed product(s).

Use, reproduction, transfer, publication or disclosure is prohibited except as specifically provided for in your License Agreement with Software AG.

Table of Contents

About this Guide.....	7
Document Conventions.....	7
Online Information.....	8
Application Integration.....	9
Application Integration.....	10
Developing Application Integrations.....	11
Software AG Designer.....	11
Application Platform.....	12
CloudStreams.....	12
Adapters.....	13
EntireX.....	14
ApplinX.....	16
Web Enablement.....	17
Instant Solution.....	17
HTML Emulation Solution.....	17
Composite Solution.....	17
SOA Enablement.....	17
Webpage Integration.....	18
Executing Application Integrations.....	18
Integration Server.....	18
Universal Messaging.....	19
Application Integration Examples.....	20
Synchronization.....	20
Propagation.....	20
Composition.....	21
Extending Your Application Integrations into the Mobile Space.....	22
Software AG Designer.....	22
Mobile Designer.....	23
Mobile Support.....	23
Mobile Administrator.....	23
B2B Integration.....	25
B2B Integration.....	26
Building, Managing, and Monitoring a Trading Network.....	27
Trading Networks.....	27
eStandards Modules.....	28
Optimize for B2B.....	29
B2B Analytics.....	29
Managing File Transfers.....	30
Extending Your B2B Integrations into the Mobile Space.....	31

Software AG Designer.....	31
Mobile Designer.....	31
Mobile Support.....	31
Mobile Administrator.....	32
Business Process Management.....	33
Business Process Management.....	34
Designing Business Processes.....	34
ARIS Connect, ARIS Designer and ARIS Architect.....	35
Software AG Designer.....	36
Governing Business Process Models.....	37
Implementing Business Process Models.....	37
Software AG Designer.....	38
Developing Services.....	38
Developing Rules.....	39
Blaze.....	40
Developing Tasks.....	42
Task Engine.....	43
Executing Business Processes.....	43
Integration Server.....	43
Universal Messaging.....	44
My webMethods Server.....	44
Business Process Model Implementation Details.....	45
Configuring Process Steps.....	45
Creating Documents, Services, Tasks, and Rules.....	45
Developing Tasks Within a Process.....	46
Simulating and Debugging Business Processes.....	47
Logging and Monitoring Processes.....	47
Mapping Process Steps to Run Times.....	48
Developing in a Collaborative Environment.....	48
Developing Composite Applications.....	48
Software AG Designer.....	49
My webMethods Server.....	51
Integration Server.....	51
Developing in a Collaborative Environment.....	51
Administering and Monitoring Business Processes.....	51
Administering Business Processes.....	52
Monitoring Business Process Instances.....	52
Administering and Working with Business Processes and Tasks.....	53
Solving Problems with Closed Loop Analytics.....	53
Extending Your Business Processes into the Mobile Space.....	55
Software AG Designer.....	55
Mobile Designer.....	55
Mobile Support.....	55
Mobile Administrator.....	56

Master Data Management.....	57
Master Data Management.....	58
OneData.....	58
Intelligent Business Operations.....	63
Intelligent Business Operations.....	64
Business Event Processing.....	64
Business Activity Monitoring.....	65
Products Involved in Intelligent Business Operations.....	65
Intelligent Business Operations Example.....	65
Visualizing Data.....	66
Presto.....	66
Terracotta.....	67
Mining Process Data and Analyzing Root Causes.....	67
Process Performance Manager.....	67
Defining Baseline, Metrics, and Alerts.....	68
Optimize for Process.....	68
Terracotta.....	69
Detecting and Analyzing Complex Patterns in Real Time.....	69
Developing Apama Applications.....	70
Integrating Apama with External Data Sources.....	70
Building Dashboards.....	71
Managing and Monitoring Apama Platform Deployments.....	71
Event-Driven Architecture.....	71
Universal Messaging.....	71
Software AG Designer.....	71
Integration Server.....	72
Terracotta.....	72
Extending Your Intelligent Business Operations into the Mobile Space.....	72
Software AG Designer.....	72
Mobile Designer.....	73
Mobile Support.....	73
Mobile Administrator.....	73
SOA Governance and API Management.....	75
SOA Governance and API Management.....	76
Exposing and Consuming Services.....	76
Exposing and Consuming APIs.....	77
Governing Your SOA and API Management System.....	78
CentraSite.....	78
Mediator.....	79
Integration Server and Enterprise Gateway Server.....	80
Monitoring and Managing Your SOA or API Management System.....	80
Insight.....	80

About this Guide

This guide describes webMethods and Intelligent Business Operations products and how to use them to accomplish your goals.

Document Conventions

Convention	Description
Bold	Identifies elements on a screen.
Narrowfont	Identifies storage locations for services on webMethods Integration Server, using the convention <i>folder.subfolder:service</i> .
UPPERCASE	Identifies keyboard keys. Keys you must press simultaneously are joined with a plus sign (+).
<i>Italic</i>	Identifies variables for which you must supply values specific to your own situation or environment. Identifies new terms the first time they occur in the text.
Monospace font	Identifies text you must type or messages displayed by the system.
{ }	Indicates a set of choices from which you must choose one. Type only the information inside the curly braces. Do not type the { } symbols.
	Separates two mutually exclusive choices in a syntax line. Type one of these choices. Do not type the symbol.
[]	Indicates one or more options. Type only the information inside the square brackets. Do not type the [] symbols.
...	Indicates that you can type multiple options of the same type. Type only the information. Do not type the ellipsis (...).

Online Information

Software AG Documentation Website

You can find documentation on the Software AG Documentation website at <http://documentation.softwareag.com>. The site requires Empower credentials. If you do not have Empower credentials, you must use the TECHcommunity website.

Software AG Empower Product Support Website

You can find product information on the Software AG Empower Product Support website at <https://empower.softwareag.com>.

To submit feature/enhancement requests, get information about product availability, and download products, go to [Products](#).

To get information about fixes and to read early warnings, technical papers, and knowledge base articles, go to the [Knowledge Center](#).

Software AG TECHcommunity

You can find documentation and other technical information on the Software AG TECHcommunity website at <http://techcommunity.softwareag.com>. You can:

- Access product documentation, if you have TECHcommunity credentials. If you do not, you will need to register and specify "Documentation" as an area of interest.
- Access articles, code samples, demos, and tutorials.
- Use the online discussion forums, moderated by Software AG professionals, to ask questions, discuss best practices, and learn how other customers are using Software AG technology.
- Link to external websites that discuss open standards and web technology.

1 Application Integration

■ Application Integration	10
■ Developing Application Integrations	11
■ Executing Application Integrations	18
■ Application Integration Examples	20
■ Extending Your Application Integrations into the Mobile Space	22

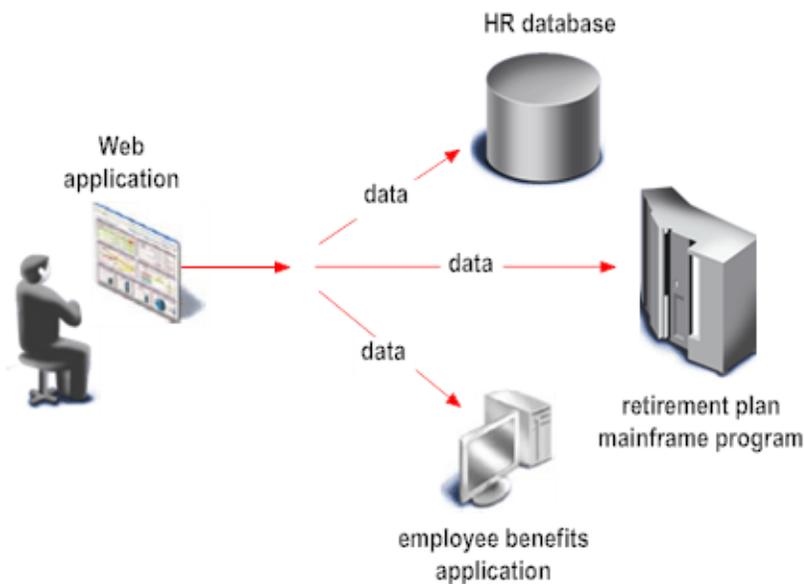
Application Integration

Application integration enables disparate resources to share business data. Resources include software applications such as SAP and Siebel, and systems such as databases and mainframe programs. Common application integrations include:

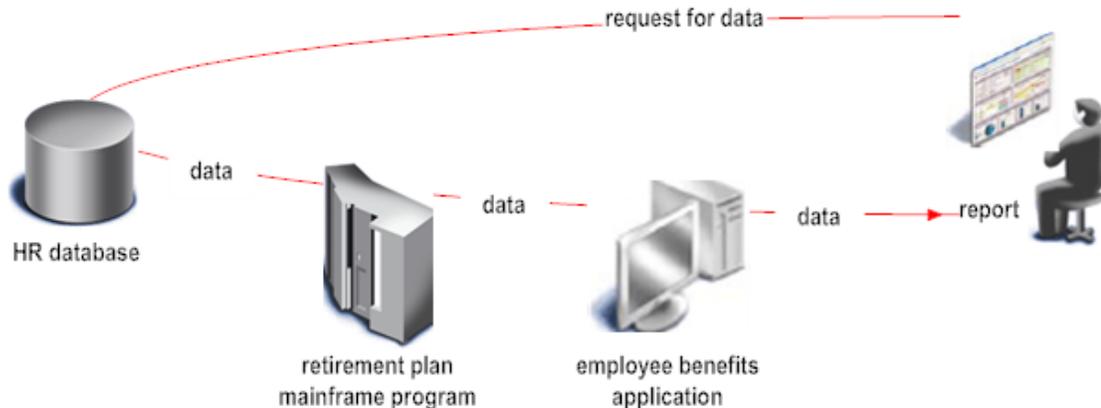
- **Synchronization.** For example, two stores in a chain of retail stores maintain customer, product SKU, and product price information. One store maintains the information in a database and the other in a mainframe program. When information is added to or changed in the resource at one store, an application integration updates the resource at the other store.



- **Propagation.** For example, a human resources (HR) person enters data for a new employee in a composite application and an application integration propagates the data to a human resources (HR) database, a retirement plan mainframe program, and an employee benefits application.



- **Composition.** For example, an HR person requests a report on an employee through a composite application. An application integration gathers the data for the report from an HR database, a retirement plan mainframe program, and an employee benefits application, and then returns the data to the web application for display in report format.



Developing Application Integrations

The primary products you use to develop application integrations are Software AG Designer, Application Platform, CloudStreams, adapters, Terracotta, EntireX, and ApplinX.

Software AG Designer

Software AG Designer is an Eclipse-based graphical development tool you use to design and test services, the primary elements in applications integrations. A service is logic that performs a unit of work. For example, a service could post a purchase order received from a customer to an ordering system, or perform a credit check for a loan application. You can develop these types of services:

- *Simple* services that perform one unit of work.
- *Aggregate* services, in which services call other services (for example, to propagate data from one resource to several other resources).
- *Composite* services, in which a service is wrapped around multiple simple or aggregate services that execute in sequential order (for example, to compose a report by gathering data from one resource after another). The wrapper service manages the flow of data from service to service.

You develop services in the Software AG Designer Service Development perspective, using Software AG *Flow* language or other languages such as Java. When working in Software AG Designer, you are always connected to an Integration Server; Software AG Designer builds and edits services directly on the Integration Server. Integration

Server comes with a library of built-in services that you can use in your application integrations, and executes your services at run time.

Service development is an iterative process of building, testing, and correcting (debugging) your code. Software AG Designer provides a range of tools to assist you during the testing and debugging phases. You can test services with input values you specify manually, inspect the results, and investigate errors. You can also set up audit logging for services and documents.

Multiple Software AG Designer users can collaborate on an application integration, developing different pieces of the application integration and then deploying the pieces to a single Integration Server for testing. Software AG Designer enables you to lock objects you are working with, and can interact with a third-party version control system (VCS) repository.

Another tool for collaboration is CentraSite, which operates as a shared database of metadata about assets that are stored in Software AG Designer, Integration Server, and CentraSite. Application integration developers can publish services and document types from Software AG Designer to CentraSite, and can drag and drop these assets from CentraSite into Software AG Designer. You can incorporate web services from CentraSite (and other SOA registries) into application integrations you build in Software AG Designer. Conversely, Software AG Designer can create web services from services that reside on Integration Server and can register the web services with CentraSite (and other SOA registries).

Application Platform

Application Platform enables you to develop application logic in Java without having to use proprietary webMethods APIs. You can use popular Java development models such as POJOs or Spring Beans for your application logic, and then deploy the logic to run in Integration Server. You develop the application logic using the Application Platform perspective in Software AG Designer. You can also develop web UI applications based on Java servlet technology.

You can integrate Application Platform services with Integration Server services. You can use simple Java annotations to expose Application Platform services within Integration Server and make those services callable from Flow services, adapter services, triggers, or web service connectors. You can create Java class wrappers for Integration Server services to make those services callable from Application Platform services without using proprietary Integration Server client APIs.

CloudStreams

CloudStreams is a multi-component product that enables you to develop and govern integration flows between *software as a service* (SaaS) providers such as Salesforce.com and on-premise applications such as CRM and ERP. CloudStreams includes a server, an analytics dashboard, and Software AG Designer perspectives.

The CloudStreams Server performs these functions:

- Runs on Integration Server and hosts *connectors*. Connectors contain assets that enable the CloudStreams Server to connect with and perform operations on SaaS providers. Software AG provides many connectors, including the Salesforce.com connector, but you can use the CloudStreams framework to create custom cloud connectors for other SaaS applications.
- Collects run-time performance metrics and events for your CloudStreams connectors and services. You can use the CloudStreams analytics dashboard, which is powered by MashZone, to monitor collected metrics and events.

For inbound processing (that is, from SaaS providers to on-premise applications), you use the Software AG Designer CloudStreams Development perspective to create virtual services and their governance policies. These services and policies perform user-defined security checks and processing, send requests to on-premise applications, and handle the responses.

For outbound processing (that is, from on-premise applications to SaaS providers), you use the Software AG Designer Service Development perspective to create cloud connector services that perform SOAP operations or access REST resources. You use the CloudStreams Development perspective to create a special kind of virtual service, called a connector virtual service, that sends requests to SaaS providers and handles the responses.

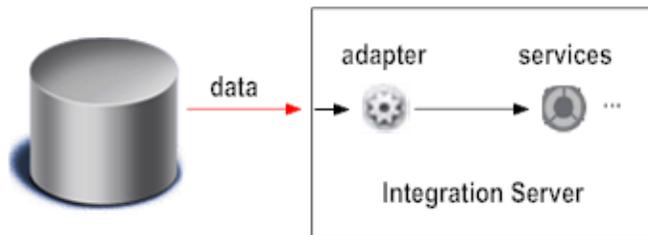
CloudStreams uses Terracotta BigMemory to cache the virtual services you deploy to the CloudStreams server. It also uses BigMemory to cache many of the artifacts associated with the virtual services, such as performance metrics and governance policies. When you install CloudStreams on a cluster of Integration Servers, BigMemory maintains these caches on the Terracotta Server Array.

Adapters

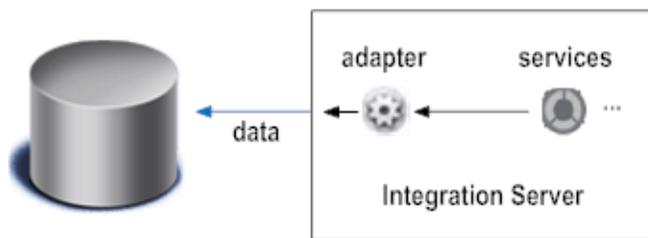
If an application integration must interact with an application for which Software AG provides an *adapter*, you can create services that invoke services in the adapter. Adapters are software components with browser-based interfaces that connect resources in your enterprise to Software AG products, and, through those products, to each other. While Integration Server supports a variety of standards such as XML, adapters support proprietary protocols for accessing packaged applications such as SAP, Siebel, JD Edwards EnterpriseOne, Oracle Applications, and PeopleSoft; databases such as Oracle, SQL Server, Informix, Sybase, and DB2; and mission-critical programs on mainframes and UNIX systems. Adapters transform data from resource-specific format into the format used by the Software AG product, and vice versa. They enable you to incorporate resources into application integrations without having to build complex custom code. Adapters run on Integration Server.

Adapters convey data from resources to Software AG products. Adapters can either actively poll resources for new or changed data or passively receive new or changed data from resources. For example, the Adapter for JDBC can receive data from a database, transform it from the database-specific format into the appropriate

Software AG internal format, and send the transformed data to services on Integration Server for further processing.



Adapters convey data from Software AG products to resources. For example, an Adapter for JDBC service can receive data from an Integration Server service, transform it from the Software AG internal format into the format required by the database, and insert it into the database.



EntireX

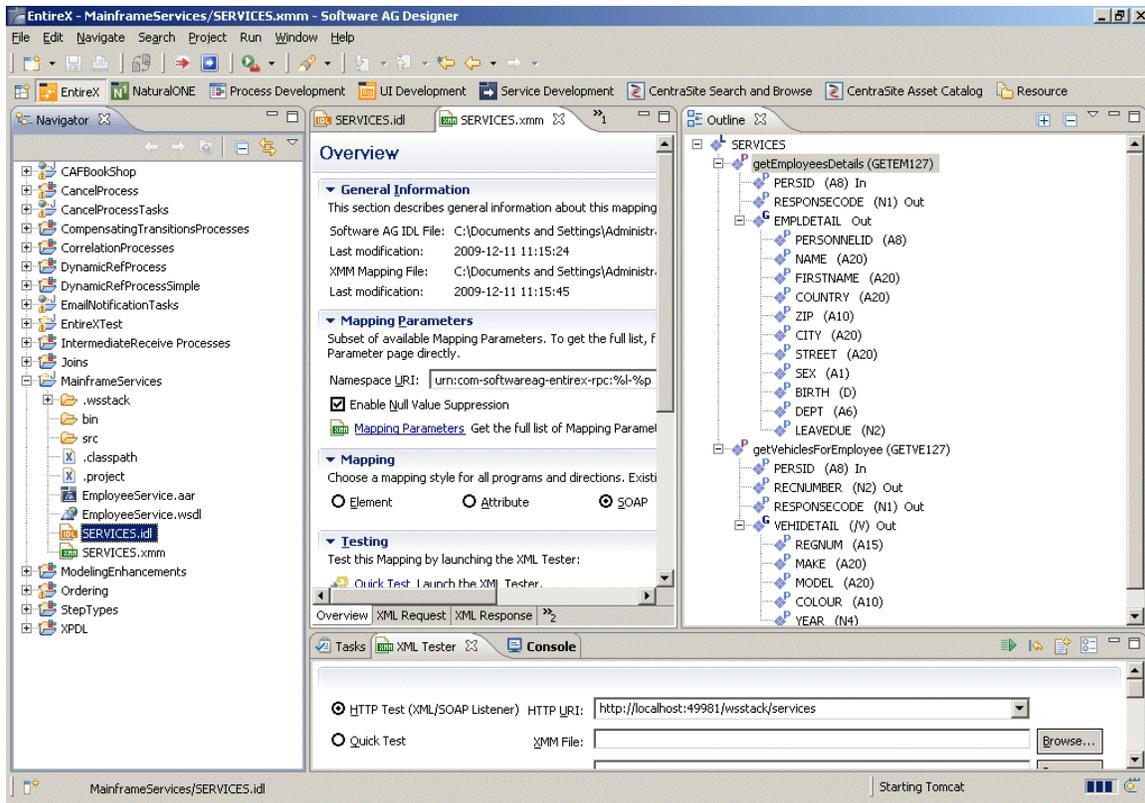
EntireX enables you to easily connect services that run on Integration Server to mission-critical programs written in languages like COBOL or Natural

The EntireX runtime environment includes these main components:

- The EntireX Adapter, which runs on Integration Server.
- The EntireX Broker, which supports load balancing, security, large messages, and high availability.
- Client or Server endpoints, like CICS RPC Server.
- The EntireX perspective in Software AG Designer, which enables you to generate the necessary assets, like adapter services on Integration Server.

Suppose you want to call a CICS transaction programmed in COBOL from Integration Server. One architectural variant to implement this synchronous request-reply scenario with EntireX is EntireX Adapter to EntireX Broker to CICS RPC Server. You would use the EntireX perspective in Software AG Designer to extract the transaction's interface from the COBOL source, copybooks, or both. EntireX wizards guide you through interface extraction and code generation of the adapter service on an Integration Server that hosts the EntireX Adapter. EntireX creates all technical assets needed, relieving you from having to work with technical details. You can directly test the adapter service in Software AG Designer.

For advanced use cases, however, the technical details are still accessible. The illustration below shows extracted signatures for two mainframe programs in the EntireX perspective.



When you run the adapter service on Integration Server, it invokes the CICS transaction on the mainframe side using values you provide for the transaction's extracted inputs. You can use the adapter service in application integrations as you would any other Integration Server service. Software AG Designer can create a web service from the adapter service and can register the web service in SOA registries such as CentraSite. You can then easily use the web services to include mainframe resources in business-to-business integrations (B2B) and business processes.

Since the EntireX architecture is symmetric, it allows for outbound as well as inbound connectivity. For example, suppose your CICS transaction wants to call an Integration Server flow service (that is, mainframe outbound). EntireX helps you create the necessary assets in the same manner.

You can extract the signatures of programs written in COBOL, PL/I, or Natural. On the mainframe, EntireX supports CICS, IMS, and batch environments.

ApplinX

ApplinX is a server-based technology that provides an efficient, robust, and easy way for web applications to access and integrate data and transactions from core system applications without changing those applications.

ApplinX exposes core system applications and encapsulates them in components such as web applications and web services using standard programmatic environments such as .NET or J2EE. You can then use these components as advanced building blocks in any modern development platform, for new or existing applications such as CRM applications. In this way, you can integrate core system applications into new strategic IT platforms, and can re-engineer your workflow using a more streamlined and efficient task-oriented and role-based approach.

ApplinX offers three types of solutions: web enablement, SOA enablement, and webpage integration. You use the ApplinX perspective in Software AG Designer for these types of development work. The illustration below shows the ApplinX perspective.

The screenshot displays the Software AG Designer interface in the ApplinX perspective. The main window is titled "Path Procedure" and shows a tree view of procedure logic. The logic includes a loop and several conditional steps:

```

[Count] New Object count, value:0
[Continue] New Object isContinue, value:true
[While_1] Loop While IsIsNot((While_1_Counter) > "10") And Is((isContinue) = "true") And ...
  [Condition_1] Test If Is((While_1_Counter) > "0")
    [S_3] Step (From: BrowseCustomers1, HostKey: "[pf0]")
    [Condition_1_True] Case True
    [Condition_1_False] Case False
    [Condition_2] Test If Is Content("End of data reached") in Field((/Message))
    [Condition_2_True] Case True
    [Mapper_2] Map Links: 1
    [Condition_2_False] Case False
    [Mapper_1] Map Links: 1
    [ScreenMapper_1] Map Screen (BrowseCustomers1 Links: 7)
    [S_2] Step (From: BrowseCustomers1, HostKey: "[pf11]")
  
```

On the left, the "ApplinX Explorer" shows a project structure with folders like "CompositeDemo", "Repository", and "customer", containing various components such as "BrowseCustomers1", "CreateCustomer1", and "CollectAllCustomers".

At the bottom, the "Screen Mapper" view shows a mapping between a screen and a data table. The screen "ScreenMapper_1 BrowseCustomers1" has fields like "Select_Customer_ID", "Select_Lastname", and "Action". These are mapped to the "BrowseCustomers1Table" which has fields like "Customer_ID", "Lastname", "Firstname", "Birthday", "Gender", and "Type". The "Out" panel on the right shows the resulting data structure with fields like "houseNumber", "gender", "street", "lastName", "city", "type", "country", "zipcode", "birthday", "customerID", and "firstName".

Web Enablement

Web enablement allows you to turn existing host 'green screens' into web interfaces. You can achieve a true web look and feel without touching existing applications or changing any code. ApplinX offers the web enablement solutions described below.

Instant Solution

The instant solution enables you to turn core system applications into modern-looking web applications with little or no coding. Simple configurations to the web applications can improve the general look and feel of the application; for example, you can design a template with top and side images and a company logo, and modify the style sheet using different fonts, colors, and so on. These changes enable the end user to access information more quickly and with improved visibility.

In Software AG Designer, you can use *screen groups*, which allow you to apply the same design to multiple screens. You can use transformations to convert host screen patterns into web components. Transformations can include formatting the screen's header area, message line, titles, and borders; transforming host function keys into hyperlinks, buttons, or images; converting input fields into GUI elements such as combo boxes, radio buttons, or check boxes; adding calendar components to date input fields; and removing unnecessary characters.

HTML Emulation Solution

ApplinX's thin client HTML emulation is available in .NET and J2EE environments. ApplinX can instantly turn a host terminal emulation into a web browser terminal emulation that provides host key and print support and maintains existing color schemes.

Composite Solution

The composite solution, available in .NET and J2EE environments, enables you to fully customize and extend your web applications. You can aggregate information from multiple core system applications into a single webpage. You can integrate legacy assets at the screen, transaction, or data level, and support various web environments using the ApplinX Base Objects API for customizing the web framework.

SOA Enablement

SOA enablement allows you to reuse core system application functionality and leverage the operational qualities of a system's transaction platform to new IT projects. You expose core system application functionality and data as web services at the screen or transaction level, and integrate with other environments that invoke web services.

In Software AG Designer, you can create procedures and use them to turn application functionality into web services. You can combine disparate data sources into web services using ApplinX entities such as screens, programs, databases, and external web services. ApplinX supports web service standards such as SOAP and WSDL, and JAXR for registering services in SOA registries such as CentraSite.

Webpage Integration

Applinx offers a webpage integration solution that enables you to simulate web browser activity within Applinx and expose it as a standard web service or integrate it with Applinx procedures. Specifically, you use the simulation to record web content and create an Applinx Web Procedure. The web procedure is designed to enable integration of webpage user interfaces within Applinx. You can expose the web procedure as a service in the same way you can expose Applinx procedures such as the Path and Flow Procedures.

Executing Application Integrations

The products that execute application integrations include Integration Server and Universal Messaging.

Integration Server

Integration Server is a run-time server with a browser-based user interface whose primary function in application integrations is to execute services. Integration Server does the following:

1. Receives requests from client applications and authenticates and authorizes the requesting users.
2. Invokes the appropriate services and passes them input data from the requesting clients.
3. Receives output data from the services and returns it to the clients.

Integration Server supports a wide range of established and emerging standards so you can interact with virtually any business partner that is connected to the Internet.

Integration Server supports...	Such as...
Transport standards	HTTP, HTTPS, FTP, FTPS, SMTP
Message formats	MIME, S/MIME
Data standards	XML and XML Schema, custom flat file formats with delimited fixed- or variable-length records, JSON
Protocols	JMS, OData, SFTP, SOAP, SOCKS, XML RPC
Specifications	Web Service Description Language (WSDL)

Integration Server supports...	Such as...
Integration patterns	REST, web services, event-driven, request-reply
Authorization framework	OAuth

Integration Server hosts packages that contain services and related files, such as specifications, document types, and DSPs. Integration Server comes with many packages, and developers can create their own packages to hold services they create.

Your Integration Server might sit behind an internal firewall and not be allowed to directly interact with external clients through the DMZ. In this case, you can use Enterprise Gateway to allow the Integration Server to process requests from external clients. In an Enterprise Gateway configuration, you place another Integration Server, called an Enterprise Gateway Server, in your DMZ. The Enterprise Gateway Server acts as an intermediary between external clients and the internal Integration Server, protecting the internal Integration Server and its applications, services, and data from malicious attacks.

You can improve service execution performance by using Terracotta BigMemory to cache data that services use frequently. You can create large in-memory caches using Terracotta BigMemory Max or multi-terabyte shared caches using a Terracotta Server Array. When you cluster Integration Servers using Terracotta, the members of the cluster share data in caches on the Terracotta Server Array. Each Integration Server in the cluster connects to the Terracotta Server Array to store and retrieve items from the shared caches.

When acting as an OAuth authorization server, Integration Server maintains registered clients, scope definitions, access tokens, and refresh tokens in Terracotta BigMemory Max. When running Integration Server in a clustered environment, BigMemory maintains these caches on the Terracotta Server Array.

Universal Messaging

Universal Messaging is fast, reliable, scalable, and flexible Java message-oriented middleware (MOM) that provides messaging functionality such as clustering, scheduling, and interface plug-ins, with standard support for the messaging paradigms of publish/subscribe, message queues, and P2P, as well as support for JMS.

Universal Messaging serves as the intermediary that routes data from publishers to subscribers. It can function as the messaging backbone for resources across your enterprise and can support an event-driven architecture, in which subscribers receive documents and then perform predefined actions. It supports the publishing of documents in Integration Server's native IData format, and Integration Server can encode/decode those documents as protocol buffers.

Application Integration Examples

You can implement application integrations in a wide variety of ways. Below are some examples.

Synchronization

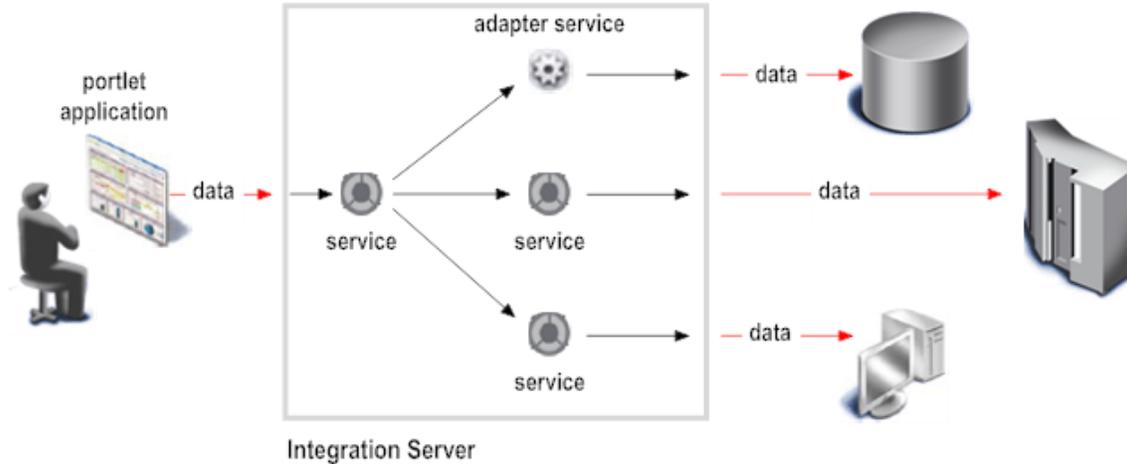
Suppose you want to synchronize customer information between Siebel and a mainframe program. You set up the Siebel Adapter to interact with the Siebel application and the EntireX Adapter to interact with the mainframe program. Since EntireX architecture is symmetric, you can use push or pull methods.

To get data from Siebel to the mainframe program, you have the Siebel Adapter monitor the Siebel application for new or changed data. The adapter transforms the data from Siebel's proprietary format into the appropriate Software AG internal format and passes it to Integration Server. Integration Server processes the data and then passes it to the EntireX Adapter, which transforms it into the mainframe program's format and inserts it into the system. To get data from the mainframe program to Siebel, you use the same process, in reverse.

Propagation

Suppose you want HR people to change employee information using a composite application, and you want to propagate the changes to the HR database, the employee benefits application, and the retirement plan mainframe program. You develop the following:

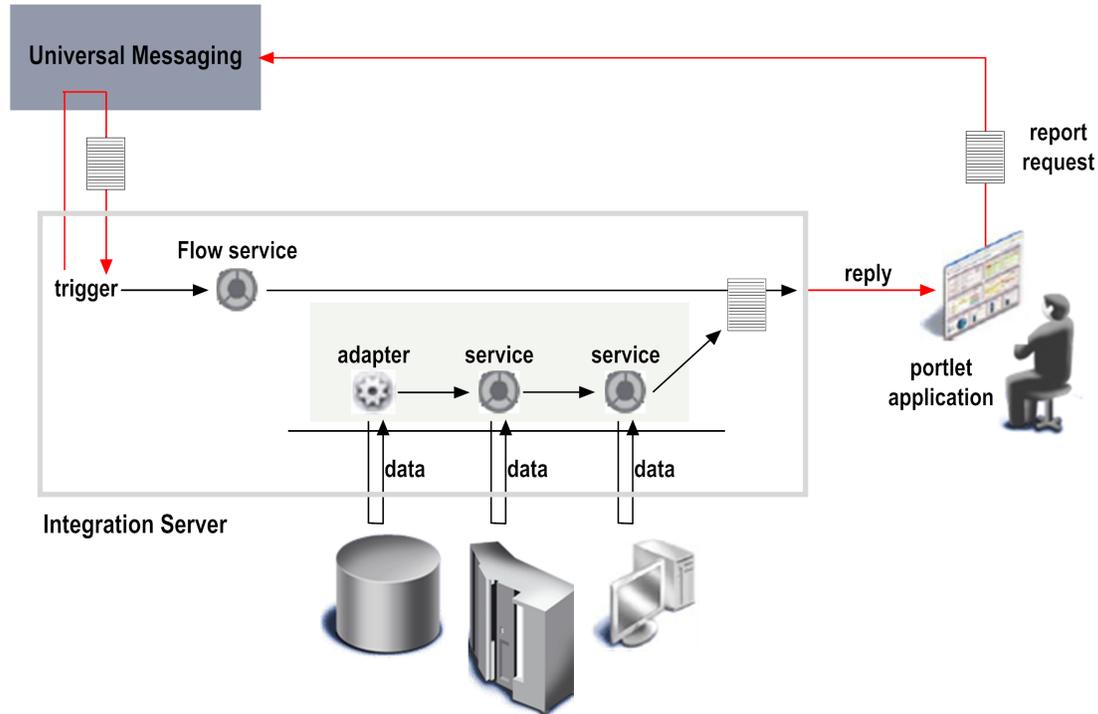
- A composite application in Software AG Designer that provides a user interface for entering employee information, and passes data entered on the form to Integration Server for processing.
- A synchronous service that receives the data from Integration Server, simultaneously invokes the three services described below, waits for confirmation from the three services, and then returns the confirmation to the composite application to display to the user.
- An Adapter for JDBC service and two other services that insert the data into the database, mainframe program, and benefits application, respectively, and return confirmations to the synchronous service described above.



Composition

Suppose HR wants to create reports containing employee information from the HR database, the retirement plan mainframe program, and the employee benefits application. You want to use Universal Messaging and its publish/subscribe model to route the information. You develop the following:

- A composite application in Software AG Designer that provides a user interface for requesting the report, publishes the request to Universal Messaging, and displays the returned data in report form to the user.
- A synchronous Adapter for JDBC service to extract data from the database.
- Two synchronous services that extract data from the mainframe program and the benefits application.
- An asynchronous Flow service that wraps the three services described above and maps the data from service to service.
- A *trigger* on Integration Server that subscribes to the report request on Universal Messaging and invokes the Flow service when Integration Server receives the request.



Extending Your Application Integrations into the Mobile Space

The Mobile Suite enables you to extend your integrations and business processes into the mobile space; with these products, users anywhere in the world can interact with integrations and business processes from their mobile devices. The Mobile Suite fully supports Mobile Enterprise Application Platform (MEAP) requirements and spans the full mobile lifecycle, from developing and deploying mobile apps to securely integrating and managing them. The Mobile Suite can also bring existing IT services and processes to mobile devices in an efficient way.

The products you use to support mobile devices are Software AG Designer, Mobile Designer, Mobile Support, and Mobile Administrator.

Software AG Designer

You can develop mobile apps using the Mobile Development perspective in Software AG Designer. Mobile Development provides logic that reduces the amount of code you must write to develop a mobile app.

Mobile Development uses the principles of model-view-controller (MVC) architecture, which separates user interfaces from business logic and data. With Mobile Development, mobile app developers design a user interface and then generate Java code that displays the user interface and responds to user-initiated events, such as the clicking of a button.

The developers code business logic separately from the user interface, in Java classes created by Mobile Development.

Mobile Designer

Mobile Designer is a mobile application development tool. With Mobile Designer, mobile app developers create mobile apps in Java and then automatically convert the apps for any mobile device, from an iPad to an Android to a Windows Phone. The converted apps provide the user experience that each device owner expects. Mobile Designer reduces the need for in-house mobile knowledge, which in turn reduces development and maintenance costs.

Mobile Support

Mobile Support is a set of tools that enable mobile app developers and business integration developers to create mobile data synchronization solutions. Such solutions transfer data between mobile devices and backend enterprise applications and resolve conflicts that occur when backend data is updated by multiple sources at the same time.

Mobile Support includes two components:

- Mobile Support Client provides a Java client library of classes composed of methods that initiate requests to synchronize data. Mobile app developers incorporate the synchronization methods in their apps.
- Mobile Support package runs on Integration Server and provides a service that synchronizes data sent by mobile devices. Business integration developers incorporate the synchronization service in their integration services.

Mobile apps submit synchronization requests to Mobile Support Client, which asks Integration Server to execute the services that process the requests. Mobile apps can submit either of the following synchronization requests:

- Download changed data from the backend application to the mobile device.
- Upload changed data from the mobile device to the backend application, synchronize that data with the backend data, and download changed data from the backend application to the mobile device.

Mobile Administrator

Mobile Administrator enables you to do the following:

- Manage mobile applications for multiple platforms using an enterprise app store to distribute apps to mobile devices based on the access rights of the user.
- Build Mobile Designer or native apps in a distributed, cloud-enabled build and test environment.
- Monitor your runtime data, events, and mobile app behavior (for example, when the app started, crashed, closed, and so on).

- Add application policies to and enable push notifications for mobile apps on the fly.
- Manage devices, including device inventory, device policies, and access rights.

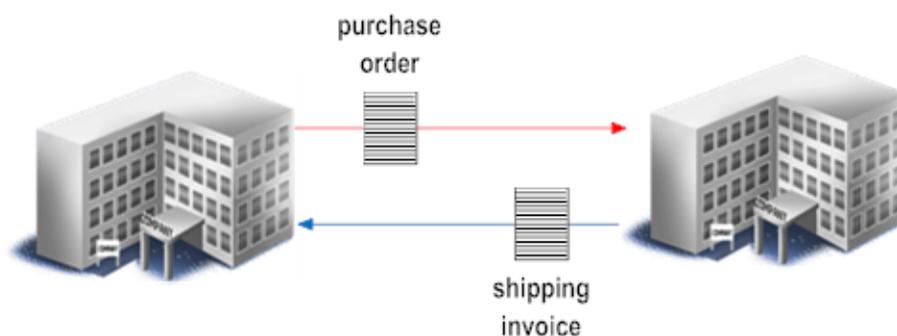
2 B2B Integration

■ B2B Integration	26
■ Building, Managing, and Monitoring a Trading Network	27
■ Managing File Transfers	30
■ Extending Your B2B Integrations into the Mobile Space	31

B2B Integration

Business to business (B2B) describes electronic transactions between businesses (as opposed to between businesses and consumers). Businesses that engage in electronic transactions are called *trading partners*, and can include retailers, manufacturers, suppliers, and marketplaces.

Transactions between trading partners usually involve the exchange of business documents using automated processes. A B2B integration network, or *trading network*, consists of a set of trading partners that conduct business by exchanging mutually agreed-upon business document types electronically. For example, a trading network might include computer retailers, a computer manufacturer, and computer parts suppliers. The retailers might send purchase order documents to the manufacturer, which returns purchase order acknowledgment, shipping notice, and invoice documents. Similarly, the manufacturer might send purchase order documents to the parts suppliers, and so on.



Some industries have developed B2B standards for exchanging business documents. For example, many manufacturing companies use the EDI messaging standard to conduct business electronically. A B2B standard typically defines the business document types and transport protocols that trading partners need to use in exchanges, and specifies document exchange rules.

B2B can include managed file transfer (MFT), a process that ensures protected internal and external data transfers in a centralized system. MFT provides reliable, secure data transfer you can automate based on policies, partners, and permissions. MFT also provides insight and control at every stage of the transfer process, including real-time monitoring, error and receipt logging, auditing, and data tracking. Most MFT solutions include these key components:

- MFT servers that perform MFT exchange behind a firewall and support all communications and security protocols.
- Proxies/reverse proxies that operate in the demilitarized zone and protect the IP addresses and ports of file senders and receivers.
- Clients that enable both human users and applications to administer, report, schedule, and script file transfers.

- APIs that third-party applications use to interact and communicate with MFT servers.

Building, Managing, and Monitoring a Trading Network

The primary products you use to build, manage, and monitor a trading network are Trading Networks, eStandards Modules, and Optimize for B2B.

Trading Networks

Trading Networks enables you to build and manage a peer-to-peer or hub-and-spoke network of trading partners. Trading Networks enables trading partners to exchange business documents in XML and structured flat-file formats.

Trading Networks offers a browser-based user interface in which you build the elements you need to define and link trading partners. The illustration below shows a partner profile definition in this user interface.

The screenshot displays the 'My webMethods Administrator' interface in a browser window. The main content area is titled 'Partner Profiles > Partner Profile'. Below this, there is a 'Partner Details' section with a form containing the following fields:

- Corporation Name: Receiver
- Organization Unit: ABC
- Status: Active (dropdown menu)
- Preferred Language: Root Locale (dropdown menu)

Below the form are several tabs: External IDs, Addresses, Contacts, Partner Group Membership, Delivery Settings, and Extended Fields. The 'External IDs' tab is active, showing a table with one entry:

Id Type	Value
DUNS	9088877320000

At the bottom of the interface, there is a footer with the text 'All times IST (UTC+5:30)' and 'powered by webMethods'.

Elements you need to define and link trading partners are as follows:

Element	Purpose
Trading partner definitions	Define and administer your trading partners.
Business document types	Define the business documents that trading partners want to exchange. A business document type can define an industry-standard document, such as an EDI, RosettaNet, cXML, CBL, or OAG document, or a custom business document.
Processing rules	Define how to process business documents. For example, the processing rule for a purchase order you receive from a trading partner might verify the sender's signature and then submit it to your order management system.
	<p>Note: If you need more complicated processing than is possible in the Trading Networks user interface, you can design a business process in Designer to use in addition to or instead of a processing rule.</p>
Trading Partner Agreements (TPAs)	Customize the way in which documents are exchanged between trading partners. For example, you and a trading partner might use a TPA to specify a custom ID field to include in all business documents.

Trading Networks provides templates that enable you to easily create onboarding questionnaires. Partners can respond to onboarding campaigns by replying to email invitations or by accessing self-service portals. Trading Networks administrators can easily generate reminder emails to partners that have not responded.

Trading Networks also includes a component that runs on Integration Server and manages the exchange of business documents among trading partners. When Trading Networks receives a document, it processes the document according to the processing rule for that document type.

Trading Networks uses Ehcache to cache profiles, document types and attributes, and trading partner agreements.

eStandards Modules

Software AG supports B2B standards for various industries. This support comes in the form of eStandards Modules that run on Integration Server and usually require Trading Networks. Each eStandards Module defines the industry-standard or proprietary transport protocol, provides the B2B standard's business document types, and specifies the standard's document exchange rules. When Integration Server receives a document

that matches a business document type in an eStandards Module, it processes the document according to the document exchanges rules specified in the module.

Software AG provides eStandards Modules for the industries below.

Industry	eStandards Modules
Manufacturing	RosettaNet, Chem, PIDX, papiNet
Financial services	SWIFT, ACH, FIX
Consumer packaged goods and retail	1SYNC, ebXML
Healthcare	HIPAA, HL7
E-commerce	EDI, EDIINT, AS4

Some eStandards Modules uses Terracotta BigMemory to create system caches in which to store transaction-related data and configuration information related to the use of the module. When you use these modules in a Terracotta clustered environment, BigMemory maintains their system caches on the Terracotta Server Array.

Optimize for B2B

Optimize for B2B enables business administrators to monitor Trading Networks transactions in real-time. You use Optimize for B2B to:

- Identify business document types and attributes to monitor.
- Define KPIs to monitor for transactions.
- Monitor transactions and other data related to exchanging data with trading partners.
- Compare the performance of different KPIs and analyze historical KPI performance to find positive or negative trends.

B2B Analytics

Trading Networks provides a variety of charts and dashboards for performing deep targeted analysis on Trading Networks partner transactions.

Some charts and dashboards provide a snapshot of current transaction volume trends that indicate top-tier and bottom-tier partners based on business activity. You can drill down into the charts to gain additional insight into the underlying data. You can export the data in any of these charts and dashboards to CSV files.

Other dashboards and charts to monitor a variety of metrics either across all partners or by partner. The metrics include transaction volume trends; total value trends based

on purchase order amount, invoice amount, and so on; late functional acknowledgment (FA) violations; and the split between successful and failed transactions. You can filter the information shown in the charts to focus on a subset of partners (top-tier, bottom-tier, and so on).

Managing File Transfers

The product you use to manage the transfer of business documents in your trading network is ActiveTransfer.

ActiveTransfer offers a browser-based user interface that enables administrators to manage file transfers, servers, and users; browse and search audit logs of file transfers; and examine analytics that show metrics, comparisons, and file upload and download activity. The interface also enables you to configure actions that are triggered based on criteria you specify. For example, you could set up an action to execute a specific Trading Networks service if a file transfer is successful. ActiveTransfer also offers a web client for end users who need to upload and download files to and from ActiveTransfer Server.

ActiveTransfer Server resides behind a firewall and performs file transfers. ActiveTransfer Gateway is a reverse proxy server that acts as an intermediary between the Internet and the internal ActiveTransfer Server for secure file transfer.

ActiveTransfer Support package runs on Integration Server and provides functionality for monitoring file transactions and managing ActiveTransfer Server, ActiveTransfer Gateway, users, post-processing events, scheduled actions, and data files and folders. You invoke this functionality through the ActiveTransfer user interface.

ActiveTransfer uses virtual folders for file transfers. You can organize these virtual folders by partner name, and you can associate ActiveTransfer users with partners. If you are using ActiveTransfer with Trading Networks, ActiveTransfer Server retrieves the list of partners from Trading Networks. If you are not using ActiveTransfer with Trading Networks, you can define your own partners.

You can use virtual folder definitions in scheduled and post-processing events instead of using the direct IP address for remote endpoints involved with file transfers. In this way, you can manage all endpoints centrally through virtual folders. If you change an endpoint address, you need only edit it once in the virtual folder definition and the change will be automatically reflected in all events that access that endpoint.

ActiveTransfer supports HTTP, HTTPS, FTP, FTPS (SSL), SFTP (SSH), SCP, WebDAV, and WebDAV's protocols, and offers complete data security and support for the most stringent encryption standards, including SSL and integrated PGP. ActiveTransfer enables accelerated file transfers that use a server's bandwidth to the maximum capacity regardless of network latency or distance.

Extending Your B2B Integrations into the Mobile Space

The Mobile Suite enables you to extend your integrations and business processes into the mobile space; with these products, users anywhere in the world can interact with integrations and business processes from their mobile devices. The Mobile Suite fully supports Mobile Enterprise Application Platform (MEAP) requirements and spans the full mobile lifecycle, from developing and deploying mobile apps to securely integrating and managing them. The Mobile Suite can also bring existing IT services and processes to mobile devices in an efficient way.

The products you use to support mobile devices are Software AG Designer, Mobile Designer, Mobile Support, and Mobile Administrator.

Software AG Designer

You can develop mobile apps using the Mobile Development perspective in Software AG Designer. Mobile Development provides logic that reduces the amount of code you must write to develop a mobile app.

Mobile Development uses the principles of model-view-controller (MVC) architecture, which separates user interfaces from business logic and data. With Mobile Development, mobile app developers design a user interface and then generate Java code that displays the user interface and responds to user-initiated events, such as the clicking of a button. The developers code business logic separately from the user interface, in Java classes created by Mobile Development.

Mobile Designer

Mobile Designer is a mobile application development tool. With Mobile Designer, mobile app developers create mobile apps in Java and then automatically convert the apps for any mobile device, from an iPad to an Android to a Windows Phone. The converted apps provide the user experience that each device owner expects. Mobile Designer reduces the need for in-house mobile knowledge, which in turn reduces development and maintenance costs.

Mobile Support

Mobile Support is a set of tools that enable mobile app developers and business integration developers to create mobile data synchronization solutions. Such solutions transfer data between mobile devices and backend enterprise applications and resolve conflicts that occur when backend data is updated by multiple sources at the same time.

Mobile Support includes two components:

- Mobile Support Client provides a Java client library of classes composed of methods that initiate requests to synchronize data. Mobile app developers incorporate the synchronization methods in their apps.
- Mobile Support package runs on Integration Server and provides a service that synchronizes data sent by mobile devices. Business integration developers incorporate the synchronization service in their integration services.

Mobile apps submit synchronization requests to Mobile Support Client, which asks Integration Server to execute the services that process the requests. Mobile apps can submit either of the following synchronization requests:

- Download changed data from the backend application to the mobile device.
- Upload changed data from the mobile device to the backend application, synchronize that data with the backend data, and download changed data from the backend application to the mobile device.

Mobile Administrator

Mobile Administrator enables you to do the following:

- Manage mobile applications for multiple platforms using an enterprise app store to distribute apps to mobile devices based on the access rights of the user.
- Build Mobile Designer or native apps in a distributed, cloud-enabled build and test environment.
- Monitor your runtime data, events, and mobile app behavior (for example, when the app started, crashed, closed, and so on).
- Add application policies to and enable push notifications for mobile apps on the fly.
- Manage devices, including device inventory, device policies, and access rights.

3 Business Process Management

■ Business Process Management	34
■ Designing Business Processes	34
■ Governing Business Process Models	37
■ Implementing Business Process Models	37
■ Executing Business Processes	43
■ Business Process Model Implementation Details	45
■ Developing Composite Applications	48
■ Administering and Monitoring Business Processes	51
■ Extending Your Business Processes into the Mobile Space	55

Business Process Management

A business process is a series of business activities that are performed in a specific order by a variety of applications, systems, employees, and external businesses. Examples of business processes include handling a purchase order from receipt through fulfillment, taking a product from inception to market, and preparing for a new employee. Business processes are more complex and long-running than application integrations and can include activities performed by humans.

Business process management enables you to automate business processes. For example, the process of preparing for a new employee could be automated as follows:

1. The hiring manager submits an online form that contains information about the new employee.
2. The submission of the form triggers the first step in the process. This step adds the employee's information to the internal human resources (HR) database and registers the employee in various systems and applications, such as the enterprise's e-mail application.
3. The next step in the process uses the enterprise's trading network to notify the external payroll company to set up an account for the employee.
4. The next step sends the facilities department a task to assign office space and provide office equipment before the employee's start date.

Business processes typically involve many variables and conditions, and the longer they run, the more likely the variables and conditions are to change. For example, a supplier might temporarily run out of parts needed to fill orders. Business process management enables you to act on running processes in response to such changes; in the example above, you could suspend order fulfillment processes until parts are available again.

Designing Business Processes

Experts in company procedures create *business process models* that are the designs for business processes. These business process designers define the steps in business processes and dictate the order of the steps and the conditions under which each step should run.

The products that business process designers use to create business process models are ARIS Connect Designer, ARIS Designer or ARIS Architect, and Software AG Designer. The ARIS products offer a wide range of graphical representation tools that enable the designer to focus on the high-level design of a business process without having to get involved in developing the underlying technology. The designer then passes the model to business process developers to implement in Software AG Designer (*model-to-execute*).

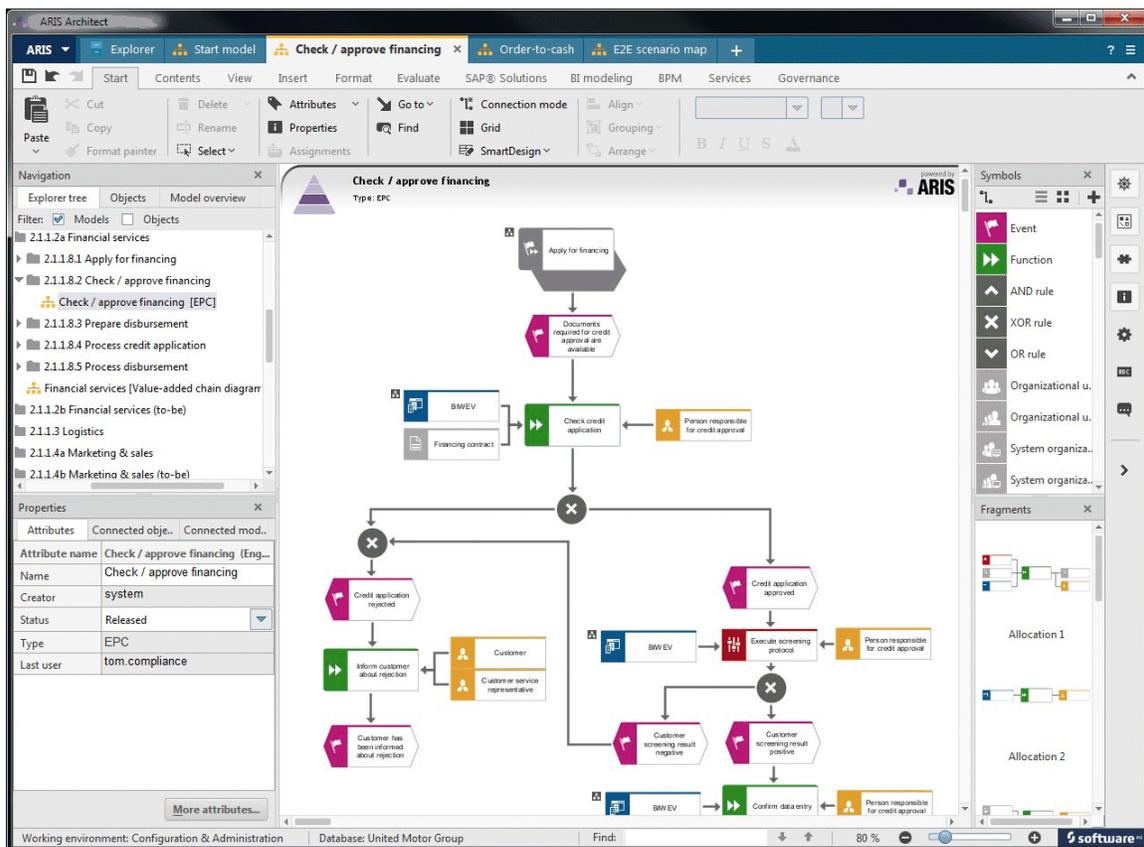
ARIS Connect, ARIS Designer and ARIS Architect

ARIS Connect is a design tool based on HTML5 technology that allows business process designers to design, and optionally collaborate on the design of, business process models.

ARIS Designer is a Java client that enables business analysts and process owners to design and analyze more complex business process models. ARIS Designer offers greater functionality than ARIS Connect, such as the ability to run analyses that identify bottlenecks in business process.

ARIS Architect is ARIS Designer with administrative features, such as database and functional ARIS application management.

The illustration below shows a business process model in ARIS Designer.



Business process models are stored on a central server, and all ARIS Connect, ARIS Designer, and ARIS Architect users connect to this central server. Configuration settings supplied by ARIS Architect users are also stored on the central server, and are propagated to all ARIS Connect, ARIS Designer, and ARIS Architect clients.

After a business process model has been designed, ARIS users can do the following:

- ARIS Connect users can publish the model to the rest of the company. ARIS Publisher, a set of JavaServer Pages that display in an Internet browser, can also

be used to publish models. Anyone in your company who has viewed a published business process model can ask a business process designer to add, modify, or delete a process step to the model.

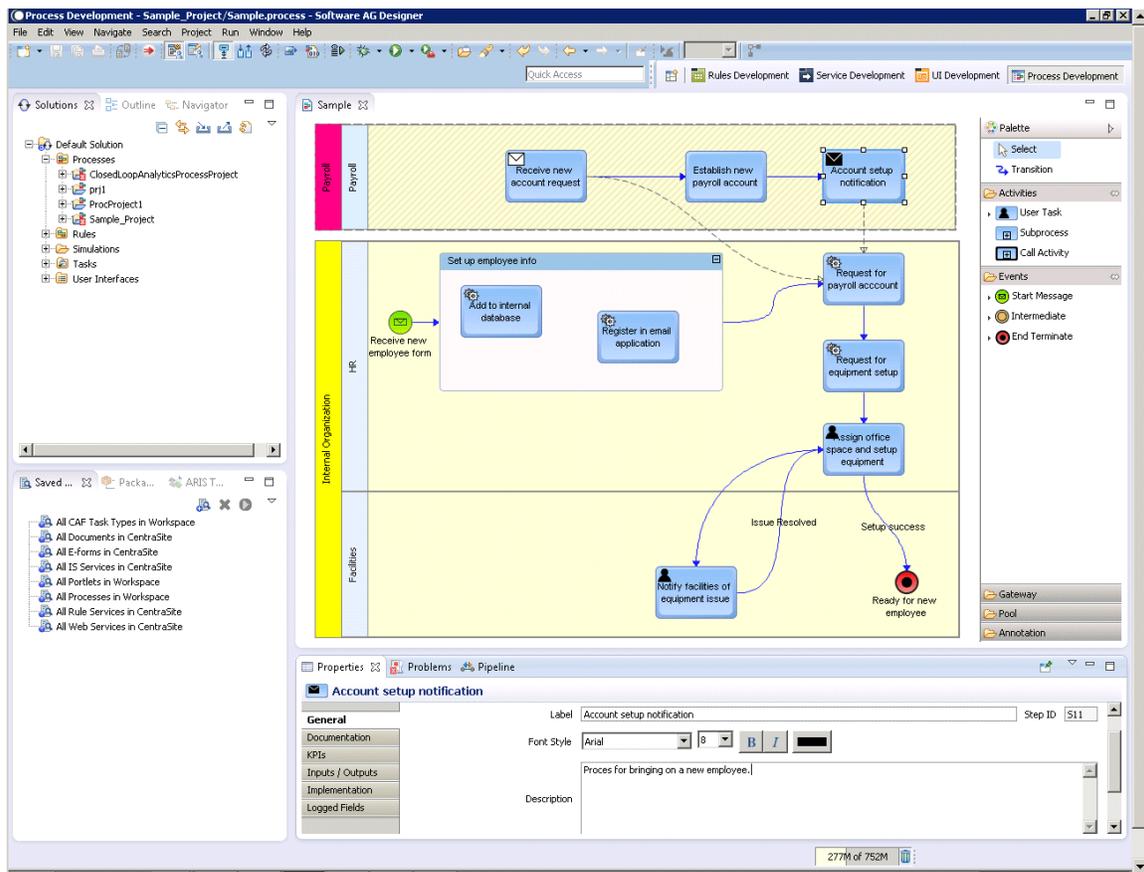
- ARIS Architect users can convert the business process into Business Process Modeling Notation (BPMN), a graphical representation for specifying business process models and the functionality and behavior of process steps, and then share the model with Software AG Designer business process developers.
- ARIS Architect users can share data models, user interface designs, and service information with Software AG Designer business process developers.

Software AG Designer

Software AG Designer is an Eclipse-based graphical development tool that offers *perspectives* for different types of work. The Business Analyst capability of the Software AG Designer Process Development perspective enables business analysts to document the requirements of the business process. This documentation helps guide the business process developers who implement the business process model. For example, documentation can:

- Specify the input that each process step requires and the output that each process step should produce.
- Identify humans who are involved in the process (for example, a facilities manager, to set up office space).
- Specify how to handle errors.
- List KPIs to track to measure the effectiveness of the process (for example, the period of time it took to set up the office space).

The illustration below shows a business process model in the Business Analyst capability in Software AG Designer.



Governing Business Process Models

ARIS business process governance lets you control the lifecycle of business process models using standards, policies, and guidelines via out-of-the-box workflows. You can define approval and release workflows that govern aspects of BPM such as business process model change requests, approvals, and release. Workflows can include tasks that send notifications to users who need to perform the actions in each step of the approval and release workflow, user interfaces for tasks, and task-to-task data maps.

Automated tasks can call services provided with your ARIS products. The services include ARIS business services for operations on the ARIS server and ARIS document storage services for document operations and user management services for operation on UMC.

Implementing Business Process Models

The primary product that business process developers use to implement business process models is Software AG Designer. Developers might also use Blaze Rules Development, CentraSite and Task Engine.

Software AG Designer

Business process developers use the Software AG Designer Process Development perspective to implement business process models.

Business process developers implement a business process model in Software AG Designer by dragging and dropping graphical representations of process steps onto a design canvas, then configuring each step to perform a specific function. For example, process steps can:

- Receive data. Data can be in the form of documents from Software AG products, JMS messages, EDA events, and output data from web services and services such as database query services.
- Invoke services such as Flow services, web services, rule services, and adapter services that in turn invoke programs on mainframes and UNIX systems.
- Invoke rules.
- Invoke other business processes.
- Publish data for other business processes and services to consume.
- Publish EDA events.
- Assign activities, called *tasks*, to a human or group of humans to perform. Tasks in turn can invoke rules, rule sets, and services.

Business process developers define the way data passes through and is handled by your business processes. They define the order of steps in the processes, and the conditions under which the steps start, run, pass data, and end. Business Process Modeling Notation (BPMN) provides a graphical representation of the underlying model and of the functionality and behavior of its steps.

Business process developers can also define milestones within a business process model by defining stages. A stage spans process steps, has a start milestone step, and has a complete milestone step that must be reached within a specified period of time. For example, a stage might be defined as starting at the Receive Purchase Order step (start milestone), completing at the Ship Product step (complete milestone), and transpiring within three days.

Developing Services

Developers can develop services such as Flow, web, and adapter services using the Software AG Designer Service Development perspective.

After you develop the services, you can drag and drop them onto the business process model design canvas. In a collaborative design environment, you can also drag and drop assets from CentraSite. CentraSite operates as a shared SOA registry/repository of metadata about assets that were developed in Software AG Designer and that are stored on run-time servers such as Integration Server.

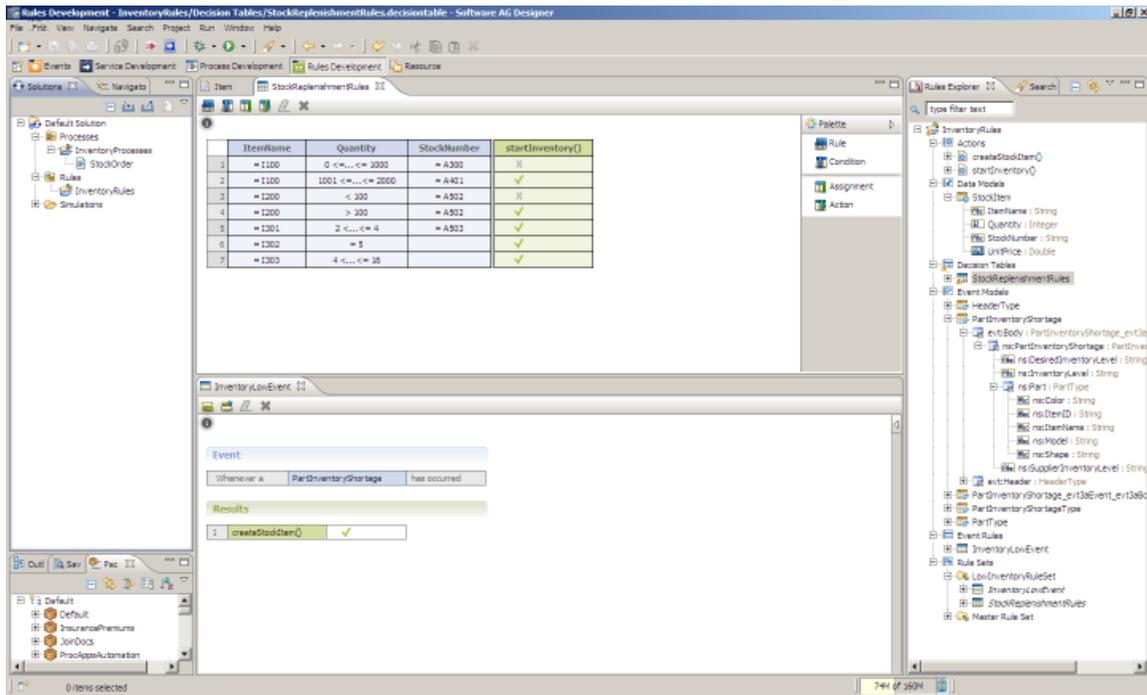
Developing Rules

Developers can develop business rules using the Software AG Designer Business Rules Development perspective. Rules can be expressed as decision tables or event rules. Rules can invoke services. Rules can operate on processes (for example, rules can start or stop, or suspend or resume processes) and tasks (for example, rules can assign tasks to users). Rule expressions can contain function calls, literals, parameter references, the mathematical operators + - * /, and groups of parentheses.

A decision table is made up of rows and columns, and each row defines a rule. A rule includes one or more condition columns and one or more result columns. For example, a decision table for health care insurance premiums could include a rule that specifies condition columns for gender, age, and smoker/non-smoker, and a result column that contains premiums to be paid.

An event rule consists of an event and one or more results. A result can assign a value or execute an action. For example, an event rule for an automobile parts distributor could specify that when inventory on parts decreases to certain levels, the event rule creates a data action. This data action creates data that is evaluated by rules in a decision table. Depending on the content of the new data, one of the rules in the decision table fires. Some of the rules launch a business process to replenish the stock in a result column.

You can group multiple decision tables and event rules into a *rule set*, and invoke the rule set from a process step or task. The rules in a rule set interact with each other, so that the conclusion drawn from one rule (the result) is used as input information (the condition) for a second rule. This is called *forward chaining*. In the example above for the automobile parts distributor, the event rule and the decision table could be grouped into a rule set. The illustration below shows the decision table and event rule in the rule set named LowInventoryRuleSet.



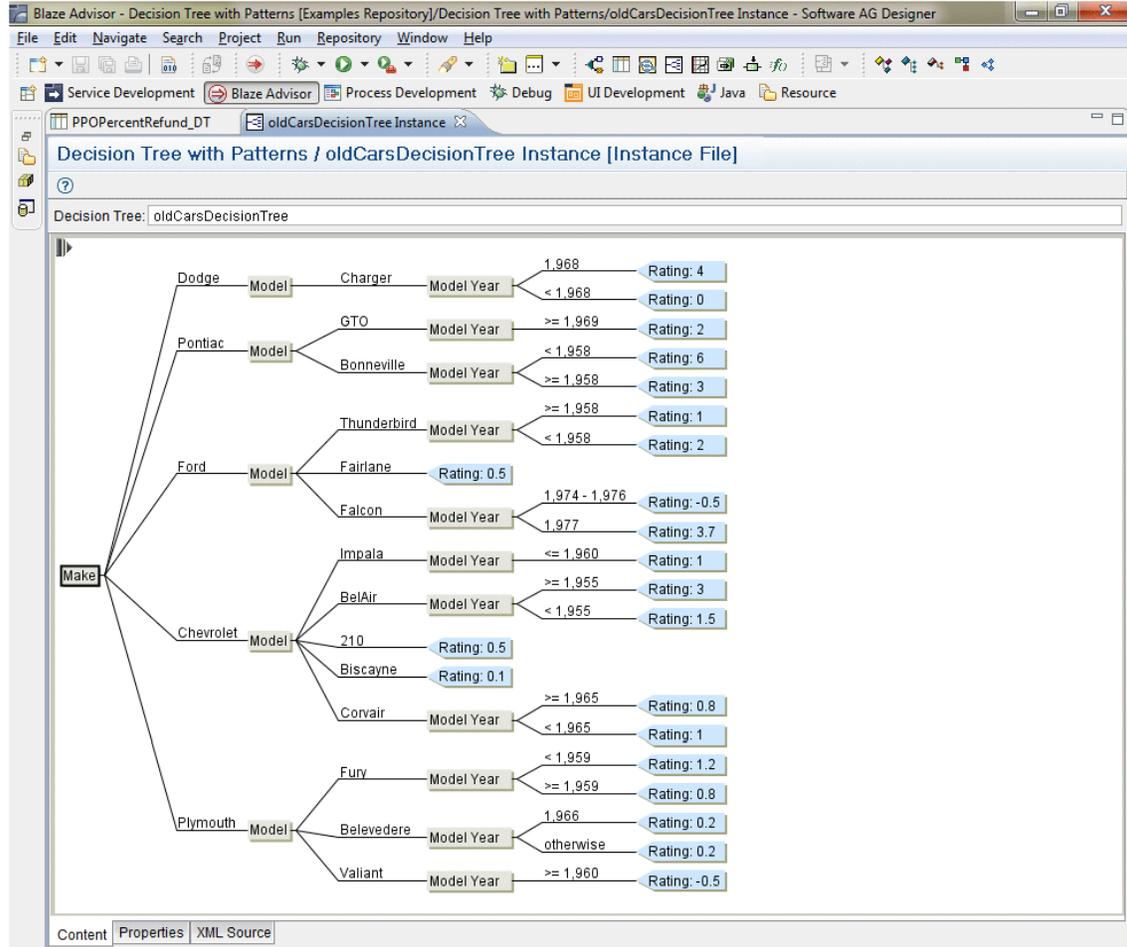
After you develop rules in Software AG Designer, you can drag and drop them onto the business process model design canvas. In a collaborative design environment, you can also drag and drop assets from CentraSite. CentraSite operates as a shared SOA registry/repository of metadata about assets that were developed in Software AG Designer and that are stored on run-time servers such as Integration Server and My webMethods Server.

You export rules developed in Software AG Designer to Integration Servers equipped with a Rules Engine. Each Rules Engine executes the rules that it hosts when those rules are invoked by process steps or tasks.

Software AG offers a browser-based user interface that allows business administrators to make simple changes to rules you develop. You make business rules available to business administrators by exporting them to the My webMethods Server content repository. Business administrators use the browser-based user interface to edit the rules and then to redeploy the rules to the appropriate Rules Engines, and also to customize the dates and times when rules are to take effect.

Blaze

Developers can develop business rules using Blaze Rules Development. Rules can be expressed using metaphors such as decision tables or decision trees. The illustration below shows a rule decision tree.



You deploy rules developed in Blaze Rules Development to Integration Servers equipped with a Blaze Rules Service Support package. The Integration Server generates the rules as rule services and the rule services appear in the Eclipse Package Navigator view. You can drag and drop rule services onto the business process model design canvas from that view. Each Integration Server executes the rule services it hosts when those rules are invoked by process steps or tasks.

Software AG offers a browser-based user interface that allows business administrators to make simple changes to rules you develop. You make rules available to business administrators by generating *rule maintenance applications* (RMAs) from Blaze Rules Development. An RMA is a composite application that includes a browser-based user interface that displays the rule in editable form. You then deploy the RMAs from Blaze Rules Development to My webMethods Server and check the rules into the Blaze repository, which serves as a version control system for the rules. Business administrators use the browser-based user interface to check out rules from the Blaze repository, edit the rules, check the rules back in to the repository, and redeploy the rules to Integration Server.

Developing Tasks

Developers can develop tasks (that is, activities performed by humans as part of a business process) as specialized composite applications using the Software AG Designer UI Development perspective. Within task applications, you develop user interfaces that present the tasks to end users. You can also define actions to occur in response to specified conditions. For example, you can define an action that assigns critical priority to tasks that are not completed within a specified period of time. Software AG Designer provides a variety of built-in actions the task can use, or the task can invoke a service.

The user interfaces for a task can be displayed as webpages in the My webMethods Server user interface. The illustration below shows a task user interface.

My Inbox > New_Employee_Setup Details

Data View Details View Audit View Comments Collaboration Content

Employee Data:

Employee Name:

Office Number:

Telephone:

Furnishings:

Delivered

Pending

Toggle Task Info

Task Info:

Name: Setup: ssmith

Description: New employee setup for Sam Smith.

Created On: 3/8/2011 8:27 AM By My webMethods Administrator

Last Modified On: 3/8/2011 8:32 AM By My webMethods Administrator

Expires On:

Status: Active

Complete Assign to Users... Submit Return

You can also develop your own user interfaces with technology such as Grails and Google Widget Toolkit and access the task with the Task Engine RESTful services.

Some tasks require the performance of multiple activities. You could construct detailed logic within a task to anticipate all possible outcomes, but such logic would be labor intensive, error prone, and difficult to maintain. Instead, you can define *collaboration tasks*; that is, tasks configured to operate in a collaborative work environment. You can implement collaboration tasks in these ways:

- Automatic (within a *collaboration process*). You can configure a task so that when it receives specific business data, it creates child collaboration tasks and assigns them to specific roles or users. Suppose a satellite television company has a new order process that includes a task to install a dish and receiver. Different installation teams and equipment are required depending on the service ordered by the customer. You can configure the parent task to queue one collaboration task to the appropriate installation team and another to the appropriate equipment team based on specific

information in the customer order. You can configure the parent task to complete automatically when the collaboration tasks are completed.

- Manual (by a user). You can configure a task so that a user who opens the task in his inbox can create child collaboration tasks and assign them to other users to help complete the parent task. Suppose a support person from a magazine company receives a task to resolve a customer complaint that issues of a magazine have stopped arriving. The support person could create and assign collaboration tasks to the database administrator and the circulation manager that request information needed to determine the problem.

You can use business rules to assign tasks to users.

Task Engine

The Task Engine provides APIs that enable you to search and manipulate tasks from services or other applications. These APIs are provided as Java classes and as services on the Integration Server to which the Task Engine is connected (see "[Executing Business Processes](#)" on page 43). The Java APIs are created when you create a task application portlet. The APIs provided as services are as follows:

- Built-in Java service API. You can use these services as templates to create custom services in Software AG Designer that can create, modify, locate, queue, and delete tasks on the Task Engine to which Integration Server is connected.
- RESTful service API. You can use these services with the REST service capabilities of Integration Server and My webMethods Server. The services enable you to develop your own user interfaces using technology such as Grails and Google Widget Toolkit.
- Web service API. You can use these services to add, delete, locate, and update tasks on a Task Engine from a client program. The Task Engine hosts a version of the task web service for .NET clients and a version for regular web service clients.

Executing Business Processes

The products that execute business processes are Integration Server, Universal Messaging, and My webMethods Server.

Integration Server

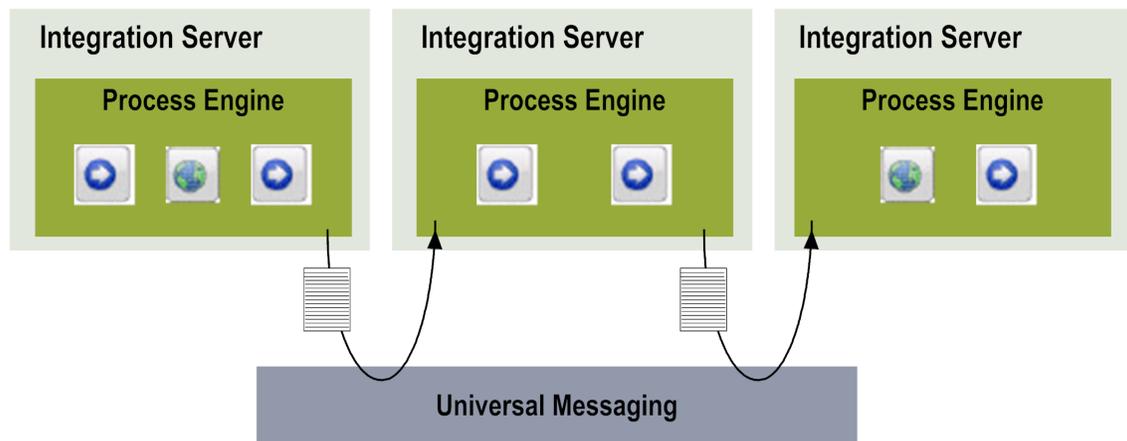
Integration Server is a run-time server that has several functions in business process management.

- Run-time logic for process steps developed in Software AG Designer is created on Integration Server. Integration Server executes business processes and process steps. Every Integration Server that runs process steps is equipped with a Process Engine that controls and directs process execution.

- Services such as Flow, web, Blaze rules, and adapter services that are developed in Software AG Designer are built on Integration Server. Integration Server executes the services when they are invoked by process steps, tasks, or services.
- Business rules developed in Software AG Designer are exported to Integration Servers. The Rules Engines on the Integration Servers execute the rules when they are invoked by process steps or tasks.
- If a process step exchanges documents with an external trading partner, the step sends the document to Integration Server, which sends the document to the partner. The partner returns a document to Integration Server, which returns the document to the process so it can continue to its next step.

Universal Messaging

To improve performance and reliability, you can distribute business process steps across multiple Integration Servers. In this scenario the Integration Servers must connect to Universal Messaging, which routes process data across the Integration Servers.



My webMethods Server

My webMethods Server has two functions in business process management:

- Execution of tasks. Software AG Designer publishes tasks applications, to My webMethods Server, where they are invoked by processes at run time. You equip each My webMethods Server that runs tasks with a Task Engine that controls and directs task execution. At run time, data and control pass from Process Engines to Task Engines and back again until the business process completes.
- User interfaces for editing rules developed in Software AG Designer.

Business Process Model Implementation Details

This section provides a high-level description of the activities a business process developer performs to implement a business process model.

A major feature of Software AG Designer is its collaborative focus, which allows users with different types of expertise to work together on different aspects of a process. The Software AG Designer perspectives that business process developers use to implement business process models are the Process Development, Process Debugging, Process Simulation, Service Development, UI Development, and Rules Development. Each of these perspectives supplies the tools needed by a particular category of users. You can also use Blaze Rules Development.

Configuring Process Steps

The Software AG Designer Process Development perspective offers an extensive set of programming tools that enable a technical user to focus on the detailed implementation of a business process model. You receive a new business process model from a business analyst and you work in the Process Development perspective to configure the process steps. For example, you configure steps to:

- Subscribe to business documents on or publish business documents to Universal Messaging.
- Subscribe to business events on or publish business events to the event-driven architecture (see "[Event-Driven Architecture](#)" on page 71).
- Send or receive JMS messages.
- Invoke integration services or web services.
- Invoke rules, tasks, other business processes, or Trading Networks.

You define the way data passes through and is handled by your business processes. You define the order of steps in your processes, and the conditions under which they start, run, pass data, and end. Business Process Modeling Notation (BPMN) provides a graphical representation of the underlying model and of the functionality and behavior of its steps.

Creating Documents, Services, Tasks, and Rules

Some Software AG products provide data to processes in the form of *documents*. Each document is associated with a *document type*, a schema-like definition that describes the document's structure. For example, a document type named PurchaseOrder might describe the structure of data in a purchase order document.

In Software AG Designer, you can identify existing document types, services, tasks, and rules, or you can create the document types, services, tasks, and rules required by a business process, as follows:

- Browse Integration Servers for existing document types and services, and drag and drop them onto the design canvas.
- Browse CentraSite for document types, services, tasks, rules, and business processes, and reference these assets from process steps.
- Import supported electronic form (*e-form*) templates from a file system folder, a web server, a My webMethods Server configured as a JSR-170 content repository, or a Content Service Platform repository. Software AG Designer converts the imported templates into Integration Server document types.
- Use the Software AG Designer Service Development perspective to develop Flow services and create document types for service inputs and outputs.
- Use the Software AG Designer UI Development perspective to develop tasks (see ["Developing Tasks Within a Process" on page 46](#), below).
- Use the Software AG Designer Rules Development perspective or Blaze Rules Development to develop rules.
- If you are working in a collaborative design environment, publish assets you develop in Software AG Designer to CentraSite.

Developing Tasks Within a Process

If a business process you are developing includes human activities, you use the Software AG Designer UI Development perspective to develop tasks. Tasks are created within a special type of composite application called *task application projects*. You can create a single task within a project, or you can group multiple related tasks within a project. In the UI Development perspective, you can test and debug tasks you develop.

You can define how you want a task to behave when certain events occur. For example, you can specify that the task status should change to Critical when an uncompleted task reaches its expiration date. Software AG Designer provides a variety of built-in actions you can use, or you can call a service.

You can create a task application that uses data from a supported e-form as some or all of the task's business data. You can also implement e-form-enabled tasks with download and upload capability. This capability lets the task user:

- Connect to My webMethods Server or the Content Service Platform to download the e-form data from the task in its original e-form format.
- Disconnect from My webMethods Server or the Content Service Platform and work with the e-form in the local environment.
- Reconnect to My webMethods Server or the Content Service Platform and upload the e-form. My webMethods Server applies the modifications to the task business data.

User interfaces present tasks to end users. You can use the Software AG Designer UI Development perspective to design task user interfaces, which end users then view in the browser-based My webMethods. Alternatively, you can use non-Software AG technologies such as Grails or Google Widget Toolkit to design task user interfaces. In this case, you use RESTful services to create the interaction between the user interface and tasks that are executing in the Task Engine.

After you develop a task, you drag and drop it onto a business process. If you have a complicated task, you might want to break it up into a series of simpler tasks, connected into a *task workflow*. For the end user, the task workflow appears as a seamless flow of task user interfaces in My webMethods. As the user completes each task in the workflow, the next user interface appears, eliminating the need for the user to locate each new task and open it manually.

When you are done developing a business process, you use Software AG Designer to deploy tasks to a My webMethods Server equipped with a Task Engine, so the process can invoke the tasks at run time.

Simulating and Debugging Business Processes

Detailed business process model implementation is an iterative procedure that involves developing, testing, and correcting your code. The Software AG Designer Process Simulation perspective enables you to test a business process by sending a document through it. Among other things, simulation enables you to:

- Discover business process bottlenecks, points of failure, and time lags.
- Predict business process behavior in multiple scenarios.
- Compare the behavior (performance, utilization, cost, and so on) of two or more different processes, or of two or more versions of the same process.

You can use actual historical run-time data in simulations.

You can debug the business process using the debugging tools offered by the Software AG Designer Process Debugging perspective and then simulate the process again, and so on until the process meets your requirements.

Logging and Monitoring Processes

Within each business process model, you set the maximum level for audit logging for that process. The business administrator who monitors processes will refine this setting later to suit his needs. You also define quality of service settings that determine how the process executes at run time and that let you balance process performance, reliability, visibility, and control. For example, you can choose between improving performance by storing process run-time data in RAM or improving reliability by persisting the data.

Mapping Process Steps to Run Times

When you are done developing the business process, you work with your system administrator to map process steps to the Integration Servers (equipped with Process Engines) on which the steps should run. You then deploy the process steps from Software AG Designer to those Integration Servers. Software AG Designer creates a package containing run-time execution information on each of the Integration Servers.

Developing in a Collaborative Environment

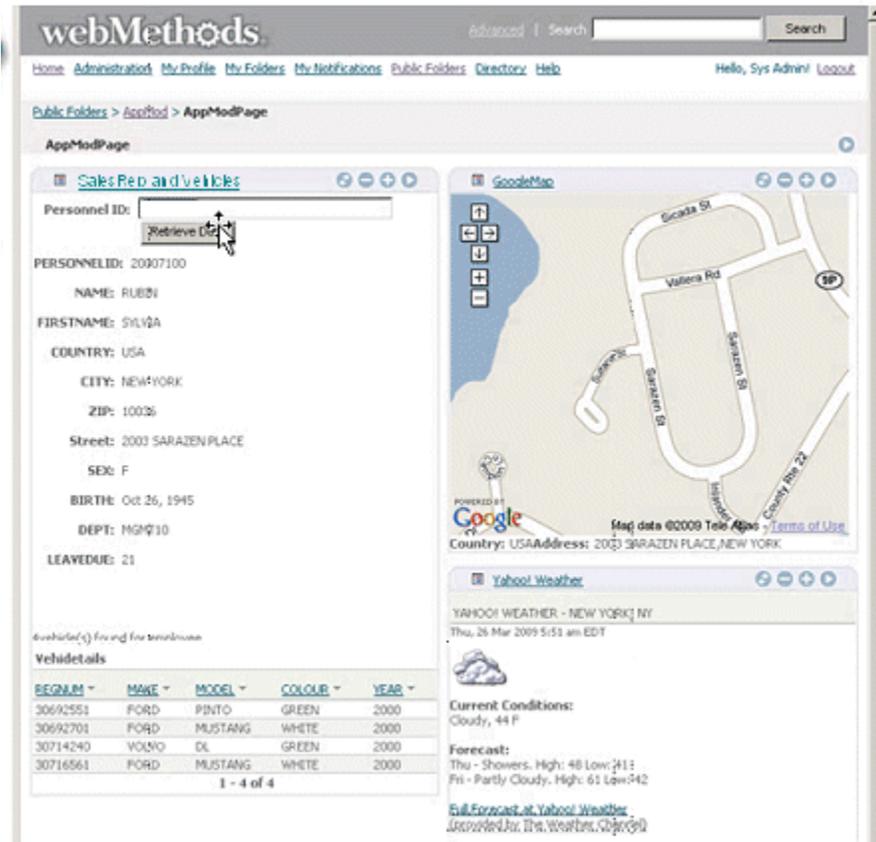
Software AG offers a variety of tools and methods you can use to create a collaborative development environment in which developers can easily share and reuse assets.

- A main tool for collaboration is CentraSite, which operates as a shared database of metadata about assets that are stored in Software AG Designer, Integration Server, and CentraSite. Business process developers can publish services, document types, and business processes from Software AG Designer to CentraSite, and can drag and drop these assets from CentraSite into Software AG Designer.
- Task application developers can define child collaboration tasks and processes that help complete a parent task. Developers can configure a parent task so that:
 - When the task receives specific business data, it creates child collaboration tasks and assigns them to specific roles or users. The task application developer can configure the parent task to complete automatically when the collaboration tasks are completed.
 - A user who opens the task in his inbox can create child collaboration tasks and assign them to other users. When the child tasks are completed, the user can complete the parent task.

For more information on tools and methods you can use to create a collaborative development environment, contact Software AG Global Consulting Services.

Developing Composite Applications

A *composite application* is made up of portlet applications and services that present data from multiple resources on one or more webpages for the end user. Composite applications are also used to create modernized front ends for legacy systems. Suppose you have a mainframe program that stores customer orders and the names of sales representatives. You could create a customer management composite application that pulls that data from the mainframe program and displays it on a webpage. When the end user clicks an order, the composite application gets the order details from the mainframe program and displays it on another webpage. When the end user clicks a sales representative, the composite application gets customer data from the mainframe program and displays the locations and contact information for the sale representative's customers on a Google map.



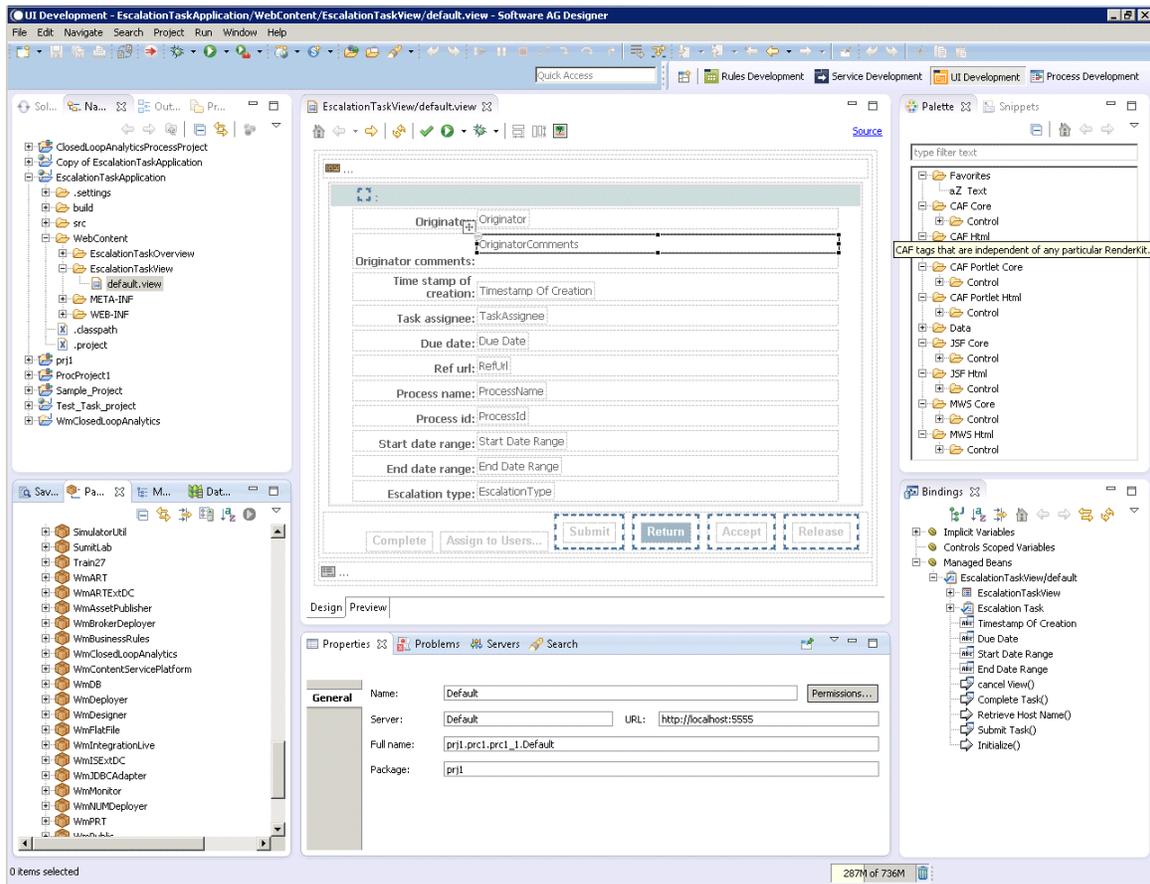
The primary product developers use to develop composite applications is Software AG Designer. Developers might also use CentraSite. The products that execute composite applications are My webMethods Server and Integration Server.

Software AG Designer

Software AG Designer offers software developers a UI Development perspective for designing composite applications.

You create the composite application interfaces in Software AG Designer by dragging and dropping JavaServer Faces (JSF) controls onto a design canvas. You configure each JSF control to perform a specific function, such as submitting a command, alerting a user, obtaining user input (for example, through check boxes or drop-down lists), or adding rendering logic to webpages. Other JSF controls enable you to link and navigate among portlets and webpages. Software AG Designer offers an extensive library of JSF controls. You can then add other components such as Java or web services to the composite application to retrieve and manipulate data. You can completely control the look and feel of these applications by importing HTML templates and CSS style sheets. You can enrich the user experience using your favorite Java script libraries.

The illustration below shows a view from a portlet application in Software AG Designer.



The webpages in your composite applications can invoke services such as Flow, web, adapter, and rule services. Within Software AG Designer, you can develop Flow, web, and rule services, and you can create adapter services that invoke programs on mainframes and UNIX systems. Software AG Designer builds its services on Integration Server.

You can drag and drop the services you want the webpages to invoke onto the design canvas from Integration Server. In a collaborative design environment, you can also drag and drop services from CentraSite. CentraSite operates as a shared SOA registry/repository of metadata about assets that were developed in Software AG Designer and that are stored on run-time servers such as Integration Server and My webMethods Server.

The webpages in your composite applications can also access and display data stored in databases. You can connect to a database from Software AG Designer and then drag and drop database-related items, such as database tables, that you want to use in the webpages onto the design canvas.

Composite applications run on My webMethods Server. Composite applications built in Software AG Designer can use Asynchronous JavaScript and XML (Ajax) technology. With this technology, an Ajax engine acts as the intermediary between the user and My webMethods Server, significantly improving My webMethods Server's response to user

input. Alternatively you can use modern client-side tools like AngularJS to develop parts of your composite application.

My webMethods Server

Software AG Designer publishes composite applications to My webMethods Server. My webMethods Server also hosts composite applications that provide browser-based user interfaces for many Software AG run-time products.

My webMethods Server is a run-time server with a browser-based user interface that lets you customize the look and feel of webpages and control user access to them. My webMethods Server user management also provides a single location from which to define and manage users for many Software AG products.

My webMethods Server provides a built-in Jetty web server that supports both HTTP and HTTPS. You can use an external web server, or cluster of web servers, with My webMethods Server (for example, if an external web server better complies with your corporate IT security policies). My webMethods Server can integrate with the leading web servers, such as Microsoft Internet Information Server or Apache HTTP Server.

Integration Server

Services developed in Software AG Designer are built on Integration Server. Integration Server's function in composite applications is to execute services that are invoked by the composite application's webpages.

Developing in a Collaborative Environment

Software AG offers a variety of tools and methods you can use to create a collaborative development environment in which developers can easily share and reuse assets. A main tool for collaboration is CentraSite, which operates as a shared database of metadata about assets that are stored in Software AG Designer, Integration Server, and CentraSite. Composite applications developers can publish services, document types, and composite applications from Software AG Designer to CentraSite, and can drag and drop these assets from CentraSite into Software AG Designer.

For more information on tools and methods you can use to create a collaborative development environment, contact Software AG Global Consulting Services.

Administering and Monitoring Business Processes

Several instances of a business process or task can run at the same time. For example, your enterprise could hire several new employees at one time, and each new employee would trigger a new instance of the new employee process. For tasks, you could have a purchase order process that sends out an approval request task instance to three

different managers. You administer business process models and task applications, but you monitor process instances and task instances.

The primary products process administrators use to administer business process models and task applications, and process owners and participants use to monitor process instances and task instances, are Monitor, Task Engine, and Mobile Monitor.

Administering Business Processes

Monitor is a browser-based user interface that displays data logged by Integration Server and Optimize for Process for services, documents, and business processes. You can use Monitor to:

- Allow business process instances to run or prevent business process instances from running.
- Resubmit process steps when problems occur.
- Define time-outs that alert you when process steps or tasks run too long.
- Set up audit logging so you can track when process instances and steps start running, change status, end successfully, or fail, and so you can record the path that each process instance took at run time.
- Delete business processes.

You can use Task Engine to:

- Assign tasks to users or roles.
- Allow task instances to run or prevent task instances from running.
- Manually start tasks (for example, to kick off business process instances).
- Delete tasks.

Monitoring Business Process Instances

You can use Monitor to:

- View process instances, find problems, and determine whether the problems are caused by system resources, services, or documents.
- Suspend, resume, or stop problematic process instances, and edit data values and resubmit process instances.
- Create, modify, and delete process model stages, and view the status of stages in process instances and whether the milestones were reached within the specified time periods.
- View projections of when a process instance might complete based on percent complete, and view the average cycle time for running instances.

Mobile Monitor is an application for mobile devices and tablets that provides real-time information about the status of executed business processes. You can use Mobile Monitor:

- View process instances, find problems, and determine whether the problems are caused by system resources, services, or documents.
- View and work on tasks that are assigned to you as you would using Task Engine (see below).

You can use Task Engine to:

- Assign task instances to users or roles.
- Suspend, resume, or delete task instances.
- View task status and the audit log for tasks assigned to you.

You can use Optimize for Process to create, modify, and delete process model stages, and to view the status of stages in process instances, and whether the milestones were reached within the specified time periods. You can also use Presto to view stage status. For more information about Optimize for Process and Presto, see "[Intelligent Business Operations](#)" on page 63.

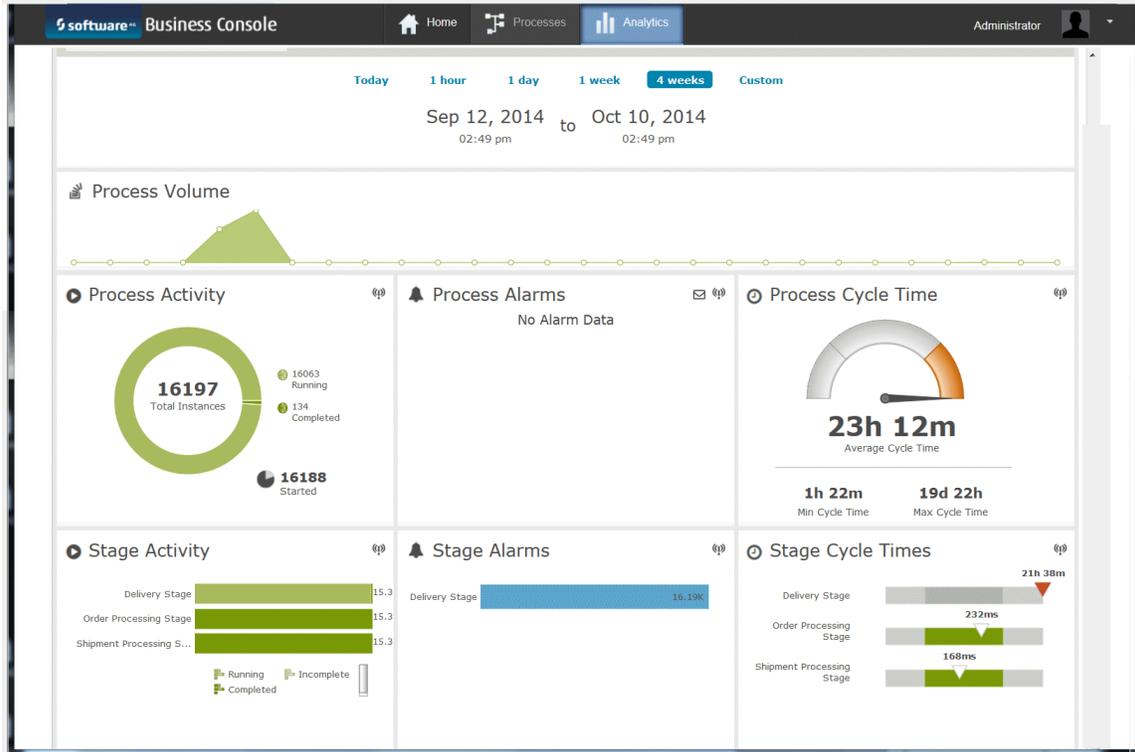
Administering and Working with Business Processes and Tasks

Business Console is a web- and tablet-friendly user interface that provides role-based administration, monitoring, and social guidance for business process owners and participants to manage business processes and tasks. You can use Business Console to:

- Monitor process instances and task instances in real time using a browser.
- Work on tasks that are assigned to you, collaborate with other task assignees and process participants, and save and re-use task searches.
- Endorse other task and process participants and network with experts who can help you complete tasks more quickly and effectively.
- View trending information relating to business processes on social media, and view process analytics.

Solving Problems with Closed Loop Analytics

Closed Loop Analytics is a solution that enables you to quickly and easily improve the performance of your business processes. Closed Loop Analytics collects data about process health from products involved in business process execution and analysis. For example, it collects data on process volume and cycle times from Process Engine, and data on SLA or KPI rule violations from Optimize for Process. Closed Loop Analytics then displays the data on the business process analytics dashboard in Business Console (see illustration below).



Process administrators review the data on the dashboard and escalate problems to users in the form of escalation tasks. Users find the escalation tasks in their inboxes, perform root cause analyses, and write the results in the tasks. Process administrators review the results and use them to improve the processes. The illustration below shows a user task inbox.

The screenshot shows the 'Inbox' view of the Business Console, displaying a list of 'EscalationTask' items. The table below represents the data shown in the screenshot:

Task Name	Task ID	Status	Accept	Priority	Assigned To	Created By	Created Date	Description	Task Type	Actions
Escalation ...	26032		Accept	None	thomas	administrator	10-10-2014, ...	LuxuryLine...	26BBC3A6-...	
Escalation ...	26031		Accept	None		administrator	10-10-2014, ...	LuxuryLine...	26BBC3A6-...	
Escalation Tas...	26030			None	thomas	administrator	10-10-2014, 1...	LuxuryLineOrder	26BBC3A6-99...	
Escalation Tas...	26029			None		administrator	10-10-2014, 1...	LuxuryLineOrder	26BBC3A6-99...	
Escalation Tas...	26028			None		administrator	10-10-2014, 1...	LuxuryLineOrder	26BBC3A6-99...	
Escalation Tas...	26027			None	thomas	administrator	10-10-2014, 1...	LuxuryLineOrder	26BBC3A6-99...	
Escalation Tas...	26026			None	thomas	administrator	10-10-2014, 1...	LuxuryLineOrder	26BBC3A6-99...	
Escalation Tas...	26025			None		administrator	10-10-2014, 1...	LuxuryLineOrder	26BBC3A6-99...	
Escalation Tas...	26024			None		administrator	10-10-2014, 1...	LuxuryLineOrder	26BBC3A6-99...	
Escalation Tas...	26023			None	thomas	administrator	10-10-2014, 1...	LuxuryLineOrder	26BBC3A6-99...	

For information about Optimize for Process, see "Intelligent Business Operations" on page 63.

Extending Your Business Processes into the Mobile Space

The Mobile Suite enables you to extend your integrations and business processes into the mobile space; with these products, users anywhere in the world can interact with integrations and business processes from their mobile devices. The Mobile Suite fully supports Mobile Enterprise Application Platform (MEAP) requirements and spans the full mobile lifecycle, from developing and deploying mobile apps to securely integrating and managing them. The Mobile Suite can also bring existing IT services and processes to mobile devices in an efficient way.

The products you use to support mobile devices are Software AG Designer, Mobile Designer, Mobile Support, and Mobile Administrator.

Software AG Designer

You can develop mobile apps using the Mobile Development perspective in Software AG Designer. Mobile Development provides logic that reduces the amount of code you must write to develop a mobile app.

Mobile Development uses the principles of model-view-controller (MVC) architecture, which separates user interfaces from business logic and data. With Mobile Development, mobile app developers design a user interface and then generate Java code that displays the user interface and responds to user-initiated events, such as the clicking of a button. The developers code business logic separately from the user interface, in Java classes created by Mobile Development.

Mobile Designer

Mobile Designer is a mobile application development tool. With Mobile Designer, mobile app developers create mobile apps in Java and then automatically convert the apps for any mobile device, from an iPad to an Android to a Windows Phone. The converted apps provide the user experience that each device owner expects. Mobile Designer reduces the need for in-house mobile knowledge, which in turn reduces development and maintenance costs.

Mobile Support

Mobile Support is a set of tools that enable mobile app developers and business integration developers to create mobile data synchronization solutions. Such solutions transfer data between mobile devices and backend enterprise applications and resolve conflicts that occur when backend data is updated by multiple sources at the same time.

Mobile Support includes two components:

- Mobile Support Client provides a Java client library of classes composed of methods that initiate requests to synchronize data. Mobile app developers incorporate the synchronization methods in their apps.
- Mobile Support package runs on Integration Server and provides a service that synchronizes data sent by mobile devices. Business integration developers incorporate the synchronization service in their integration services.

Mobile apps submit synchronization requests to Mobile Support Client, which asks Integration Server to execute the services that process the requests. Mobile apps can submit either of the following synchronization requests:

- Download changed data from the backend application to the mobile device.
- Upload changed data from the mobile device to the backend application, synchronize that data with the backend data, and download changed data from the backend application to the mobile device.

Mobile Administrator

Mobile Administrator enables you to do the following:

- Manage mobile applications for multiple platforms using an enterprise app store to distribute apps to mobile devices based on the access rights of the user.
- Build Mobile Designer or native apps in a distributed, cloud-enabled build and test environment.
- Monitor your runtime data, events, and mobile app behavior (for example, when the app started, crashed, closed, and so on).
- Add application policies to and enable push notifications for mobile apps on the fly.
- Manage devices, including device inventory, device policies, and access rights.

4 Master Data Management

- Master Data Management 58
- OneData 58

Master Data Management

Master data is business information that is critical to the operation of an enterprise. Business information might include data about customers, trading partners, employees, suppliers, products, materials, and assets.

In a large organization, different departments might have different versions of the same master data. The data might also be inaccurate or incomplete. The use of this data can therefore cause serious problems. Master data management (MDM) offers an organization the ability to create and maintain a single, trusted, shareable version of master data. MDM offers processes and tools that collect, aggregate, persist, and distribute master data consistently throughout an organization.

The products you use for MDM are OneData and Locate.

OneData

OneData reconciles, cleanses, synchronizes, and stores all your master data, as well as all your data definitions, structures, workflow, processes, and rules. Enterprises deploy OneData as the MDM solution to improve process performance and enable better business decisions.

OneData provides a repository in which to store your master data, and an Internet browser-based user interface that lets you work with that data. It also provides the capability for data interchange with other systems using mechanisms such as schedulable data interchange jobs for importing and export data in multiple formats and REST-based service layers calls.

OneData has an open, extensible data model. You can import your existing model, or use built-in templates. Any domain or subject area that is in the data model becomes part of the solution; there are no proprietary structures. You can easily modify or extend the model at any time, and changes are reflected immediately in the OneData user interface.

OneData supports multiple MDM architectural styles. These architecture styles include centralized, consolidated, coexistence, and hybrid, as well as combinations of these styles. OneData supports both operational and analytical MDM use cases.

OneData integrates with other Software AG products, such as:

- **Locate.** The accuracy of addresses and other location data is critical to business processes, especially business processes that drive results and revenue. In MDM, it is important to not only have cleansed and standardized address information; you must be sure that the captured address actually exists. Locate is the engine behind verifying international addresses for OneData. Locate provides technology for address verification and geocoding, as well as address reference data for more than 240 countries worldwide. Locate checks address data for errors and omissions and fills in missing address attributes when needed.

- **Integration Server.** Integration Server is a run-time server whose primary function is the execution of services. OneData can call Integration Server services (for example, to perform data validation or enrichment).
- **Universal Messaging.** Universal Messaging is fast, reliable, scalable, and flexible Java message oriented middleware (MOM) that provides messaging functionality such as clustering, scheduling, and interface plug-ins, with standard support for the messaging paradigms of publish/subscribe, message queues, and P2P, as well as support for JMS.

OneData uses Universal Messaging as its JMS provider for data acquisition and deployment in real time. If you modify data in OneData, OneData can publish a message to topics or queues in the JMS provider for subscribers to pick up. Conversely, OneData can subscribe to topics and queues in the JMS provider for messages about data changes in back-end systems, such as CRM.

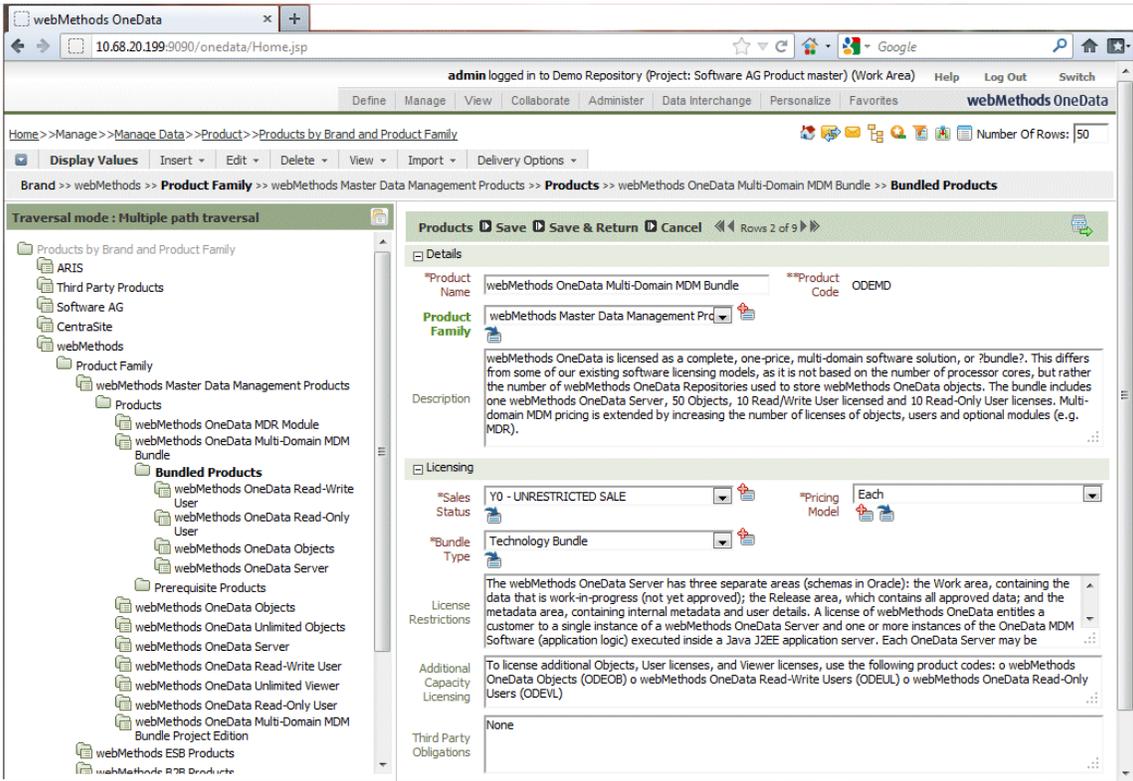
- **Terracotta.** OneData uses Terracotta BigMemory to cache data that can be accessed using the REST API.
- **webMethods Business Process Management (BPM).** You can use webMethods BPM with OneData to define advanced workflow processes.

OneData also integrates with any industry-standard enterprise service bus (ESB) or service-oriented architecture (SOA) environment, to share and reuse the same master data both in operative systems and in data warehousing and reporting systems.

OneData supports end-to-end master data life-cycle capabilities, such as:

- Creation and maintenance of data through theOneData user interface.
- Modeling and configuration, with no coding needed. OneData implementations typically can be accomplished in a matter of weeks.
- Data acquisitions from and data deployment to data sources such as databases, flat files, and XML files.
- Hierarchy management.
- Data change request and approval workflows.
- Data quality (that is, cleansing, matching, and survivorship). OneData can also integrate with third-party data quality tools.
- Data security, auditing, and reporting.

The illustration below shows a hierarchy in the OneData user interface. Details for a node in the hierarchy appear on the right.



The illustration below shows a data grid whose contents reflect a data object created in OneData.

The screenshot shows a webMethods OneData application interface. At the top, a browser window displays the URL 10.68.20.199:9090/onedata/Home.jsp. The application header indicates that 'admin' is logged in to the 'Demo Repository (Project: Software AG Product master) (Work Area)'. Below the header, there are navigation tabs: Define, Manage, View, Collaborate, Administrator, Data Interchange, Personalize, and Favorites. The main content area shows a breadcrumb trail: Home > Manage > Manage Data > Product > Products. A toolbar includes options for 'Display Values', 'Insert', 'Edit', 'Delete', 'View', 'Import', and 'Delivery Options'. Below the toolbar, a table lists various products. The table has columns for 'Product Name' and 'Product Family'. Each row includes a checkbox, an 'Edit' icon, and a 'Delete' icon. The products listed include 'webMethods OneData MDR Module', 'webMethods Trading Networks', 'webMethods Reverse Invoke Server', 'webMethods Integration Server', 'webMethods Integration Server for BPMS Bundle', 'webMethods ESB', 'webMethods ESB Component', 'webMethods Broker', 'webMethods Broker for BPMS Bundle', 'webMethods Integration Platform Project', 'webMethods Arriba Supplier OnRamp', 'webMethods OnRamp for Commerce One MarketSite', 'webMethods Enterprise JavaBeans Adapter', 'webMethods PeopleSoft Enterprise Adapter', 'webMethods Remedy Adapter', 'webMethods Siebel Adapter', 'webMethods Test', 'webMethods Communicate Server Software Base Package', 'webMethods Workflow', 'webMethods B2B Quickstart Edition', 'webMethods B2B Quickstart - Additional Adapters', 'webMethods CAF Designer', 'webMethods Designer', and 'webMethods Communicate Add-on for Integration Server'. The 'Product Family' column lists categories such as 'webMethods Master Data Management Products', 'webMethods B2B Products', 'webMethods ESB Products', 'webMethods BPMS products', and 'Common Products'. The bottom of the interface shows a status bar with 'Rows 1..50 of 130' and a 'Request' button.

	Product Name	Product Family
<input type="checkbox"/>	webMethods OneData MDR Module	webMethods Master Data Management Products
<input type="checkbox"/>	webMethods Trading Networks	webMethods B2B Products
<input type="checkbox"/>	webMethods Reverse Invoke Server	webMethods ESB Products
<input type="checkbox"/>	webMethods Integration Server	webMethods ESB Products
<input type="checkbox"/>	webMethods Integration Server for BPMS Bundle	webMethods ESB Products
<input type="checkbox"/>	webMethods ESB	webMethods ESB Products
<input type="checkbox"/>	webMethods ESB Component	webMethods ESB Products
<input type="checkbox"/>	webMethods Broker	webMethods ESB Products
<input type="checkbox"/>	webMethods Broker for BPMS Bundle	webMethods ESB Products
<input type="checkbox"/>	webMethods Integration Platform Project	webMethods ESB Products
<input type="checkbox"/>	webMethods Arriba Supplier OnRamp	webMethods ESB Products
<input type="checkbox"/>	webMethods OnRamp for Commerce One MarketSite	webMethods ESB Products
<input type="checkbox"/>	webMethods Enterprise JavaBeans Adapter	webMethods ESB Products
<input type="checkbox"/>	webMethods PeopleSoft Enterprise Adapter	webMethods ESB Products
<input type="checkbox"/>	webMethods Remedy Adapter	webMethods ESB Products
<input type="checkbox"/>	webMethods Siebel Adapter	webMethods ESB Products
<input type="checkbox"/>	webMethods Test	webMethods ESB Products
<input type="checkbox"/>	webMethods Communicate Server Software Base Package	webMethods BPMS products
<input type="checkbox"/>	webMethods Workflow	webMethods BPMS products
<input type="checkbox"/>	webMethods B2B Quickstart Edition	webMethods B2B Products
<input type="checkbox"/>	webMethods B2B Quickstart - Additional Adapters	webMethods B2B Products
<input type="checkbox"/>	webMethods CAF Designer	webMethods CAF Products
<input type="checkbox"/>	webMethods Designer	Common Products
<input type="checkbox"/>	webMethods Communicate Add-on for Integration Server	webMethods BPMS products

5 Intelligent Business Operations

- Intelligent Business Operations 64
- Intelligent Business Operations Example 65
- Visualizing Data 66
- Mining Process Data and Analyzing Root Causes 67
- Defining Baseline, Metrics, and Alerts 68
- Detecting and Analyzing Complex Patterns in Real Time 69
- Event-Driven Architecture 71
- Extending Your Intelligent Business Operations into the Mobile Space 72

Intelligent Business Operations

Intelligent Business Operations is a method for monitoring and analyzing key performance indicators (KPIs) and linking them to business processes and daily activities to make sure the entire enterprise is moving toward stated performance goals. Intelligent Business Operations enables you to detect and collect events, visualize and analyze data, and define rules and alerts so you can improve your organization's operational or strategic decision-making.

Business Event Processing

Event processing is an emerging technology in business systems. It provides continuous, real-time insight into events that flow through business networks. Originally used in ultra high-volume stock trading scenarios, event processing has expanded into other aspects of business due to its ability to improve end-to-end visibility, situational awareness, and business agility.

Business event processing begins with the arrival of events containing business data such as a condition, a trend, or a change. Typically, the events are emitted in streams from live data sources, such as tweets from Twitter feeds. The streams are analyzed in real time to identify and extract events that are meaningful to your organization.

As business event processing technologies become more mainstream, they are being applied to more and more business cases. These include:

Business Case	Examples
Financial	Credit card and debit card fraud detection, trading optimization.
Logistics monitoring	Package tracking, fleet management, route optimization.
Manufacturing	Defect detection, machine monitoring, operations optimization.
Healthcare	Fraudulent claims detection, patient monitoring, safety operations.
Government	Homeland security, system security, suspicious activity monitoring.

Business Activity Monitoring

Business activity monitoring (BAM) enables an enterprise to monitor the performance of the business processes it uses to deliver goods and services. Business optimization enables an enterprise to use the monitoring information to proactively manage and optimize those business processes.

Business process data is monitored via key performance indicators (KPIs). KPIs are quantifiable measurements that reflect the critical success factors of an enterprise. For example, in a car loan application process, you might define a KPI to monitor process cycle time. KPIs monitor data for exceptions and trends and alert you to problems.

Business activity monitoring also monitors process health, such as transition times, step times, and number of tasks at each step.

Products Involved in Intelligent Business Operations

The primary products you use in Intelligent Business Operations are Presto, Process Performance Manager, Optimize for Process, Universal Messaging, Apama, and Terracotta.

Intelligent Business Operations Example

Suppose your company is a bank that wants to improve its process for car loan applications. With Intelligent Business Operations, you could do the following:

Category	Activity	Example
Pattern detection	Run continuous queries to detect and collect identified events.	Monitor Twitter feeds to detect and collect negative tweets that contain keywords like the name of your bank and “unhappy,” “slow,” or “bad.”
Visualization	View the collected data in various formats.	Text of each complaint, a Google map of the complaint locations, and a graph of the process cycle times.
Historical process mining	Examine historical data to determine whether similar complaints occurred in the past.	Aggregate fast process cycle times, aggregate slow process cycle times, and compare the results. Filter data to find the age of applicants in the slow processes.

Category	Activity	Example
Root cause analysis	Drill into historical data to find the cause of the problem.	Applications from people under 18 in certain countries loop back through the process because of a form that requires a guardian's signature.
Process improvement	Improve the business process model.	Add a line to the application for a guardian's signature.
Real-time baselines, metrics, and alerts	Monitor key performance indicators (KPIs) and create rules that send alerts when data is out of compliance.	Set up a KPI for process cycle time and create a rule that alerts when cycle time is one standard deviation slower than normal.

Visualizing Data

Presto

Presto is a data mashup and visualization tool used to deliver data stored in other Software AG products to end users. Its tooling is entirely browser-based to facilitate easy configuration of widgets. It allows end users to connect to distributed data sources such as databases, XML files, Excel spreadsheets, Apama events, graphs from Process Performance Manager, and process instance details from Optimize for Process. Presto can also visualize data from web feeds (RSS or Atom) or web services (REST or WSDL).

In the webMethods product suite, Presto is part of business process management (BPM) and is used to visualize process event data collected by Optimize for Process. In the context of BPM, Presto offers these dashboards:

- Business Process Analytics dashboard that shows data on process volume and cycle times from Process Engine, and data on SLA or KPI rule violations from Optimize for Process, as part of Closed Loop Analytics solution. For more information, see ["Solving Problems with Closed Loop Analytics "](#) on page 53.
- Business Process Dashboards that display summary metrics, instance detail, and other performance information about processes that has been collected by Optimize for Process. These dashboards are as follows:
 - Process Overview Dashboard. Provides high-level information about a selected process, including intrinsic metrics, historical views, and normality comparisons across a specified time range. This dashboard helps you quickly identify potential issues, obtain information about alarms and stage activity, and use email or escalation actions to share findings with others.

- **Stage Instances Dashboard.** Shows stage activity and performance for a process across a specified time range. The top of the page shows metrics about each stage, including information about activity, cycle time, and alarms. Selecting table rows reveals information about individual stage instances and enables you to send email notifications or escalation actions for individual stage instances.
- **Process Instances Dashboard.** Shows process activity and performance across a specified time range. A chart at the top of the page displays historical information about process volume activity and cycle time performance. A table at the bottom of the page shows individual process instances and enables you to send email notifications or escalation actions for individual process instances.

These dashboards are displayed in My webMethods and Business Console.

Presto provides a customizable dashboard that comes with out-of-the-box components such as line charts and grids. You can generate dashboards based on a flexible layout, visualizing both streaming data and data from Terracotta BigMemory. Data from Terracotta BigMemory can be refined and condensed using Real-Time Analytics Query Language (RAQL) before being visualized.

Presto also provides standards-based connectors you can leverage to consume data from other Software AG products such as Apama, CentraSite, and Integration Server.

Terracotta

Presto uses Terracotta BigMemory for caching and for in-memory stores for large datasets along with Presto Analytics and Real-time Analytics Query Language (RAQL). Presto Analytics enables clients to analyze large datasets containing traditional, transactional data, event-driven data, and unstructured data collected throughout the enterprise.

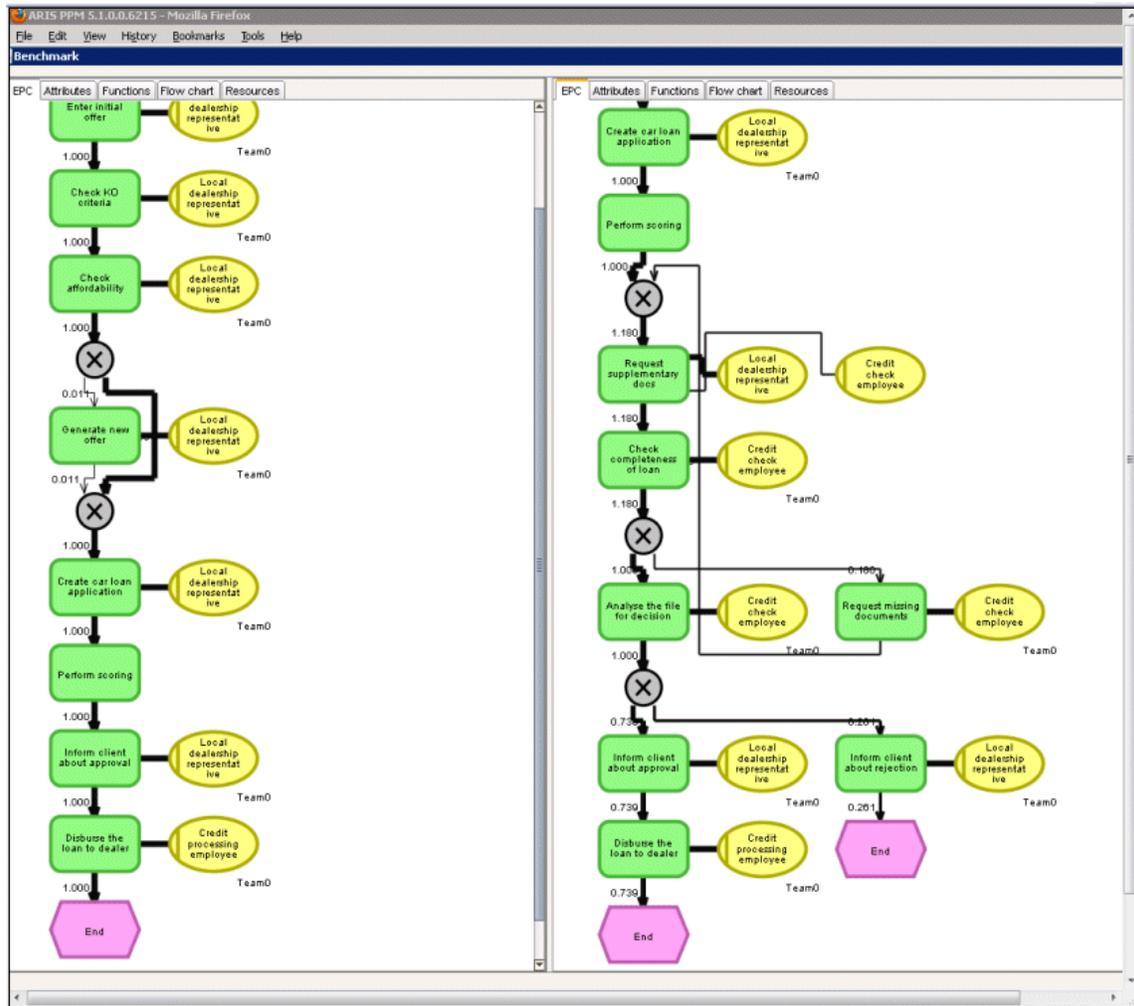
Mining Process Data and Analyzing Root Causes

Process Performance Manager

Process Performance Manager draws data from data sources you identify, such as databases containing business process execution and audit data, or data stores of trading partners contacted by business process steps, or third-party applications such as Oracle Applications, SAP, or Salesforce. In addition to facilitating process, function, and organizational analyses, Process Performance Manager enables you to evaluate process-independent data using data analytics.

You specify the KPIs (cycle times, frequencies, costs, adherence to deadlines, and quality guidelines) to investigate, and Process Performance Manager automatically reconstructs the entire business process in an easy-to-read, event-driven process chain view. Process Performance Manager provides a wide selection of chart types and tables in which to display the results of your analyses clearly and in detail. Process Performance Manager data is displayed in an Internet browser-based user interface. The illustration below

shows the comparison of the aggregated fast-running processes versus the aggregated slow-running processes described in "Intelligent Business Operations Example" on page 65.



Defining Baseline, Metrics, and Alerts

Optimize for Process

Optimize for Process is a Java application that enables you to monitor business processes, activities, events, and other business-related data in real time. The collected monitoring data is displayed in the browser-based Optimize for Process user interface. In this interface you can view and evaluate business data, and define rules that will trigger alerts when problems arise.

Optimize for Process uses data collectors to gather data from business sources and other Software AG products. Optimize for Process comes with key performance indicators (KPIs) that help you measure business activities, and business process developers can

develop custom KPIs that reflect critical success factors for your organization's business processes. For example, developers might define KPIs to monitor margin, revenue, customer satisfaction, and inventory levels. At run time, Universal Messaging routes the KPI readings from the Process Engines that execute the business process steps to Optimize for Process.

You can define rules in Optimize for Process that define problem conditions in your business processes. For example, you might define a rule that specifies acceptable inventory levels. Optimize for Process compares KPI readings against the rules and alerts you when a monitored business process is out of compliance.

Optimize for Process offers APIs and other services that enable you to extract monitoring data from its database for use in other systems, such as Presto.

Terracotta

For maximum performance, Optimize for Process uses Terracotta BigMemory to cache much of the data it uses in support of business activity monitoring. Data that it caches includes:

- Metadata such as dimension definitions, rule definitions, and KPI definitions.
- Raw data that is provided by data collectors for analysis.
- Readings, statistics, and other results produced during monitoring.
- Messages and notifications related to the handling of events.

Optimize for Process also uses Terracotta for clustering. The members of the cluster share data using caches on the Terracotta Server Array. Each member of the cluster connects to the Terracotta Server Array to store and retrieve items from the shared caches.

Detecting and Analyzing Complex Patterns in Real Time

Apama is an event processing platform that monitors rapidly moving event streams, detects and analyzes important events and patterns of events, and immediately acts on events of interest according to your specifications.

You develop Apama applications that contain custom monitors that specify the events or patterns of events that interest you. These specifications are the logical equivalent of database queries. After you load your monitors into the Apama correlator, incoming events flow over the monitors and they monitor incoming event streams for the events and patterns you specified. When a monitor finds a matching event or pattern, it processes the event or pattern according to the rules you specified in the monitor. Apama can support huge numbers of individual monitors operating simultaneously.

Developing Apama Applications

Apama offers multiple approaches to developing Apama applications. Your development team can use one, two, or all three approaches in a single application.

- Developers can make use of the full set of APIs and technologies within the Apama architecture to create sophisticated, customized CEP solutions. Developers can create applications directly in Apama native event processing language (EPL) or in Java via Apama's Java in-process API for processing events.
- Developers and business analysts can create scalable applications that process events originating from very large populations of real-world entities. Scaling, both vertically (same machine) and horizontally (across multiple machines), is inherent in Apama query applications. Scaled deployments on multiple machines use distributed cache technology to maintain and share application state. This makes it easy to deploy across multiple servers, and keep the application running even when some servers are taken down for maintenance or fail. Apama queries can be used alongside EPL monitors in the same correlator process, interacting by sending events between them.
- Business analysts can also use Apama's Event Modeler to create *scenarios* without having to write code. A scenario is a sequence of event monitoring and event responses that constitutes a single business activity. The Event Modeler is a graphical environment that complements Apama's event processing language.

Apama's Data Player accelerates the development/deployment cycle of EPL applications, Java applications, and scenarios by letting users pre-test applications using event streams captured in Apama. Users can investigate the likely behavior of Apama applications prior to deployment via simulation, or analyze the actual performance of those applications already in production. Users can select event streams to test, and specify segments of time from the past to use (for example, an entire day or a specific 30-minute period). Users can accelerate replay speeds to many times the actual live speeds, or slow them down or pause for more careful exploration of event processing operations.

Integrating Apama with External Data Sources

You can connect Apama to any event data source, database, messaging infrastructure, or application. There are three ways to do this:

- Develop client applications with Apama Java, .NET, or C++ APIs.
- Create applications that use the Java Message Service (JMS) connectivity integrated into the Apama correlator.
- Implement Integration Application Framework (IAF) adapters. The IAF is a server component that adapters plug into for run-time invocation.

IAF adapters provide bi-directional connectivity between an external event source and Apama, and XML-based mapping between the event source's data format and Apama's internal format. Apama offers a range of standard adapters for capital markets,

infrastructure, and connectivity to data and messaging sources, and provides APIs for building custom adapters.

Building Dashboards

Apama provides a visual design and deployment environment called Dashboard Builder that non-technical users can use to create sophisticated dashboards. From these dashboards, users can start, stop, parameterize, and monitor Apama scenarios and data views. Dashboard Builder offers a wide range of visual objects and drag-and-drop technology. For applications written in EPL, users can create data views and use Dashboard Builder to create a dashboard from the data views.

Managing and Monitoring Apama Platform Deployments

Apama offers a Management and Monitoring console from which you can manage and monitor an entire Apama platform deployment. The console provides a centralized control service with incorporated dashboard, and sentinel agents that run on the machines involved in the deployment. The control service uses the sentinel agents to implement management policies you have configured; that is, policies that start, shut down, recover and configure event correlators, adapters, and Apama client applications on the machines.

Event-Driven Architecture

Intelligent Business Operations is supported by event-driven architecture (EDA). EDA is a software architecture pattern that supports the production and detection of events, and the consumption of and reaction to events. The products that enable EDA are Universal Messaging, Software AG Designer, Integration Server, and Terracotta.

Universal Messaging

EDA uses Universal Messaging as the intermediary that delivers data from producers to consumers. Event producers publish JMS messages to Universal Messaging in the form of *events*. Each EDA event is associated with an *event type*, an XML schema or schema-based definition that describes the event's structure. Each event type is associated with a JMS topic. Event consumers use JMS subscriptions to receive events on JMS topics. Products such as Integration Server publish and subscribe to events on the JMS topics. Presto subscribes to events.

Software AG Designer

Software AG Designer is an Eclipse-based graphical development tool that offers an Events Development perspective for developing event types. You create and edit event types within event type projects. You can use predefined event types in your projects.

Event types are suite-wide assets. You deploy them to a run-time repository called the Event Type Store, which is shared by all products that support EDA. The JMS objects needed to support the event types are automatically created during deployment. After an event type is deployed, any product can produce or consume events based on that event type.

You can store event type projects in the Software AG Designer workspace, and deploy them from Software AG Designer to the Event Type Store. You can also store event type projects in a version control system and use Deployer and the Asset Build Environment to deploy them to the Event Type Store. You can govern your event types using CentraSite.

Integration Server

Integration Server is a run-time server that offers built-in services that support EDA. One service transforms an event into a document, while another sends an event to a JMS topic. Integration Server can act as a bridge between EDA and other products, such as Optimize.

Terracotta

EDA uses Terracotta BigMemory for guaranteed delivery of events. At a high level, the emitting product, such as Optimize for Process, issues an emit call, and the event is stored in a persistent BigMemory cache. After event is routed to the destination endpoints, the event is removed from the Terracotta cache.

Extending Your Intelligent Business Operations into the Mobile Space

The Mobile Suite enables you to extend your integrations and business processes into the mobile space; with these products, users anywhere in the world can interact with integrations and business processes from their mobile devices. The Mobile Suite fully supports Mobile Enterprise Application Platform (MEAP) requirements and spans the full mobile lifecycle, from developing and deploying mobile apps to securely integrating and managing them. The Mobile Suite can also bring existing IT services and processes to mobile devices in an efficient way.

The products you use to support mobile devices are Software AG Designer, Mobile Designer, Mobile Support, and Mobile Administrator.

Software AG Designer

You can develop mobile apps using the Mobile Development perspective in Software AG Designer. Mobile Development provides logic that reduces the amount of code you must write to develop a mobile app.

Mobile Development uses the principles of model-view-controller (MVC) architecture, which separates user interfaces from business logic and data. With Mobile Development, mobile app developers design a user interface and then generate Java code that displays the user interface and responds to user-initiated events, such as the clicking of a button. The developers code business logic separately from the user interface, in Java classes created by Mobile Development.

Mobile Designer

Mobile Designer is a mobile application development tool. With Mobile Designer, mobile app developers create mobile apps in Java and then automatically convert the apps for any mobile device, from an iPad to an Android to a Windows Phone. The converted apps provide the user experience that each device owner expects. Mobile Designer reduces the need for in-house mobile knowledge, which in turn reduces development and maintenance costs.

Mobile Support

Mobile Support is a set of tools that enable mobile app developers and business integration developers to create mobile data synchronization solutions. Such solutions transfer data between mobile devices and backend enterprise applications and resolve conflicts that occur when backend data is updated by multiple sources at the same time.

Mobile Support includes two components:

- Mobile Support Client provides a Java client library of classes composed of methods that initiate requests to synchronize data. Mobile app developers incorporate the synchronization methods in their apps.
- Mobile Support package runs on Integration Server and provides a service that synchronizes data sent by mobile devices. Business integration developers incorporate the synchronization service in their integration services.

Mobile apps submit synchronization requests to Mobile Support Client, which asks Integration Server to execute the services that process the requests. Mobile apps can submit either of the following synchronization requests:

- Download changed data from the backend application to the mobile device.
- Upload changed data from the mobile device to the backend application, synchronize that data with the backend data, and download changed data from the backend application to the mobile device.

Mobile Administrator

Mobile Administrator enables you to do the following:

- Manage mobile applications for multiple platforms using an enterprise app store to distribute apps to mobile devices based on the access rights of the user.

- Build Mobile Designer or native apps in a distributed, cloud-enabled build and test environment.
- Monitor your runtime data, events, and mobile app behavior (for example, when the app started, crashed, closed, and so on).
- Add application policies to and enable push notifications for mobile apps on the fly.
- Manage devices, including device inventory, device policies, and access rights.

6 SOA Governance and API Management

■ SOA Governance and API Management	76
■ Exposing and Consuming Services	76
■ Exposing and Consuming APIs	77
■ Governing Your SOA and API Management System	78
■ Monitoring and Managing Your SOA or API Management System	80

SOA Governance and API Management

Service-oriented architecture (SOA) is an architectural style in which development groups within an enterprise create and maintain services that capture business capabilities and related artifacts in accordance with defined standards of quality, consistency, and interoperability. The enterprise maintains the services in a registry and exposes them to consumers inside the enterprise. Service consumers, such as developers or process designers, can combine and reuse the services to more quickly and easily create a variety of business applications.

API management is a system in which an enterprise publishes application programming interfaces (APIs) in a secure environment. The enterprise maintains the APIs in a registry and exposes them to consumers inside or outside the enterprise. API consumers, such as developers or trading partners, can use the APIs to more quickly and easily create a variety of business and mobile applications. API management typically does the following:

- Securely exposes APIs to internal and external consumers.
- Exposes API documentation to developers through a developer portal.
- Collects usage data for analytic purposes.
- Manages the lifecycle of APIs.

Services in an SOA and APIs in an API management system are distributed over a network, and are often made available to client applications through a mediation layer. The mediation layer provides a layer of abstraction that prevents client applications from having to know where the services are running or certain details about the API, or which languages, technologies, or platforms were used in their development.

Exposing and Consuming Services

The product used to expose and consume services is CentraSite. CentraSite is a standards-based registry and repository that provides the infrastructure you need for the design-time aspects of SOA.

Service providers can develop services and other reusable assets such as XML schemas, and event types using Software AG Designer (see "[Application Integration](#)" on page 9), and then register these assets in the CentraSite registry, or *catalog*. Service providers can also register assets that are customized for your environment, such as reusable Java libraries and portlets. The CentraSite catalog supports the Java API for XML Registries (JAXR) standard, and is UDDI v2- and UDDI v3-compliant. CentraSite provides a browser-based user interface that lets service providers submit services to the catalog, and lets service consumers browse the catalog for services to use. The illustration below shows the CentraSite service catalog.

The screenshot shows the CentraSite ActiveSOA interface. At the top, there is a navigation bar with the CentraSite logo and user options like Administrator, My Account, About, and Log Off. Below this is a secondary navigation bar with tabs for Home, Asset Catalog, Policies, Administration, Reports, and Operations. The main area is titled 'Assets (34)' and features a search bar and an 'Add Asset...' button. A table lists several assets with columns for Name, Key, Type, and Version. The assets include UDDI Publish API Services, UDDI Inquiry Services, UDDI Custody and Ownership Transfer AP, UDDI Security Service, LifecycleManagementService, ImportXPDLService, and SearchService. On the left, there is a 'Types' sidebar with a tree view of asset categories.

CentraSite can but does not have to store the asset itself in the catalog. A catalog entry might simply describe an asset or indicate its availability. The actual asset itself might reside elsewhere in your enterprise.

CentraSite uses Ehcache to cache asset type information and asset instance data, and to speed up database lookups and queries.

Exposing and Consuming APIs

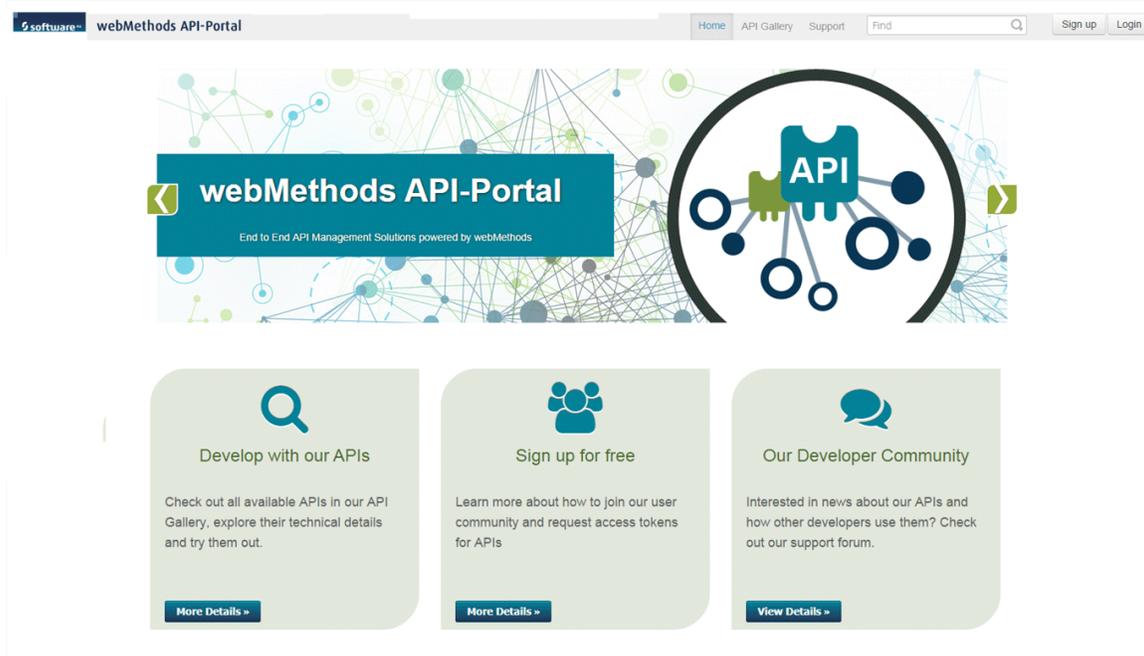
The product used to expose and consume APIs is API-Portal. API-Portal is a web-based, self-service portal through which your organization can securely expose APIs to external developers, partners, and other consumers for use in their own apps.

API-Portal supports both SOAP- and REST-based APIs. Developers can create APIs using design tools like Software AG Designer and register them in CentraSite along with other APIs from third-party environments.

If you are an API provider, you register your APIs in the CentraSite catalog, and then expose them in the API-Portal API Gallery. You can brand your API-Portal using logos, skins, corporate colors and fonts, and more; change layouts for the portal landing page, API Gallery, and API details pages; and customize the portal's privacy and usage terms. API-Portal collects statistics that help you understand where your visitors are coming from, which pages attract the most interest, and which APIs are the most popular so you can improve the portal experience and increase API adoption.

If you are an API consumer, you can quickly find the APIs you need in the API Gallery or by running a simple keyword search. API-Portal provides code samples with expected error/return codes and descriptions so you can easily try out APIs.

A collaborative community environment in API-Portal allows providers to provide information about APIs, and consumers to rate APIs and participate in discussions with other developers.



Governing Your SOA and API Management System

The products you use to govern your SOA and your API management system are CentraSite, Mediator, Terracotta, Integration Server, and Enterprise Gateway.

CentraSite

CentraSite is a standards-based registry and repository that provides the infrastructure you need for the governance and run-time aspects of SOA and API management.

You can control design-time events such as the acceptance of new assets into the catalog and the modification of existing assets in the service catalog through *policies*. For example, you could define a policy that new services submitted to the catalog must be approved by specified individuals like SOA architects. You can also use policies to define review and approval processes, perform quality assurance tests, and issue notifications when new services or APIs are added or when existing services or APIs are about to be modified.

CentraSite enables you to define security, audit logging, service level agreement (SLA) monitoring, and routing policies that control the use of services and APIs in the service

catalog or API Gallery, respectively. Policies specify actions the mediation layer is to perform when a client application requests a service or API. For example, actions can prevent unauthorized access to a service or API, route a request to the appropriate service or API, record events to a logging system, or monitor performance attributes and send alerts when specified thresholds are exceeded. The illustration below shows a policy definition for a service in CentraSite.

The screenshot displays the CentraSite ActiveSOA web interface. The top navigation bar includes the software logo, 'CentraSite™ ActiveSOA', and user options like 'Administrator', 'My Account', 'About', and 'Log Off'. Below this is a menu with 'Home', 'Asset Catalog', 'Policies', 'Administration', 'Reports', and 'Operations'. The main content area is titled 'Edit Action Parameters' and contains several sections:

- Action Information:** Action Name: Monitor Service Level Agreement. Description: The policy monitors the service parameters for each consumer and reports SLA violation to the user.
- Action Configuration:** A table with columns for Name, Operator, and Value.

Name	Operator	Value
Availability	Greater Than	90
Average Response Time	Less Than	200
- Alert for Applications:** A dropdown menu with a plus icon.
- Alert Parameters:**
 - * Alert Interval: 1 Minutes
 - * Alert Frequency: Every Time
 - Reply to Destination: CentraSite
 - Local Log Log Level: Info
 - SNMP
 - E-mail
 - Alert Message: SLA violated

The bottom of the interface shows a breadcrumb trail: Welcome > Browse > Run-Time > enforce Service Le...greement - Policy > Monitor Service Level Agreement - Action Parameters.

Mediator

Mediator is a product that runs on the run-time server Integration Server. Mediator does the following:

- Mediates between consumer applications and service or API providers. Mediator receives requests from consumer applications and forwards them to providers, then returns responses from providers to consumer applications.
- Makes sure requests from and responses to consumer applications conform to policies defined in CentraSite.
- Transforms requests from and responses to consumer applications according to transformation steps defined in CentraSite.
- Uses request context or content to route requests from consumers to different service endpoints, or to load balance requests.

Mediator uses Terracotta BigMemory to cache the virtual services you publish to it. It also uses BigMemory to cache many of the artifacts associated with the virtual services, such as performance metrics, policies, consumer applications, and registered consumers.

When you install Mediator on a cluster of Integration Servers, BigMemory maintains these caches on the Terracotta Server Array.

Integration Server and Enterprise Gateway Server

Integration Server is a run-time server that hosts Mediator and executes its services.

In an SOA or API management system, Integration Server sits behind an internal firewall and is not allowed to directly interact with external clients through the DMZ. Instead, another Integration Server in your DMZ, called an Enterprise Gateway Server, acts as an intermediary between external clients and the internal Integration Server, Mediator, and API-Portal. The Enterprise Gateway Server protects these products and their applications, services, and data, from malicious attacks from external client applications. Administrators can secure traffic between service and API consumer requests and the execution of services on Mediator by:

- Filtering requests from and blacklisting specified IP addresses.
- Detecting and filtering requests from mobile devices.
- Avoiding additional inbound firewall holes.

Monitoring and Managing Your SOA or API Management System

The product you use to monitor and manage your SOA and your API management system is Insight.

Insight

Insight is a monitoring and management tool for SOA and API management that lets you:

- Understand the interdependencies of complex networks that change frequently.
- Identify and isolate the root cause of problems quickly, preventing them from spreading and rippling through the network.
- Predict and prepare for the impact of planned changes, enabling publishing without interruption.
- Create and apply policy to monitor performance and enforce SLAs without requiring custom programming.
- Detect rogue, ungoverned services or APIs when they are consumed and register them automatically in CentraSite to put them under governance control.

An Insight Agent runs continuously on application servers on your network that host web services, such as Integration Servers. The agent collects data about service traffic from the application servers. The agent detects incoming and outgoing web service

calls, records them, and reports them to Insight Server. The agent also collects run-time statistics, evaluates them, and stores them for reporting and dashboarding purposes. You can also have the agent publish the run-time statistics to CentraSite. Insight Agents do not affect the runtime flow; clients and servers are unaware that they exist.

Insight Server is a central engine that receives data from the Insight Agents, correlates the data, and analyzes it to determine the information below.

- The logical thread. Because the agents examine each call, Insight Server identifies logical threads accurately and unambiguously. Across application platforms, Insight tracks service calls and correlates them into a complete transaction view.
- The root cause of service failures. A failure at one node usually depends on the upstream path of service calls.
- The amount of time a message spends, the amount of data transmitted, and what errors, if any, are returned from each node. This information is used to detect throughput bottlenecks.

Insight Server makes this information available to the network administrator.