Understanding Software AG Integration & API Products in the Cloud and On Premises

March 2021

Table of Contents

About this Guide

This guide describes Software AG products and how to use them to accomplish your goals.

Document Conventions

Convention	Description
Bold	Identifies elements on a screen.
Narrowfont	Identifies service names and locations in the format *folder.subfolder.service*, APIs, Java classes, methods, properties.
Italic	Identifies: Variables for which you must supply values specific to your own situation or environment. New terms the first time they occur in the text. References to other documentation sources.
Monospace font	Identifies: Text you must type in. Messages displayed by the system. Program code.
{ }	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 and Support

Software AG Documentation Website

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

Software AG Empower Product Support Website

If you do not yet have an account for Empower, send an email to empower@softwareag.com with your name, company, and company email address and request an account.

Understanding Software AG Integration & API Products in the Cloud and On Premises

Once you have an account, you can open Support Incidents online via the eService section of Empower at https://empower.softwareag.com/.

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.

If you have any questions, you can find a local or toll-free number for your country in our Global Support Contact Directory at https://empower.softwareag.com/public_directory.asp and give us a call.

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.

Data Protection

Software AG products provide functionality with respect to processing of personal data according to the EU General Data Protection Regulation (GDPR). Where applicable, appropriate steps are documented in the respective administration documentation.

1 Quick Start

Application Integration, API Management, and B2B Integration in the Cloud Quick Start

Product	Lets you...
Software AG Cloud	Cloud foundation for Software AG's cloud products and portal to those products.
webMethods.io API	Cloud-hosted API Management-as-a-Service (MaaS) platform for securely managing and exposing APIs to your developer and partner community.
webMethods.io B2B	Cloud-hosted Software-as-a-Service (SaaS) product for defining communication channels with trading partners in order to exchange business documents electronically.
webMethods.io Integration	Cloud-hosted integration Platform-as-a-Service (iPaaS) product for automating tasks by connecting apps and services, such as Marketo, Salesforce, Paypal, Cumulocity, Evernote, and Gmail.
webMethods Cloud Container	Cloud-hosted product for deploying integration-related assets from an on-premises Integration Server to the cloud for execution.
CloudStreams	On-premises framework for building and using connectors that connect software as a service (SaaS) providers such as Salesforce.com with on-premises applications such as CRM and ERP.
End-to-End Monitoring	Cloud offering for monitoring business transactions from start to finish as they pass through Software AG's cloud products.

Application Integration On Premises Quick Start

Product	Lets you...
Application Integration Development	
Software AG Designer	Design and test services that make up application integrations. Users can collaborate using CentraSite as a shared database.
webMethods Adapters	Convey data from external resources (for example, SAP, Siebel, PeopleSoft) to Software AG products and vice versa.
EntireX	Easily connect services that run on Integration Server to mission-critical programs written in languages like COBOL or Natural.
ApplinX	Enable web applications to access and integrate data and transactions from core system applications without changing those applications.
Application Integration Local Development and Testing	

Product	Lets you...
Software AG Designer	Eclipse-based graphical development tool. Develop application logic in Java without having to use proprietary webMethods APIs. Develop Integration Server packages locally as Eclipse projects; check package elements and their supporting files in to and out of a version control system (VCS) directly from Software AG Designer. Quickly assemble unit tests to improve overall development quality without custom development.

Application Integration Execution and Monitoring

Integration Server	Execute application integration services.
Universal Messaging	Route application integration data from publishers to subscribers.

Application Integration Execution and Monitoring

Monitor	View clients that called services and service status, view service and document status, and resubmit services and documents.
Optimize for Infrastructure	Monitor system resources using KPIs. Define KPI evaluation rules, discover trends in system behavior, and define actions to take in problem situations. Find rules that were violated, resources that are out of or will go out of compliance. Analyze historical KPI performance to find positive or negative trends.
Insight	Collect data and run-time statistics about service traffic from application servers such as Integration Server and analyze the data to determine the logical thread, root cause of service failures, and throughput bottlenecks.

Application Integration Microservices

Microservices Runtime	Host or deliver microservices as a package that includes a set of related services, interfaces, document types, and triggers that subscribe to topics or queues, or as a set of related packages of this kind.

API Management On Premises Quick Start

Product	Lets you...
API Gateway	Receive requests from consumer applications and forward them to API providers, then return responses from providers to consumer applications. Protect products and their applications, services, and data from malicious attacks from external client applications.
API Portal	Expose and consume APIs. Web-based, self-service portal through which to securely expose APIs to external developers, partners, and other consumers for use in their own apps.
CentraSite	Provide design-time, governance, and run-time aspects of API management.

B2B Integration On Premises Quick Start

Product	Lets you...
Trading Networks	Build and manage network in which trading partners (retailers, manufacturers, suppliers, marketplaces) exchange electronic business documents using automated processes.
eStandards Modules	Define industry-standard transport protocol, provide business document types, specify document exchange rules. Modules support B2B standards for various industries (for example, manufacturing, financial, healthcare).
Optimize for B2B	Monitor Trading Networks transactions in real time.
B2B Analytics	Analyze Trading Networks partner transactions.
ActiveTransfer	Manage transfer of business documents in the trading network.
ActiveTransfer Agent	Transfer files between the remote systems and the central hub that hosts ActiveTransfer Server.

Business Process Management On Premises Quick Start

Product	Lets you...
Business Process Management Design, Implementation, and Execution	
Software AG Designer	Eclipse-based graphical development tool. Document the requirements of the business process for implementors (for example, specify input and output for process steps and how to handle errors, identity humans involved in the process, list KPIs). Implement business process from business process model. Develop services for use in business processes. Develop business rules for use in business processes. Develop tasks for use in business processes.
AgileApps	Build and deploy process-driven application solutions. Application Platform-as-a-Service (aPaaS) product.
Task Engine	Find and manipulate tasks from services or other applications through APIs.
Integration Server	Execute business processes and services.
Process Engine	Control and direct execution of business processes.
Rules Engine	Execute rules invoked by business processes or services.
Task Engine	Execute tasks.
Business Process Management Composite Applications	

Product	Lets you...
Software AG Designer	Design composite applications. Webpages in CAF and Business Console gadgets can invoke services. Webpages can access and display data stored in databases.
My webMethods Server	Host composite applications.
Business Process Management Administration and Monitoring	
Monitor	View process instance status and KPIs; forecast process instance path; stop, suspend, or resume process instances; and resubmit process steps.
Mobile Monitor	View process instances, view and work on tasks from mobile device or tablet.
Task Engine	Assign tasks, start, stop, suspend, resume, delete task instances.
Optimize for Process	Monitor business processes, activities, events, and other business-related data in real time. View and evaluate business data, define rules that trigger alerts when problems arise.
Business Console	Schedule and work on tasks, monitor process and task instances in real time, view trending information about processes on social media, view process analytics, view decision tables invoked by processes, and view processes that include case applications.
Process Engine	View, suspend, resume, stop, restart, or resubmit process instances; cancel, skip, pause, or restart process steps; create, modify, or delete process model stages; project process instance completion, and view average cycle time.
Mobile Business Console	Collaborate on tasks and workflows, monitor process and task instances in real time, and make time-critical decisions from mobile device or tablet.
Business Process Management Mobile Development	
Software AG Designer	Design a user interface and then generate Java code that displays the user interface and responds to user-initiated events, such as clicking a button. Code business logic separately from the user interface, in Java classes created by Mobile Development.
Mobile Designer	Create mobile apps in Java and then automatically convert the apps for any mobile device.
Mobile Support	Create mobile data synchronization solutions that 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.

On-Premises Cross-Product Tools Quick Start

Product	Lets you...
Command Central	Install, patch, configure, manage, and upgrade Software AG products; create database components; and connect products to database components, remotely from one location.
Software AG Installer	Install Software AG products.
Software AG Update Manager	Install fixes and support patches on Software AG products and troubleshoot problems in Software AG product installations.
Deployer and Asset Build Environment	Deploy user-created assets that reside on source runtimes or repositories to target Software AG runtimes (for example, from servers in a development environment to servers in a staging or production environment).
Digital Event Services and Universal Messaging	Foundation for the event-driven architecture (EDA) that enables Software AG products to communicate using events.
Software AG Designer	Eclipse-based graphical development tool. Publish and monitor events.

Analytics & Decisions On Premises Quick Start

Product	Lets you...
Apama	Monitor rapidly moving event streams, detects and analyzes important events and patterns of events, and immediately acts on events of interest according to specifications.
ARIS Process Performance Manager	Draw business process data from data sources (for example, databases, trading partner data stores, third-party applications like SAP or Salesforce).
MashZone NextGen	Visualize data from Software AG data sources (for example, databases, XML files, Apama events, process instances) or from web feeds or web services.
Universal Messaging	Deliver events from producers to consumers.
Zementis Predictive Analytics	Execute predictive models built by machine learning and data mining tools to make real-time predictions.

2 Application Integration, API Management, and B2B Integration in the Cloud

Software AG Cloud

Software AG's cloud products are hosted in the Software AG Cloud. Software AG Cloud is the cloud foundation for Software AG's cloud products and the portal to those products. It offers common cloud platform services such as identity, tenant, and environment management; a scalable subscription model so you can grow at your own pace; and messaging and cloud storage. Software AG Cloud supports infrastructure providers such as Amazon Web Services (AWS) and Microsoft Azure. It supports LDAP and SSO, and is certified under ISO 27001 and the SOC2 Type II standard for security and availability.

You enter Software AG Cloud through a single, easy-to-use portal, where you can sign up for free trials of Software AG's cloud products. You can have multiple Software AG Cloud environments (for example, one environment each for development, test, and production), and you can register, link, and manage those environments from Software AG Cloud. Within each Software AG Cloud environment, you can have one tenant for each Software AG cloud product you have licensed.

webMethods.io API

webMethods.io API is a Software AG Cloud-hosted API Management-as-a-Service (MaaS) platform. webMethods.io API enables you to securely manage and expose APIs to your developer and partner community. webMethods.io API consists of API Gateway, Microgateways, API Portal, AppMesh, and API Engagement. API Engagement is described below. The other products are described in .

API Engagement is a developer outreach program that helps API providers grow their API business and revenues. With API Engagement, API providers can build an ecosystem in which to proactively promote, market, and launch APIs. Developers are the organic evangelists for an organization to spread the word and make an API provider more popular. With API Engagement, providers can engage the developer community to foster collaboration, run campaigns, and evangelize API usage and adoption. For example, API providers can do the following:

■ Create, manage, and judge API hackathons. An API hackathon is a design sprint-like event in which developers, graphic designers, interface designers, project managers, and others collaborate intensively to create a functioning product.

■ Conduct customized beta-programs for testing APIs in a beta state. API providers expose APIs in API Engagement, and API consumers collaboratively test those APIs by building apps. The API consumers then give feedback and raise defects.

webMethods.io Integration

webMethods.io Integration is a Software AG Cloud-hosted Integration Platform-as-a-Service (iPaaS) product. webMethods.io Integration enables you to automate tasks by connecting apps and services, such as Marketo, Salesforce, Paypal, Cumulocity, Evernote, and Gmail. It lets your favorite apps exchange data and talk to each other seamlessly. With webMethods.io Integration, you can create hybrid portfolios with applications and data that are located both on-premises and in the cloud.

webMethods.io Integration offers two different user interfaces for two types of users:

■ Business users can create workflows. A workflow is a connection between two or more web apps or services, and defines the steps of a recurring task that you want to automate. For an example, you might create a workflow that is triggered by the creation of a new note in your Evernote account, translates the note into Spanish using Google Translate Text, and then sends the translated text to your Gmail account. webMethods.io Integration offers hundreds of web services as well as pre-built workflow templates you can use as a starting point for your own complex use cases. You can also import recipes to bootstrap your workflows faster. Workflows run in containers.
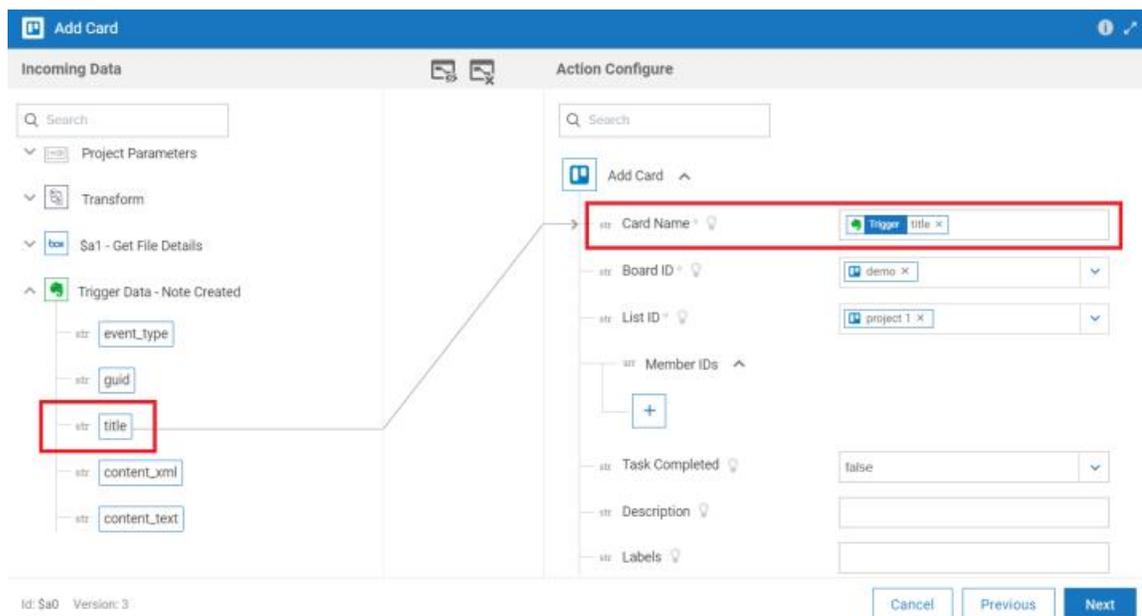
■ Developers can create point-to-point or complex integrations that orchestrate among multiple endpoints. webMethods.io Integration lets you create services, documents, and other assets required by integration flows, and supports the use of conditions, switches, looping, and

custom actions. You can implement publish/subscribe or point-to-point messaging using JMS-based messaging. You can check your flows using debugging, execution logs, execution retry, and error handling features. You can automate the launching of your flows by creating on demand, polling, or event-driven triggers.

To make creation of workflows and integrations easier, webMethods.io Integration provides workflow recipes and flow service recipes, and offers smart mapping, in which an algorithm learns from the pipeline data mappings you create and provides suggestions for mapping similar fields.

webMethods.io Integration comes with hundreds of connectors to popular SaaS, on-premises, and IoT apps. You can also use CloudStreams connectors with webMethods.io Integration, or use the webMethods.io Integration Connector Builder Framework to develop your own connectors. In your custom connectors, you can map fields, use pre-defined transformers for manipulating values such as strings, dates, numerical values, and more, and create your own transformers using Node.js.

webMethods.io Integration users can collaborate, governing projects using permission-based folder access, versioning, and locking and unlocking to prevent unintended changes. The contributions of multiple users also amplifies the usefulness of features like smart mapping.

webMethods.io Integration administrators can define and administer users, configure access rights, and enable and disable accounts. Administrators can enable or disable individual workflow actions, activate and deactivate entire workflows, monitor workflow execution, and set up email-based alert rules to send notifications of workflow errors. They can review tenant audit and activity logs.

webMethods.io Integration also offers an add-on named webMethods.io Embed that provides an SDK for creating workflows and integration flows your users can embed in their own applications. For example, your HR department might create a workflow that downloads pre-travel approval forms, uploads completed forms, and sends a notification of the outcome (approval, rejection, error, and so on). Your employees could embed this workflow in their mobile devices.

webMethods.io B2B

webMethods.io B2B is aSoftware AG Cloud-hosted Software-as-a-Service (SaaS) product. webMethods.io B2B enables enterprises to define communication channels with trading partners in order to exchange business documents electronically. Business documents typically include purchase orders, order statuses, purchase order acknowledgments, invoices, and other domain-specific business documents. webMethods.io B2B supports XML and EDI, covering hundreds of versions and more than 15,000 transaction definitions, including ANSI X12, EDIFACT, VICS, EANCOM, and ODETTE, along-with AS2 and HTTP protocols.

With webMethods.io B2B, you can:

■ Quickly onboard trading partners, and centrally manage all partner details. Partners can also onboard themselves.

■ Exchange business documents with trading partners in the cloud or on premise. Documents can be delivered individually or in batches.

■ Enable or disable document delivery to a trading partner as necessary.

■ Define rules that govern how to identify and process business documents.

■ Create trading partner agreements that define specific parameters trading partners must use when exchanging business documents.

■ Configure document delivery to trading partners by configuring queues. You can also enable, disable, drain, and suspend a queue.

■ Monitor all transactions that pass through webMethods.io B2B.

■ Search for transactions based on document attributes.

■ Download live transactions and activity logs on your local system.

■ Integrate with webMethods.io Integration for orchestration using pre-defined applications.
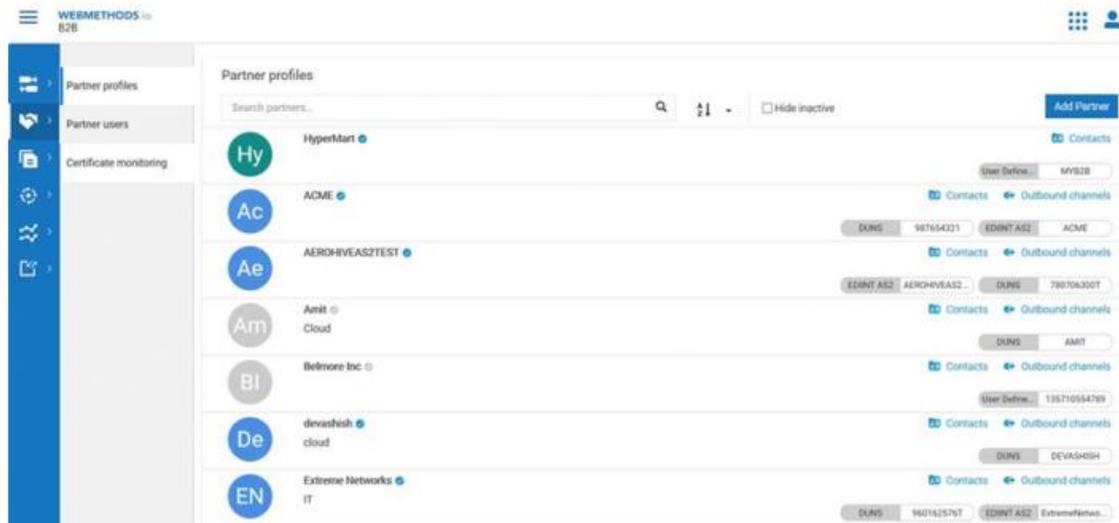
Cloud Container

First you create *solutions* in your Cloud Container tenant. Each solution includes a landscape comprising four Docker containers, orchestrated by Kubernetes, that will execute the deployed application integrations. There is one Docker image for each of the following:

■ One or more Integration Server instances.

■ One Universal Messaging instance.

■ One or more BigMemory Max instances.
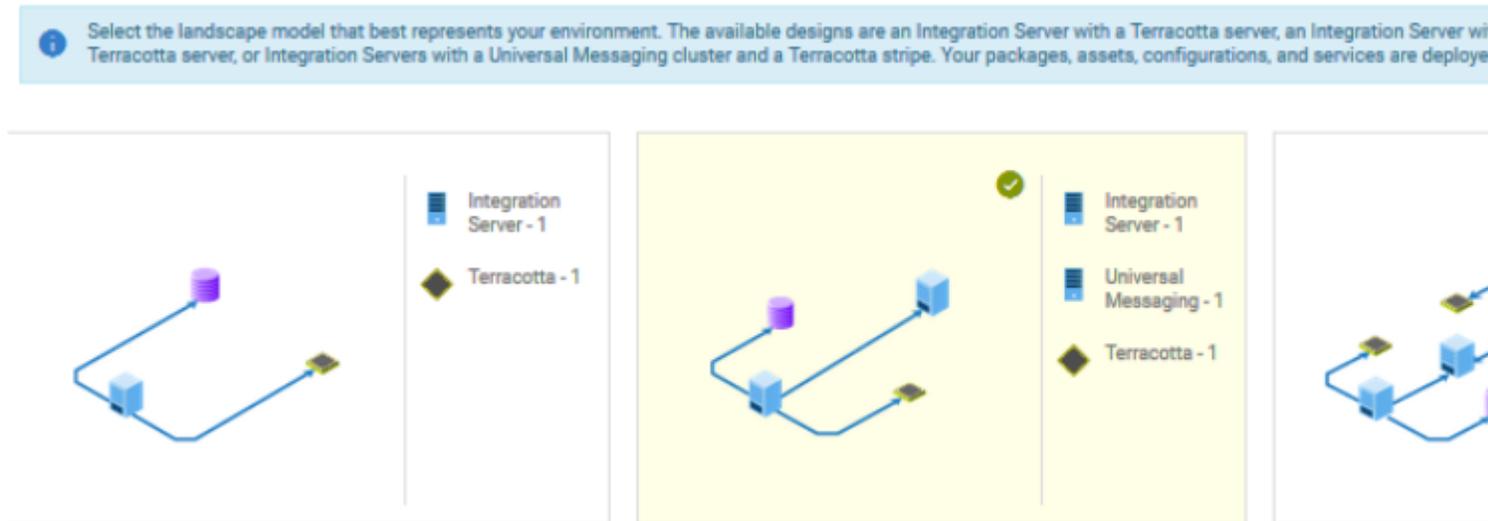
You then use Software AG Designer to deploy application integration-related assets such as Integration Server, webMethods Adapter for JDBC, and CloudStreams services and documents from an on-premises Integration Server into a solution. You can also deploy the on-premises product configurations for use with the products in the solution's landscape, substituting variables such as port numbers. You can browse Integration Server packages, services, and APIs hosted in Cloud Container from Software AG Designer, and you can then execute the application integrations on the cloud-hosted Integration Servers. For information on application integrations, Software AG Designer, Integration Server, Universal Messaging, and BigMemory Max, see "Application Integration On Premises" on page 23.

Software AG Designer supports testing features that enable you to debug assets, mock services, create test cases that can assert error messages returned by a service, generate a test from service run results, and view and access lists of all test suites and test cases for Integration Server assets.

webmethods.io End to End Monitoring

webmethods.io End to End Monitoring is a Software AG Cloud-hosted product that enables you to monitor business transactions from start to finish as they pass through Software AG's cloud products. It offers a dashboard on which you can see the business transactions list and associated details such as overall transaction status, topology, transaction duration, and error rate. You can create rules that define a set of conditions which, when met, indicate that a violation has occurred. Rule violations trigger alerts in the form of an on-screen notifications, emails, or both.
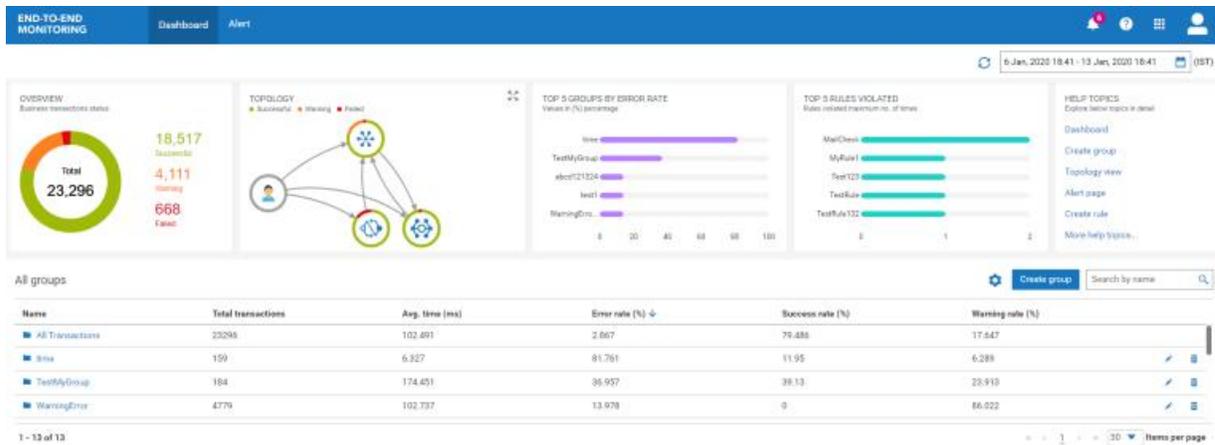
CloudStreams

CloudStreams is an on-premises framework that enables you to build and use connectors that connect software as a service (SaaS) providers such as Salesforce.com with on-premises applications such as CRM and ERP. CloudStreams comes with pre-built connectors.

You use Software AG Designer to develop CloudStreams connectors. The table below describes the types of connectors you can develop.

Connector	Description
Connect from SaaS providers to on-premises applications	You create virtual services and service governance policies that perform user-defined security checks and processing, send requests to on-premises applications, and handle the responses. You can also implement backend-specific inbound streaming.
Connect from on-premises applications to SaaS providers	You create cloud connector services that perform SOAP operations or access REST resources, and a special kind of virtual service, called a connector virtual service, that sends requests to SaaS providers and handles the responses.

The connectors and services run on CloudStreams Server, which in turn runs on Integration Server. CloudStreams Server also collects run-time performance metrics and events for your CloudStreams connectors that you can view on the CloudStreams Analytics dashboard.

You can use pre-built and user-created CloudStreams connectors with webMethods.io Integration and Cloud Container.
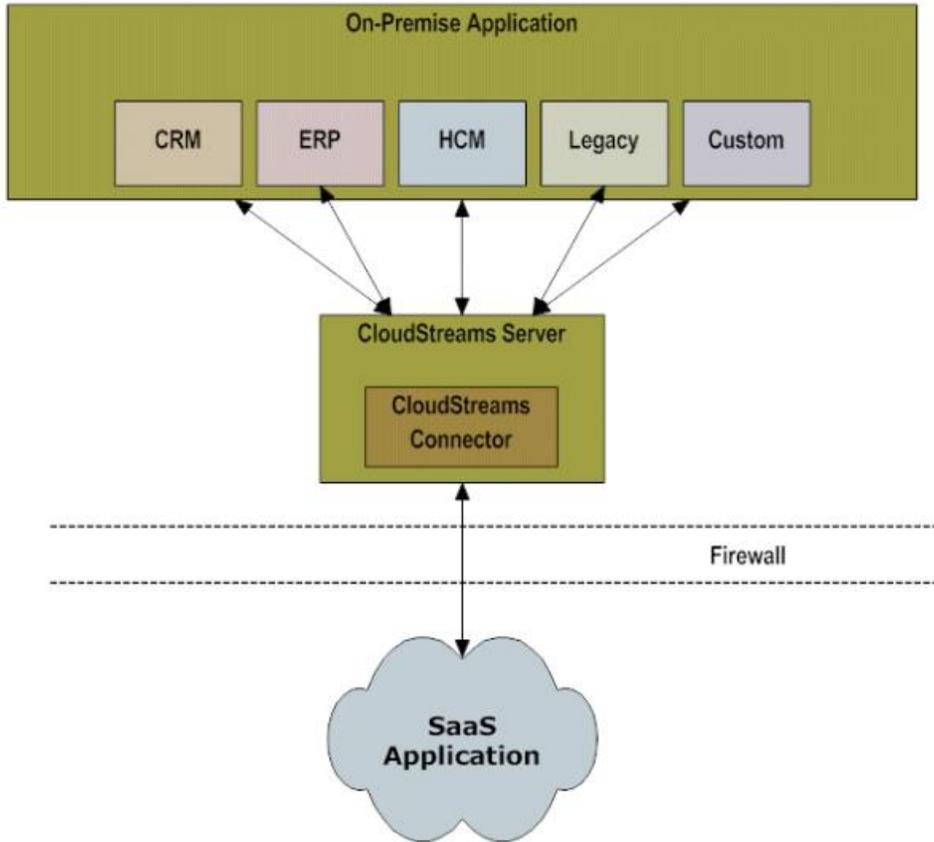
3 Application Integration On Premises

Application Integrations

Application integrations enable disparate resources to share business data. Resources include software applications such as SAP and Siebel, and systems such as databases and mainframe programs. Common integration patterns 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.
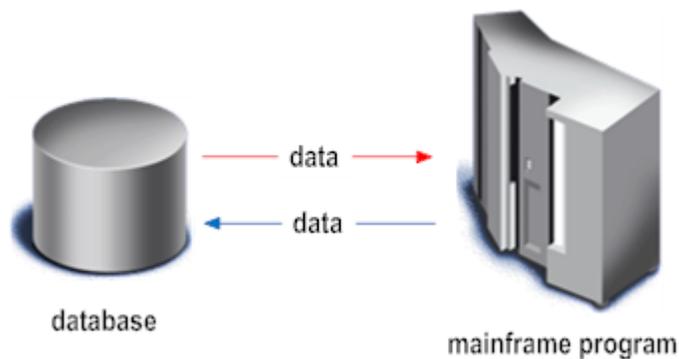
■ **Propagation**. For example, a human resources (HR) person enters data for a new employee in a composite application and an application integration propagates the data to a human resources (HR) database, a retirement plan mainframe program, and an employee benefits application.
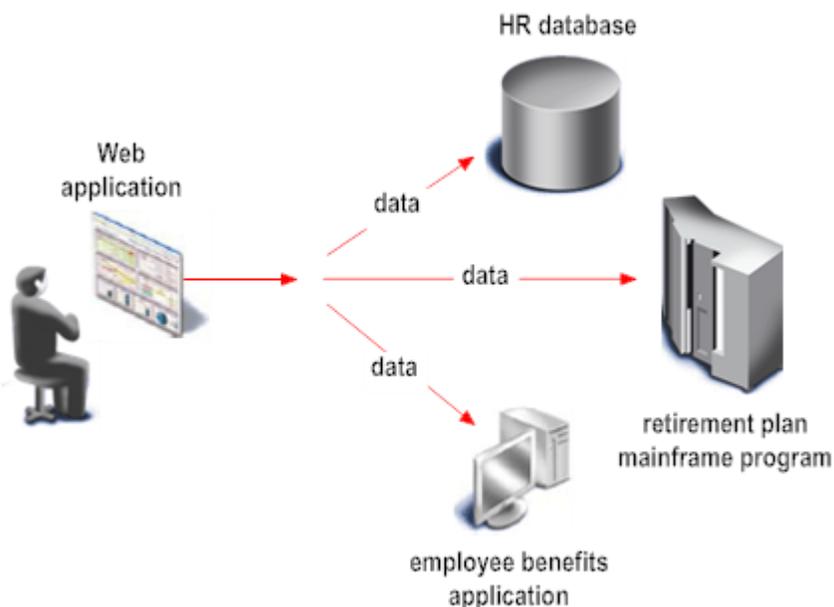
■ **Composition**. For example, an HR person requests a report on an employee through a composite application. An application integration gathers the data for the report from an HR database,

a retirement plan mainframe program, and an employee benefits application, and then returns the data to the web application for display in report format.
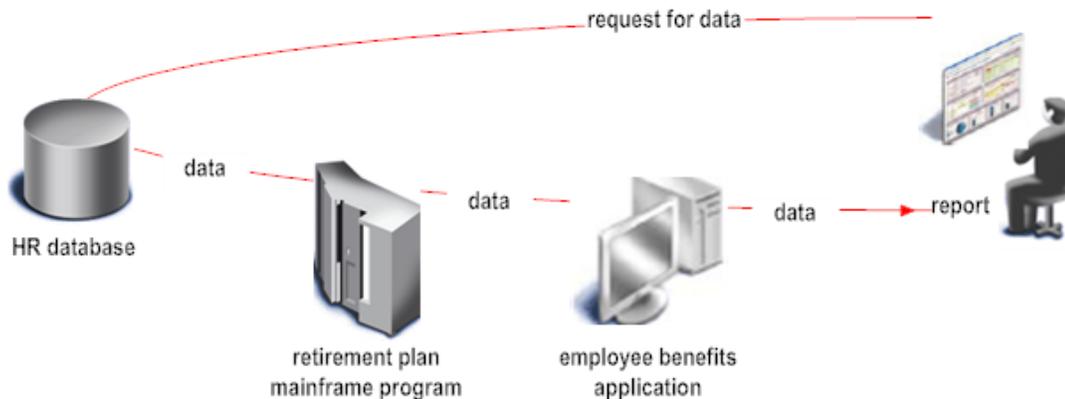
Developing Application Integrations

The primary products you use to develop application integrations are Software AG Designer, Application Platform, webMethods adapters, Integration Server, 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 application 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 Software AG Designer, using webMethods Flow language or other languages such as Java. When you work in Software AG Designer, you are always connected to an Integration Server, a run-time server whose primary function in application integrations is to execute services. Through Software AG Designer you build and edit services directly on the Integration Server. In addition, Integration Server comes with a library of built-in services that you can use in your application integrations.

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 compare differences between Flow services, document types, JMS triggers, and adapter services and connections, and you can merge the differences between two Flow services or two document types. You can set up audit logging for services and documents.

The organizational unit in Integration Server is a *package*. A package contains a set of services and related files, such as specifications and document types. Typically the services and files are grouped by function or application; for example, you might put all purchasing-related services in a package called PurchaseOrderMgt and all time-reporting services into a package named TimeCards. You can easily manage (enable, disable, delete, and so on) the entire contents of a package with one action.

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).

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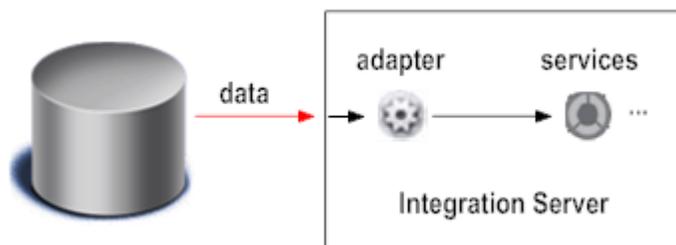
Adapters convey data from Software AG products to resources. For example, an Adapter for JDBC service can receive data from an Integration Server service, transform it from the Software AG internal format into the format required by the database, and insert it into the database.
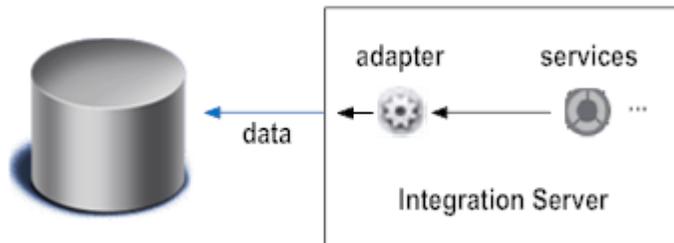
EntireX

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.

■ 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 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 Software AG Designer.
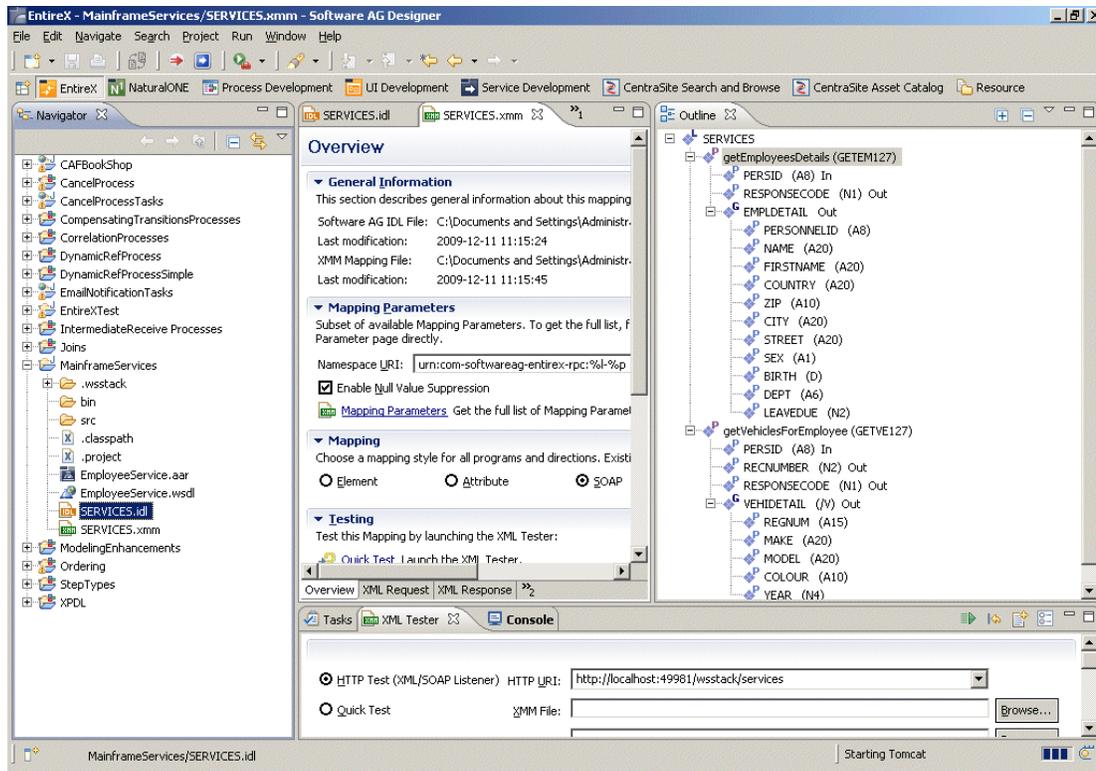
When you run the adapter service on Integration Server, it invokes the CICS transaction on the mainframe side using values you provide for the transaction's extracted inputs. You can use the adapter service in application integrations as you would any other Integration Server service. Software AG Designer 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. You can set up zero server footprint scenarios for CICS, IMS, and AS/400. This may be useful when integration spans different organizational units.

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 Software AG Designer for these types of development work, as shown below.
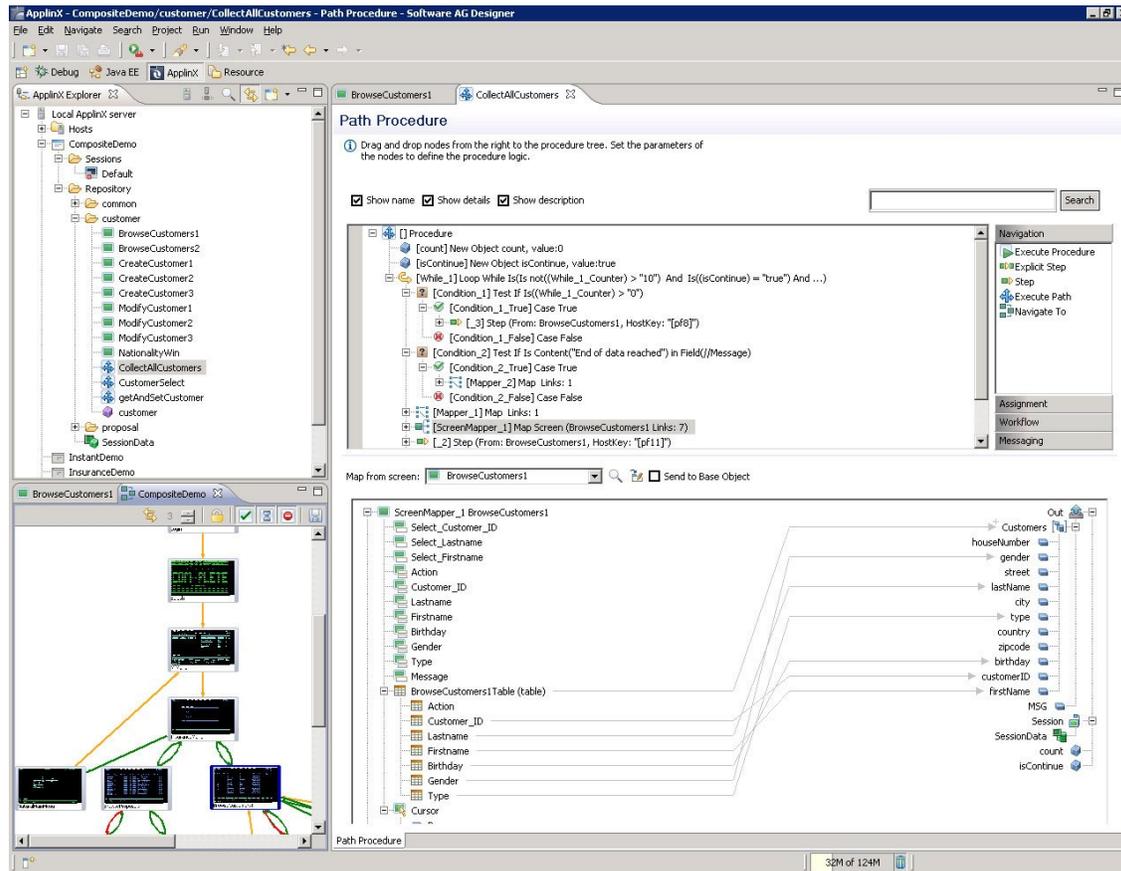
Web enablement allows you to turn existing host 'green screens' into web interfaces. You can achieve a true web look and feel without touching existing applications or changing any code. ApplinX offers the web enablement solutions described below.

Solution	Description
Instant	Enables you to turn core system applications into modern-looking web applications with little or no coding. In Software AG Designer, you can apply the same design to multiple screens. You can use transformations to convert host screen patterns into web components (for example, transforming host function keys into hyperlinks, buttons, or images, or converting input fields into GUI elements such as combo boxes, radio buttons, or check boxes.
HTML emulation	Instantly turn a host terminal emulation into a web browser terminal emulation that provides host key and print support and maintains existing color schemes. Available in .NET and J2EE environments.
Composite	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

Solution	Description
	using the ApplinX Base Objects API for customizing the web framework. Available in .NET and J2EE environments.

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 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.

Developing and Testing Integration Server Assets Locally

The collection of products that enable you to develop and test Integration Server assets locally is called webMethods Integration Server DevOps Edition. DevOps Edition contains features that improve development productivity and enable developers to locally develop, build, change, and test the services that eventually lead to continuous integration.

DevOps Edition includes Software AG Designer, Local Version Control Integration, Unit Test Framework, and Application Platform.

Local Version Control Integration

Local Version Control Integration (LVCI) is a feature in the Software AG Designer Service Development plug-in that you can use to develop Integration Server packages as Eclipse projects locally. By default, the local development part of this feature is enabled by an installation of Integration Server in the same root directory as Software AG Designer. You can also use an Integration Server running in a Docker container as the local development server for this feature. This feature also enables you to check package elements and their supporting files into and out of a version control system (VCS) directly from Software AG Designer.

Unit Test Framework

Software AG Designer provides a range of tools to assist during the testing and debugging phases of development. Unit Test Framework is a service unit and regression testing tool that enables service developers to quickly assemble unit tests and improve overall development quality without custom development. Unit Test Framework facilitates automated unit and regression testing of services, and provides a mechanism for creating automated tools for continuous integration and

delivery. Unit Test Framework can integrate with JUnit, leveraging a standard unit testing framework that already works well with continuous integration tools. In addition, Unit Test Framework provides a Java API for advanced users to create JUnit test cases. You can automate build deployment and execution of Jenkins tasks from Unit Test Framework.

The following table explains the types of testing you can do in Software AG Designer.

Testing	Description
Unit	Design, build, and execute unit test cases in the Software AG Designer Eclipse user interface. You can also execute test cases externally using Ant scripts. You can test Flow services with input values you specify manually, inspect the results, and investigate errors.
Mock	Simulate interaction with resources that are unavailable, or whose data is not consistent, for test purposes. Have a test case execute a mocked service using input values you specify, invoke an intermediary service instead of the mocked service, replace the output of the mocked service with a specified exception, or use a custom Java class to implement the mocked service.
Regression	Save test cases, along with their inputs and outputs, in XML files, and then rerun these reusable artifacts to make sure your latest changes do not re-introduce errors that were fixed in earlier versions.
Code coverage	Record and analyze which flow steps and map actions of your Flow services are executed during a particular test launch. Helps to identify untested steps and improve the corresponding tests.
Suite coverage, execution model, service coverage	Record and analyze which flow steps and map actions of your Flow services are executed during a particular test launch. Provides user-friendly reporting with coverage reports for suite coverage, execution model, and service coverage.

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 or My webMethods Server.

You develop the application logic using Software AG Designer. You can also develop web UI applications based on Java servlet technology, and set authentication and authorization requirements in these applications. You can create JUnit tests and run them in Software AG Designer to verify application functionality. The tests are executed on a running Integration Server or My webMethods Server, and the execution results are displayed on the Software AG Designer JUnit view.

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, 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.

Executing Application Integrations

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

Integration Server

Integration Server is a run-time server that 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.

The following table lists the standards supported by Integration Server and examples of each standard.

Supported Standards	Examples
Transport standards	HTTP, HTTPS, FTP, FTPS, SMTP, TLS
Message formats	MIME, S/MIME
Data standards	XML and XML schema, custom flat file formats with delimited fixed- or variable-length records, JSON and JSON schema, GraphQL
Protocols	AMQP, JMS, Kerberos, MQTT, OData, OpenID Connect, SFTP, SOAP, SOCKS, Websocket, XML RPC
Specifications	Swagger, Web Service Description Language (WSDL)
Integration patterns	REST, web services, event-driven, request-reply
Authorizations	OAuth, JWT

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

Integration Server offers a web browser-based user interface for configuring and administering Integration Server, and for monitoring running services. The user interface includes an alerting and notification framework that sends alerts about password expiry, certificate expiry, configuration changes that require a server restart, and so on. Integration Server also offers a REST-based API so DevOps personnel can create scripts to automate Integration Server administration,, automate deployment, and support continuous integration and deployment scripts. Integration Server also provides a utility to access administrator API operations via the command line interface so developers can create advanced devops scripts and automate deployment and management of Integration Server environments. .

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, and message queues, as well as support for JMS. Universal Messaging also supports a significant part of the AMQP 1.0 standard wire protocol for messaging, specifically the parts that are exposed through the JMS API.

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. Integration Server can encode/decode those documents as protocol buffers.

Monitoring Application Integrations

The products that execute application integrations include Monitor, Optimize for Infrastructure, and Insight.

Integration Server Administrator

Integration Server offers a web browser-based user interface calledIntegration Server Administrator.Integration Server Administrator provides access to all administrative tasks through an easy to use, graphically rich, tab-based interface that includes accessibility, tools, and server controls. You can see all key Integration Server Administrator metrics via an interactive dashboard that shows system status, health, usage patterns, and the overall performance of Integration Server, including JVM, usage, services, and API-related metrics. You can monitor flow services, coded services such as Java services, and documents data, including key metrics around failure, trigger retries, and error details.

Integration Server also offers a REST-based administrator API that duplicates all functions in the Integration Server Administrator GUI. This enables developers to automate deployment and support continuous integration and deployment scripts.

Optimize for Infrastructure

Optimize for Infrastructure enables you to monitor the system resources involved in application integrations. You identify the resources for which to collect data, configure KPIs to monitor the collected data, and define rules that evaluate the data. You can view and analyze the collected

monitoring data to find rules that were violated and resources that are out of compliance. Optimize for Infrastructure can notify you when a problem occurs and take actions you define.

You can analyze historical KPI performance to find positive or negative trends. Optimize for Infrastructure can predict resources that will go out of compliance in the future and notify you of potential problems based on trends.

Insight

Insight enables you to:

- 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.

An Insight Agent runs continuously on application servers such as Integration Servers that host web services. The agent collects data and run-time statistics about service traffic from the application servers, detects incoming and outgoing web service calls, and reports the data to Insight Server. The agent can also 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 and analyzes it to determine the information below.

- The logical thread. Because the agents examine each call, Insight Serveridentifies logical threads accurately and unambiguously. Across application platforms, Insight Server 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.

- Throughput bottlenecks. The amount of time a message spends, the amount of data transmitted, and what errors, if any, are returned from each node are data that help detect blockages.

Insight Server makes this information available to the network administrator.

Hosting and Delivering Microservices

Microservices is an architectural style for developing an application as a suite of small services called *microservices* that perform a single business function. A microservice is a collection of operations, effectively an API, that is remotely called, yet that is meant to be independently deployable. It can be implemented in various ways, including as a set of services or as event and channel definitions.

Microservices Runtime enables you to deliver microservices as a package that includes a set of related services, interfaces, document types, and triggers that subscribe to topics or queues, or as a set of related packages of this kind (for example, five packages relating to Human Resources functions).

Each microservice can run in its own Microservices Runtime and can communicate with lightweight mechanisms such as an HTTP resource API. However, you can also execute multiple microservices in the same Microservices Runtime. This solution enables you to separate microservices when needed, but also to group them when necessary. Suppose you have two microservices that need to be scaled together in similar ways (that is, when you need a new instance of one, you need a new instance of the other). If you discover that one microservice is more heavily loaded than the other, or needs to be enhanced or updated more often, you could deploy the two microservices to separate Microservices Runtime. If both microservices tend to be updated at the same time, you could cohost them in the same Microservices Runtime.

Microservices Runtime works with Istio Service Mesh. The microservices hosted on Microservices Runtime can be managed by the Istio Service Mesh control plane and can work with Istio components such as Envoy.

Microservices Runtime is fully compatible with Integration Server and can host services you develop using Software AG Designer and Integration Server. While Microservices Runtime is optimized to have a reduced disk and memory footprint, you can convert it into a full Integration Server by installing additional modules, such as support for an external database.

Software AG also offers a lightweight development tool called webMethods Service Designer, which comprises the Service Development functionality from Software AG Designer, Microservices Runtime, and Unit Test Framework.

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 to change employee information using a composite application, and you want to propagate the changes to the human resources (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.
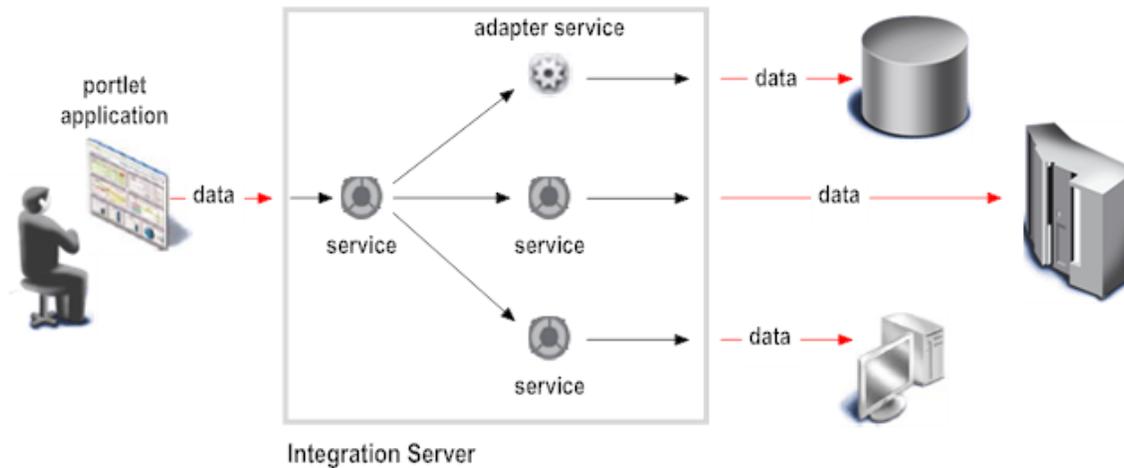
Composition

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, maps the data from service to service, and produces the reply.

■ 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.

Understanding Software AG Integration & API Products in the Cloud and On Premises
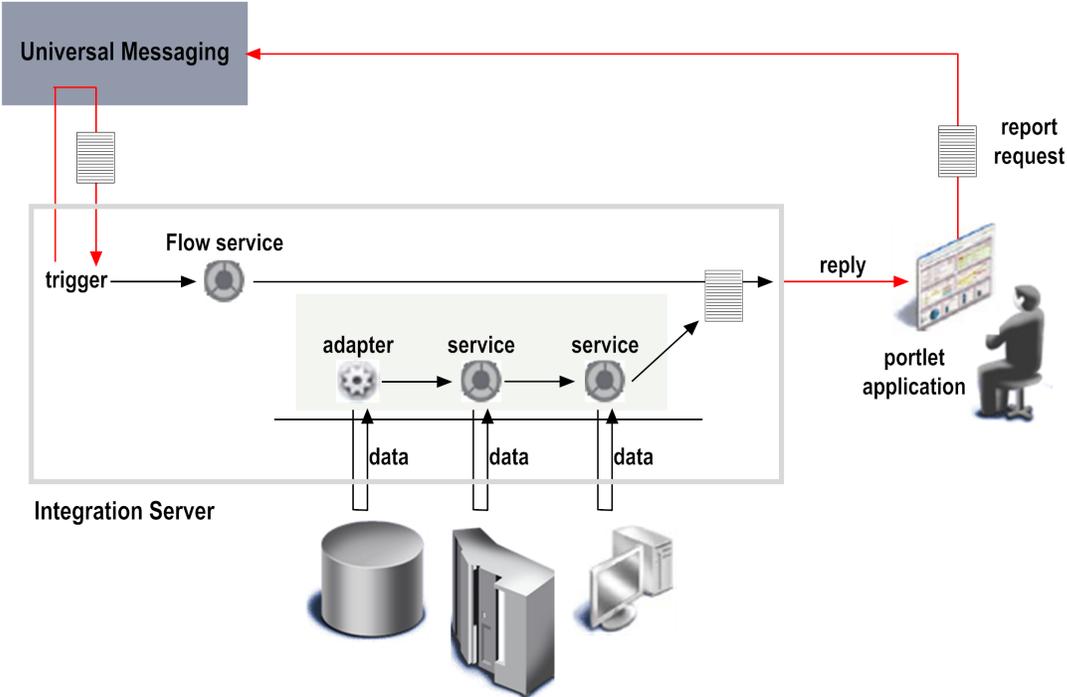
4 API Management On Premises

API Management

Organizations are increasingly publishing application programming interfaces (APIs) in order to share frequently requested real-time data with internal or external consumers. For example, the government of a country might create and publish APIs that enable the public or press to access information about police forces and crime statistics in different areas. API management is a system in which an organization, or API provider, creates APIs, governs the API design life-cycle, governs and secures API runtime execution, publishes APIs for API consumers, and monitors API usage.

Governing the API Design Life-cycle

CentraSite

CentraSite is a standards-based registry and repository that enables you to govern the API design life-cycle. You use CentraSite to control design-time events through policies you define. For example, you could define a policy that new APIs must be approved by certain people before being accepted into the registry, and that modifications to APIs in the registry must be approved by certain people before going into production. You can also use policies to define review and approval processes, perform quality assurance tests, and issue alerts when new APIs are submitted or when existing APIs have been modified.

Governing and Securing API Runtime Execution

API Gateway

API Gateway securely exposes your APIs to consumer, third-party developers, and other partners for use on the Internet, mobile, and IoT applications. API Gateway enables API providers to do the following:

- Publish your APIs for consumption inside and outside your organization.

- Define and enforce policies around aspects of runtime execution, such as user access, required access protocol and keys, data masking, and so on.

- Enforce standards and practices as every API moves through its life-cycle.

- Protect your APIs from unauthorized and malicious users, while also having full control and visibility into who is accessing your APIs. The built-in approval process workflow lets you manage third-party access requests.

- Automatically synchronize events for applications (consumers) that are registered to use APIs in API Gateway across multiple data centers (including geographically distributed ones) even if they are not clustered.

- Define and track API versions. Multiple versions can co-exist, and older versions can be retired over time.

- Monetize your APIs by creating API plans and packaged offerings.

- View usage statistics and dashboards to make sure your API service level agreements are met, and to understand how your APIs are being accessed and used. You can also build custom analytics dashboards.

- Capture design-time events data and use it to notify users and provide analytical dashboards. The data can also be sent to destinations such as API Portal as well as external solutions like Elasticsearch, relational databases, and files.

- Monitor runtime performance and send alerts when performance conditions are violated, optionally based on an SLA.

In an API management system, one API Gateway sits behind an internal firewall, while another API Gateway acts as intermediary between the internal API Gateway and external clients. In this way, API Gateway protects your applications, services, and data from malicious attacks from outside your organization.

API Gateway now provides support for AppMesh, in which API Gateway acts as the controlling and monitoring body. Microgateway acts as the body enforcing policies defined in API Gateway, acting as a sidecar to microservices. Users can now configure connection to Kubernetes clusters where service mesh resides. API Gateway also supports Istio-enabled service meshes as well as plain Kubernetes.

Microgateway

API Gateway is required to manage and monitor microservice landscapes. However, the volume of communication among microservices can cause network latency and security issues for API Gateways in such environments. Lightweight gateways are therefore helpful for supplementing API Gateway and preventing bottlenecks.

Microgateway is an independent offering—lightweight, agile, and fast. It works with API Gateway or as a standalone solution to control API access to microservices in a distributed environment and supports service mesh architectures offering independent and sidecar deployments.

Microgatewayenables microservices to communicate with each other directly. Microgateway enforces policies that perform authentication, traffic monitoring, and traffic management. You can enforce protection policies on Microgateway to ensure a secure communication channel among microservices. Microgateway works withAPI Gateway or as a standalone solution to control API access to microservices in a distributed environment.

A common scenario is to spread out your API traffic burden by using API Gateway to handle less trusted external, or north-south, traffic while using Microgateways to handle more trusted internal, or east-west, traffic.

Publishing APIs

API Portal

API Portal is a web browser-based, self-service portal that API providers and product managers can use to collaborate with consumers. Developers have a place to discover, test, and sign up for

your APIs, and to publish REST and SOAP APIs. API Portal not only exposes API documentation to developers inside and outside your organization, it manages the developer on-boarding process, and lets the developers use the exposed APIs for new and creative uses.

The developer portal is the public face of your API offerings and enables you to create and grow your network of developers and partners. The portal offers the capabilities described below.

- You can customize the portal using logos, skins, corporate colors and fonts, and more; change layouts for the portal landing page and API Gallery; and customize the portal's privacy and usage terms.

- You can view statistics that provide information about where 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.

- API consumers can quickly find the APIs they need in the API Gallery or by running a simple keyword search. webMethods.io API provides code samples with expected error/return codes and descriptions so consumers can easily try out APIs.

- API consumers can rate APIs, follow and un-follow updates to APIs, and participate in discussions with other developers.

- API consumers can upload, describe, and promote applications built using your published APIs.

Monitoring and Managing Microservices

AppMesh enables you to monitor, manage, and modify applications built from microservices. Designed to plug in to any service mesh, AppMesh consists of a set of Microgateways controlled by a central API Gateway. AppMesh allows you to apply business rules to drive application-specific behavior, create application-level governance and security policies, add new services and capabilities, and perform context-aware application routing and orchestration, all without changing existing microservices or underlying code. AppMesh makes an API signature available for every service in the mesh, enabling reuse across the service landscape. The AppMesh dashboard lets you trace the path a transaction takes through your service mesh and understand how your app is being used.

5 B2B Integration On Premises

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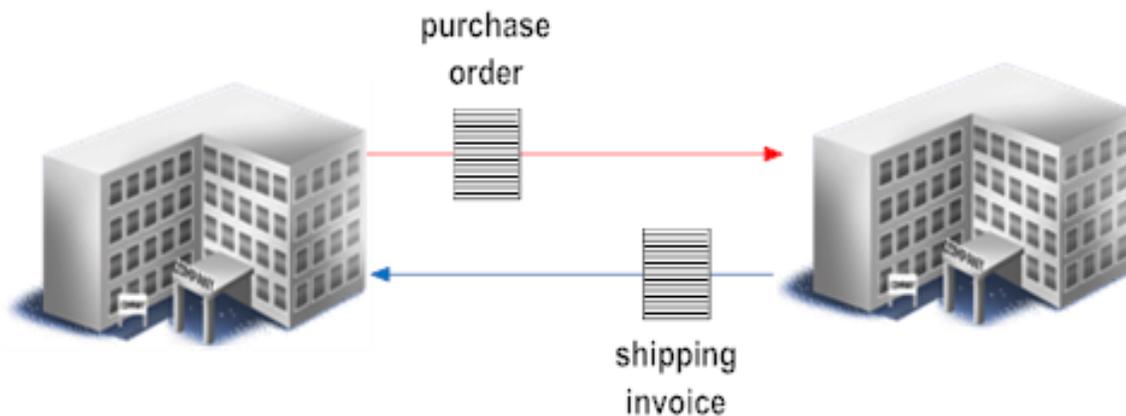
Some industries have developed B2B standards for exchanging business documents. For example, many manufacturing companies use the EDI messaging standard to conduct business electronically. A B2B standard typically defines the business document types and transport protocols that trading partners need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

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 web 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 following table describes elements you need to define and link trading partners.

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

Element	Purpose
	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. **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.
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 provides a flexible schedule-based ability to suspend document deliveries to partners. Administrators can proactively schedule document delivery suspensions as necessary to accommodate planned partner outages.

Trading Networks can selectively expose APIs available in API Gateway. Administrators can manage access granted to trading partners, and view and monitor API invocations.

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.

The following table lists the eStandards Modules Software AG provides for various industries.

Industry	eStandards Modules
Manufacturing	RosettaNet, Chem, PIDX, papiNet

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

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 products you use to manage the transfer of business documents in your trading network are ActiveTransfer and ActiveTransfer Agent.

ActiveTransfer

ActiveTransfer offers a web 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 and supports virus scanning for file uploads. ActiveTransfer Server supports encryption and decryption of files and downloads. For example, files can be decrypted as they get uploaded, and then can be stored in a file system encrypted using a different set of keys. ActiveTransfer Server also supports two-fact authentication. You can share files with external users through ActiveTransfer Gateway without compromising security. In addition, you can configure more granular access levels for an external user in order to limit the range of actions possible on the files shared, such as upload, upload, download, rename, delete, folder creation, and folder deletion.

ActiveTransfer is tightly integrated with Trading Networks so you can easily manage and monitor files that traverse the two products. You can use ActiveTransfer as the delivery method for Trading Networks and specify the remote end point to which ActiveTransfer should transfer the file. You can monitor documents delivered by ActiveTransfer in the Trading Networks user interface.

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 Agent

ActiveTransfer Agent is a component you install on remote systems so you can transfer files between the remote systems and the central hub that hosts ActiveTransfer Server. For file uploads, ActiveTransfer Agent can access files from the file system and connect to a remote server to fetch files. You can install ActiveTransfer Agent on remote systems within or outside your organization's firewall.

The central ActiveTransfer Server manages and monitors ActiveTransfer Agents. ActiveTransfer Server also manages all configurations related to file transfers through events that you push directly to one or more agents. Event definitions include a variety of options, such as the ability to rename files at source or destination, delete original files, compress files, and verify checksum to maintain file integrity during transfers. You can further simplify your setup by grouping agents that have the same file transfer characteristics for an event and applying the event across those agents. ActiveTransfer Server also tracks all activities related to file transfers involving an agent, including authentication, file transfer operations, and the status of each action.

ActiveTransfer Agent supports error handing in case of event failures. The error handling mechanisms include executing events on ActiveTransfer Server, sending email notifications, and executing Integration Server services.

6 In-Memory Data Management On Premises

Terracotta

Terracotta is a data management system that helps you access large amounts of data quickly, in a high availability configuration. It consists of the components described below.

Terracotta Server Array

A Terracotta Server Array (TSA) is a cluster of Terracotta servers. You arrange Terracotta servers into stripes that each include one server in active status and one or more servers in passive status. The active server communicates with your clients, while the passive servers can take over if the active server fails. A TSA can have a single server, a basic two-server tandem for high availability (HA), or a multi-server array for configurable scale, high performance, and deep failover coverage.

Ehcache

You use the Ehcache component to create, use, and manage application-driven data caches. The Ehcache component consists of a client library and related server elements. Applications interact with Ehcache-resident data caches through a Java API implemented by the Ehcache client library. You integrate the Ehcache client library into your applications so they can create two types of cache:

- One cache for your Terracotta Server Array, distributed across the stripes. Clients create a cache entity on every server in the cluster, and then use the cache entity on the active server in each stripe to communicate with the distributed cache.

- Local caches. Clients create caches in their own memory for their own use only

TCStore

You use the TCStore component to create, use, and manage application-driven data stores. The TCStore component consists of a client library and related server elements. Applications interact with TCStore-resident data stores through a Java API implemented by the TCStore client library. You integrate the TCStore client library into your applications so they can communicate with an operational store. You can use the store in two ways:

- As a system of record in which to persist data.

- As a temporary scratch area where data is transmogrified and then dumped.

You can store any type of data in the operational store. You can use off-heap memory in addition to Java heap memory, which enables you to store huge amounts of data. Because off-heap memory is not subject to garbage collection, overhead is reduced and operations on the data are almost as fast as in the Java heap part of the store.

Terracotta Management Console

Terracotta Management Console (TMC) is a browser-based GUI that provides a complete view of your TSA and connected clients. The TMC is served by the Terracotta Management Server (TMS),

a standard Java web application and bundled web container that gathers information from throughout the cluster and provides it to the TMC for display.

Terracotta Cluster Tool

The Terracotta cluster tool is a command-line utility that allows you to perform a variety of server and cluster management tasks. For example, you can use the cluster tool to do the following:

Configure, re-configure, or shut down a cluster.

Obtain the status and configuration information of running servers, and take backups of running servers.

Promote a suspended server on startup or failover.

Clients

The Terracotta cluster can be viewed topologically as a collection of clients communicating with a TSA. Within the overall structure of the cluster, clients represent the application end-points. They work independently but communicate through the active servers of the stripes to which they are connected.

7 Business Process Management On Premises

Intelligent Automation

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 product that business process designers use to create business process models is Software AG Designer.

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 of Software AG Designer.
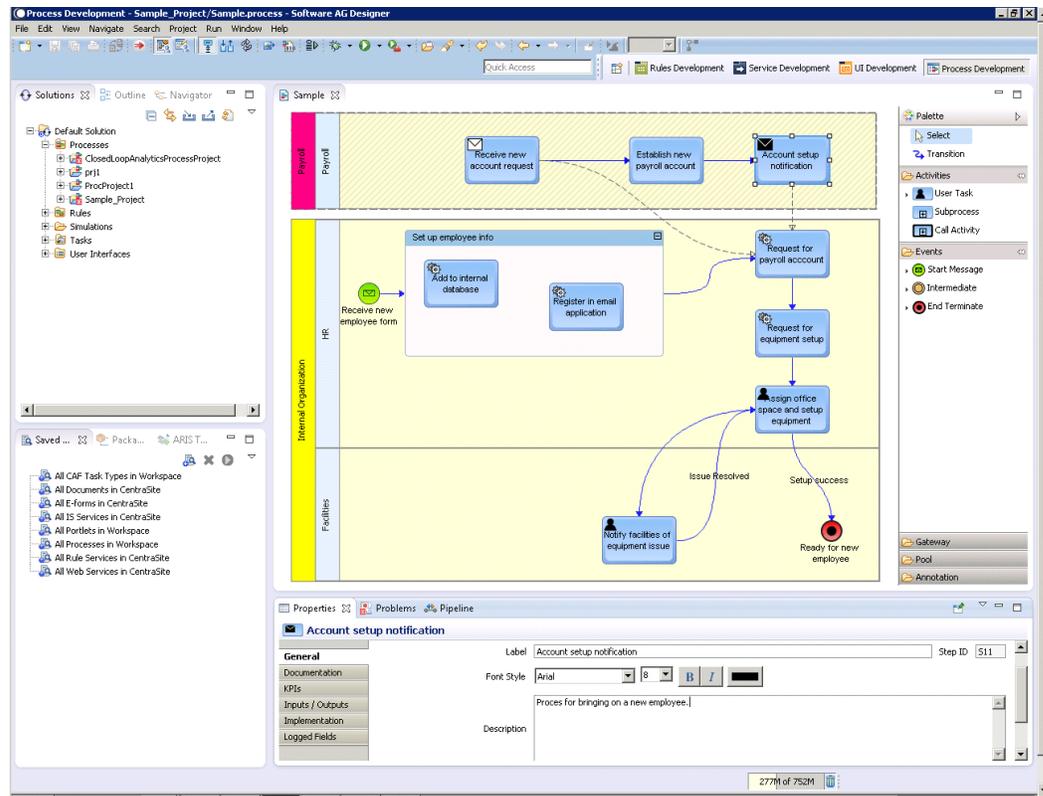
Implementing Business Process Models

The primary product that business process developers use to implement business process models is Software AG Designer. Developers might also use Task Engine and AgileApps.

Business processes might include of services (automated steps), tasks requiring human intervention, and business rules.

Software AG Designer

Business process developers receive a new business process model from a business analyst and use the Software AG Designer Process Development perspective to implement business process models. This perspective offers an extensive set of programming tools that enable a technical user to focus on the detailed implementation of a business process model.

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, and output data from web services and services such as database query services.

- Subscribe to business documents on or publish business documents to Universal Messaging.

- Invoke services such as flow services, web services, or rule services, or adapter services that in turn invoke programs on mainframes and UNIX systems.

- Invoke rules, tasks, other business processes, or Trading Networks.

- Publish data for other business processes and services to consume.

- Assign activities, called *tasks,* to a user or group of users to perform. Tasks in turn can invoke rules, rule sets, and services.

- Queue case instances in AgileApps.

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.

Developing Rules

Developers can develop business rules using the Software AG Designer Business Rules Development perspective. Rules can do the following:

- Invoke services.

- Operate on processes (for example, start or stop, or suspend or resume processes).

- Operation on tasks (for example, assign tasks to users).

- Invoke Predictive Analytics models and use their output for further rule evaluation.

Rules can be expressed as decision tables or decision trees. 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. Result columns can also contain lists such as string or document lists. For every column, you can specify a REST service that provides a list of valid values, to make sure the cell content is valid from a business perspective.

■ A decision tree displays conditions and results in a tree-like structure that consists of nodes that are linked to each other. A node can represent the root, a condition (the IF part), or a result (the THEN part) of a rule. A link can be a root link or a condition link. A root node can be linked to one or more condition nodes, and a condition node can be linked to one or more condition nodes or result nodes.

You can group multiple decision tables 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*.

After you develop rules in Software AG Designer, you can drag and drop them onto the business process model design canvas.

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

Software AG offers a web 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. You export rules projects to My webMethods Server using the Business Rules RESTful API. Business administrators use the web browser-based user interface call the Rules Management Console 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. The Rules Management Console has been separated into several independent portlets, and can be customized using CSS stylesheets.

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 gadgets in Business Console. You can also develop your own user interfaces with technology such as OpenUI and Angular, 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.

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 59). 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. You can also use these services to schedule tasks.

- RESTful service API. You can use these services with the REST service capabilities of Integration Server. The services enable you to develop your own user interfaces using technology such as OpenUI and Angular. You can also use these services to schedule tasks.

- 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.

AgileApps

AgileApps is an application Platform-as-a-Service (aPaaS) product that enables you to visually build and deploy process-driven application solutions in weeks rather than months. AgileApps applications have enterprise capabilities such as a robust process modeler, flexible business rules, document assembly, drag-and-drop e-forms creation, customized reports, interactive dashboards, social collaboration, and mobile access. You can create database and case management pattern applications, including service request, incident management, and investigative. All applications are instantly mobile-enabled so users can access data everywhere. You can customize the look

and feel of the run time user interface of the AgileApps platform using Custom User Interface (CUI) templates.

In case management applications, users can collaborate to resolve cases and can define their own lightweight processes to guide the resolution. Case management applications can interact with other web-based applications that run on the AgileApps platform, and the web-based applications can use Integration Server services or user-defined web services to get data from the Internet. With SLA management, tasks are automatically created and monitored with escalation notifications. Integrated analytics, interactive dashboards, and reporting support business insights and improved decisions.

AgileApps is the entry point for process applications that are ad hoc, situational, and forms-based, where the primary designers are line of business and subject matter experts. You can deploy it in the cloud or in on-premises environments.

You can invoke an AgileApps case from a business process. Conversely, you can invoke a business process from an AgileApps case by mapping the case document to the process start document; the case status is updated with the process execution status.

AgileApps is available as an on-premises product or as a cloud-hosted solution.

Executing Business Processes

The product that executes business processes is Integration Server. Integration Server is a run-time server that has several functions in business process management. Run-time logic for business process models developed in Software AG Designer is created on Integration Server.

Integration Servers that execute business processes are equipped with a Process Engine that controls and directs process execution. You can design a business process to execute on a single Integration Server or you can distribute the process steps that make up the business process across multiple Integration Servers. In the distributed scenario, Universal Messaging is required to route process data from Integration Server to Integration Server to run the process steps.

Services such as flow, web, 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.

Software AG Designer publishes task applications to Integration Server, where they are invoked by processes at run time. You equip each Integration 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.

Note:

In earlier releases, Software AG Designer published task applications to My webMethods Server, where they were invoked by processes at run time. You equipped each My webMethods Server that ran tasks with a Task Engine that controlled and directed task execution. At run time, data and control passed from Process Engines to Task Engines and back again until the business process completed. My webMethods Server also hosted 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, Service Development, UI Development, and Rules Development. Each of these perspectives supplies the tools needed by a particular category of users.

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.

- Import supported electronic form (*e-form*) templates from a file system folder, a web server, or a My webMethods Server configured as a JSR-170 content 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 60, below).

- Use the Software AG Designer Rules Development perspective to develop rules.

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 to download the e-form data from the task in its original e-form format, disconnect from My webMethods Server and work with the e-form in the local environment, and then reconnect to My webMethods Server 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 as gadgets in Business Console. Alternatively, you can use non-Software AG technologies such as Grails or Google Widget Toolkit to design task user interfaces. In this scenario, 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 gadgets in Business Console. As the user completes each task in the workflow, the next gadget appears, eliminating the need for the user to locate each new task and open it manually.

When you are done developing a task step for your business process, you use Asset Build Environment and Deployer to build and deploy the task to an Integration 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.

Deploying Process Steps

You deploy process steps from Software AG Designer to Integration Server. Software AG Designer creates a package containing run-time execution information on Integration Server.

## Developing in a Collaborative Environment

Software AG offers tools and methods you can use to create a collaborative development environment in which developers can easily share and reuse assets. For example, 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 Professional 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.
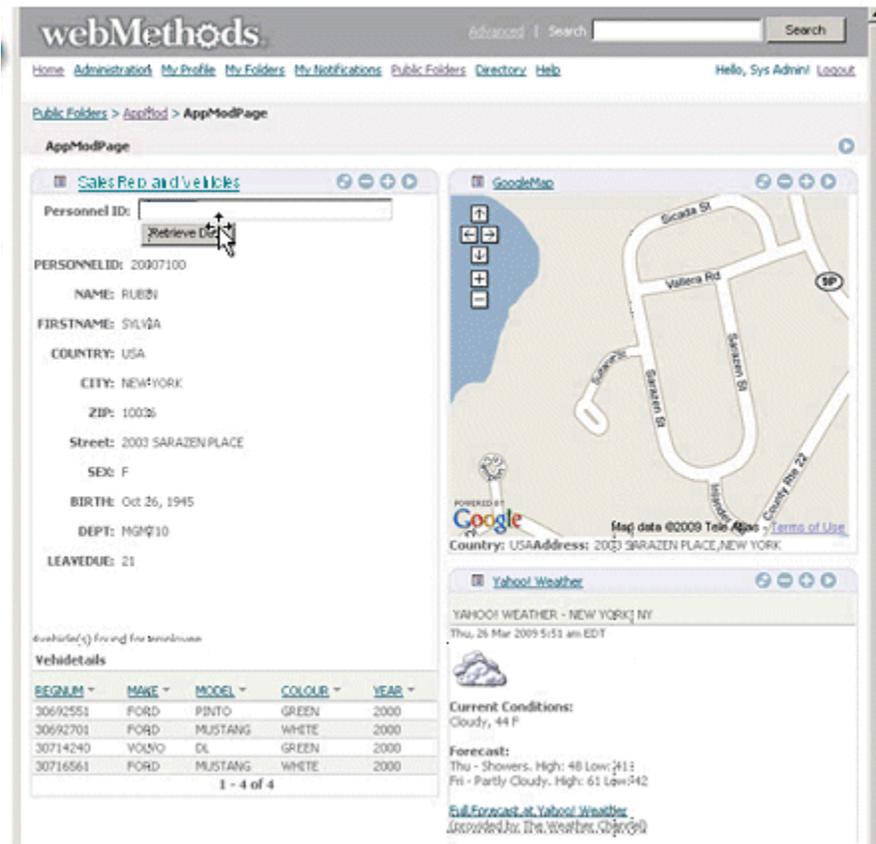
The primary product that developers use to develop composite applications is Software AG Designer. The products that execute composite applications are My webMethods Server and Integration Server.

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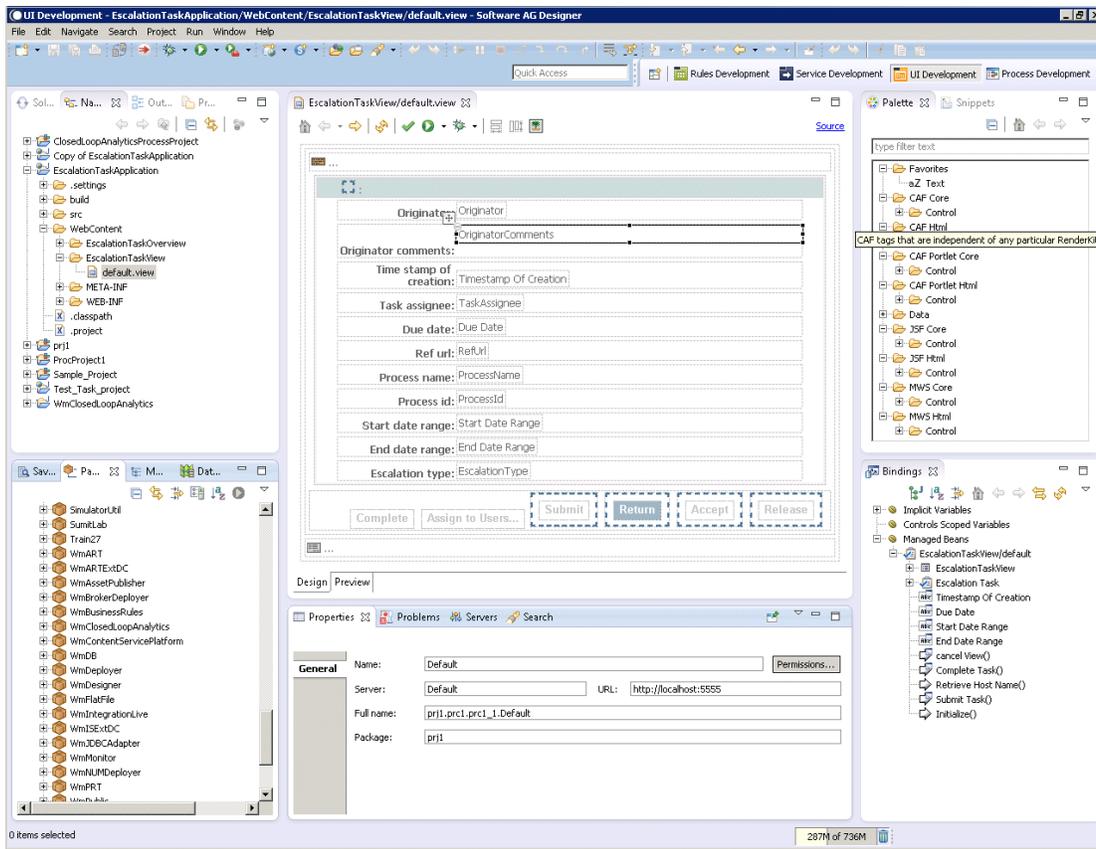
The webpages in your composite applications can invoke services such as Flow, web, adapter, and rule services. Within Software AG Designer, you can develop Flow, web, and rule services, and you can create adapter services that invoke programs on mainframes and UNIX systems. Software AG Designer builds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

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 create Business Console gadgets or 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 web browser-based user interfaces for many Software AG run-time products.

My webMethods Server is a run-time server with a web browser-based user interface that lets you customize the look and feel of webpages and control user access to them.

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.

Administering Business Processes

The primary products used to administer business process models and task applications are Monitor and Task Engine.

Monitor

Monitor is a web 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.

Task Engine

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

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. Process owners and participants can monitor process instances and task instances.

The primary products used to monitor process instances and task instances are Monitor, Task Engine, Optimize for Process, and Business Console.

Monitor

You can use Monitor to monitor business processes as follows:

- 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).

Task Engine

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.

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 web 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. You can also 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.

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 MashZone NextGen. Events generated by Optimize for Process include custom data associated with business processes.

Managing Business Processes and Tasks Using Role-Based Administration, Monitoring, and Social Guidance

The product you use to manage business processes and tasks is Business Console.

Business Console

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:

■ 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.

■ Schedule task instances using Business Console or the Task Engine built-in or RESTful services.

■ Customize and personalize workbenches using custom gadgets you create using JavaScript.

■ Monitor process and task instances in real time using a web browser.

■ View trending information relating to processes on social media, and view process analytics.

■ View decision tables that have been invoked by a process.

■ View processes that include a case and its related activity (also known as case tasks) and workflow status (also known as case process).

The following table lists Business Process dashboards that display summary metrics, instance detail, and other performance information about processes that has been collected by Optimize for Process in Business Console.

Dashboard	Displayed Information	Possible User Actions
Process Overview	High-level information about a selected process, including intrinsic metrics, historical views, and normality comparisons across a specified time range.	Send email notifications or escalation actions for individual stage instances.
Stage Instances	Stage activity and performance for a process across a specified time range. Metrics about each stage, including information about activity, cycle time, and alarms. Information about individual stage instances.	Identify potential issues, obtain information about alarms and stage activity, and use email or escalation actions to share findings with others.
Process Instance	Process activity and performance across a specified time range. Historical information about process volume activity and cycle time performance. Individual process instances.	Send email notifications or escalation actions for individual process instances.

Mobile Business Console is an application for mobile devices and tablets that enables business process owners and participants to collaborate on tasks and workflows, get real-time insights, and make time critical decisions. You can use Mobile Business Console to:

■ Work on tasks that are assigned to you, collaborate with other task assignees and process participants, and network with business process experts.

■ Add instant tasks, to-do lists, comments, and task attachments to process instances.

■ Monitor process and task instances in real time.

Mobile Development

Mobile development enables you to extend your integrations and business processes into the mobile space, so users anywhere in the world can interact with those integrations and business processes from their mobile devices.

The products you use for mobile development are Software AG Designer, Mobile Designer, and Mobile Support in the Mobile Suite. The Mobile Suite fully supports Mobile Enterprise Application Platform (MEAP) requirements and spans the full mobile life-cycle, 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.

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

Mobile Development offers native simulators for the mobile apps you develop.

Mobile Designer

You can develop mobile apps using Mobile Designer. 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 Designer is integrated with the Software AG Designer Service Development perspective so you can browse and add services from Integration Server to the application model.

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.

8 Master Data Management On Premises

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 a web browser-based user interface that lets you search for and 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.

The following table explains how OneData interacts with other Software AG products.

| Product	Description
Locate	Provides address reference data for more than 240 countries worldwide, verifies international addresses, checks address data for errors and omissions, and fills in missing address attributes when needed. Performs geocoding.
Integration Server	Executes services called by OneData (for example, to perform data validation or enrichment).
Universal Messaging	JMS provider for data acquisition and deployment in real time. If you modify data in OneData, OneData can publish a message to JMS topics or queues for subscribers to pick up. Conversely, OneData can subscribe to JMS topics and queues for messages about data changes in back-end systems such as CRM.

Product	Description
BigMemory Max	Maintains in-memory databases, which are cached versions of data objects. When you enable caching for a data object, OneData stores the records associated with the object in cache as well as in the release area. Caching improves the performance of REST services that use the data object, because the services interact with the cache instead of the database.
webMethods Business Process Management (BPM)	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.
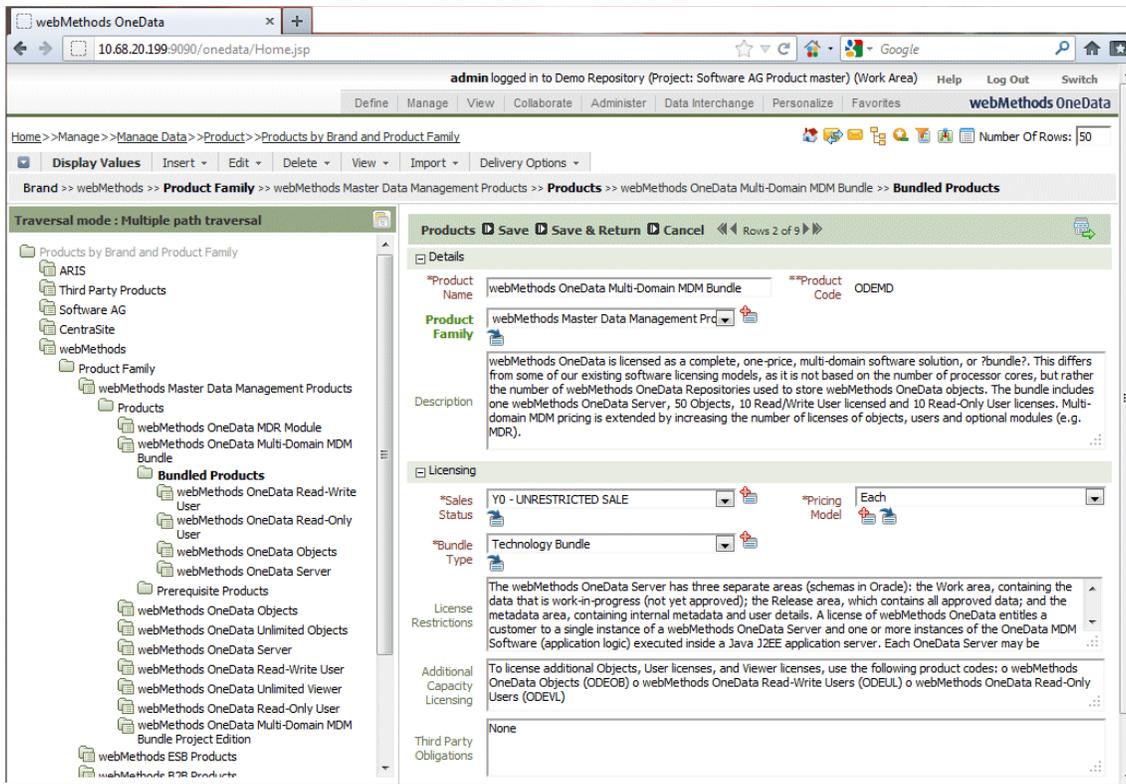
The illustration below shows a data grid whose contents reflect a data object created in OneData.

OneData and the OneData APIs are protected against cross-site script (XSS) attacks.

9 On-Premises Cross-Product Tools

The tools that support many products are Command Central, Software AG Installer, Software AG Update Manager, Deployer and Asset Build Environment, and Optimize for Infrastructure.

Software AG Installer

You use Software AG Installer to install Software AG products locally on standalone development machines. With Software AG Installer you can do the following:

- Install the latest releases of products and fixes directly from the Software AG Empower Product Support website on one machine.

- Download products and fixes from Empower into a product installation image and then install them from the image on one or more machines.

- Record a product installation script of yourself installing products and fixes from Empower or from a product installation image, and then install the products from the script with no user input on one or more machines.

- Build custom Docker images for certain products, such as the combination of Microservices Runtime, layered products such as API Gateway, and Universal Messaging. One use of such images is to aid in the move from on prem to private cloud.

Update Manager

You use Update Manager to install and uninstall fixes and support patches on Software AG products on one machine. Support patches include diagnostic collectors, test patches, and pre-QA fixes and are used to troubleshoot problems in Software AG product installations. With Update Manager you can do the following:

- Install the fixes and support patches directly from Software AG Empower Product Support website on one machine.

- Download fixes from Empower into a fix installation image and then install them from the image on one or more machines.

- Record a fix or support patch installation script of yourself installing fixes or support patches from Empower or from a fix or support patch installation image, and then install the fixes or support patches with no user input on one or more machines.

Command Central

You use Command Central in distributed staging and production environments to install, patch, configure, manage, and upgrade Software AG products; create database components; and connect products to database components. With Command Central, you can perform these tasks on multiple remote machines from one location.

With Command Central, you can work with existing standalone product installations or create new ones. If you are following DevOps practices, you can create software stacks of product runtimes.

Command Central offers a browser-based user interface. You can also automate tasks by using commands to remotely execute actions from a terminal or custom script. Command Central supports continuous integration servers such as Jenkins, and generic configuration management tools such as Chef and Puppet.

Deployer and Asset Build Environment

You use Deployer to deploy user-created assets that reside on source Software AG runtimes or repositories to target Software AG runtimes. For example, you might want to deploy assets you have developed on Integration Servers in a development environment (the source) to Integration Servers in a test or production environment (the target).

Deployer deploys assets built from sources in a development environment or VCS and stored on a repository. Asset Build Environment resides on the machines on which assets are created and supplies the scripts and libraries necessary to build assets into deployable units.

Digital Event Services

Software AG platforms are supported by an event-driven architecture (EDA). EDA is a software architecture pattern that supports the sharing of events across products - the production and detection of events, and the consumption of and reaction to events. The products that enable EDA are Digital Event Services and Universal Messaging.

Digital Event Services and Universal Messaging

Digital Event Services is the foundation for EDA in the Software AG environment; it enables Software AG products to communicate using events. With Digital Event Services, products can emit events in these ways:

■ Business processes can emit events to other business processes.

■ Parts of a product runtime can emit events to other parts.

■ Products can persist events to an event store.

Digital Event Services enables you to define the format of events to be emitted by one product and consumed by another using the native tools for each product. For example, you could emit documents based on an existing Integration Server document type as events for complex pattern detection by Apama without coding anything in either product.

Events can be visualized in dashboards.

Universal Messaging

Event producers publish messages to Digital Event Services in the form of events; Universal Messaging delivers them to consumers. Each digital event is associated with an *event type,* a schema or schema-based definition that describes the event's structure and is carried on a separate Universal Messaging channel.

Universal Messaging allows for horizontal scalability, in which you specify a set of servers or clusters to connect to within a single logical session and then can publish or subscribe to the same queue or topic across multiple servers or clusters.

10 Analytics & Decisions On Premises

Analytics & Decisions

Analytics & Decisions 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. Analytics & Decisions 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.

The following table lists example business cases for event processing.

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 Analytics & Decisions

The primary products you use in Analytics & Decisions are Apama, ARIS Process Performance Manager, MashZone NextGen, Universal Messaging and Zementis Predictive Analytics.

Analytics & Decisions Example

The following table explain activities a bank could perform to improve its process for car loan applications.

Category	Activity	Example
Detect patterns	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."
Visualize data	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.
Mine data	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.
Analyze root causes	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.
Improve process	Improve the business process model.	Add a line to the application for a guardian's signature.
Detect events	Detect and analyze complex patterns in real time.	Identify best action to take with unhappy customers based on recent customer support interactions.
Predict events	Classify data and make predictions.	Determine whether a credit card transaction is valid or fraudulent.

Visualizing Data

MashZone NextGen

MashZone NextGen is a data mashup and visualization tool used to deliver stored data to end users.

MashZone NextGen provides customizable dashboards that come with out-of-the-box components such as line charts, grids, and bubble charts. You can also create custom dashboards based on a flexible, responsive layout. MashZone NextGen also supports enterprises that need to deliver dashboards to users in multiple languages. You can have MashZone NextGen translate a dashboard's static content (dashboard title, tabs, widget labels, headings, and so on) into multiple languages at design time, and then MashZone NextGen will display the dashboard in the appropriate language based on the user's browser language, user settings, or a URL parameter. This feature substantially reduces the maintenance cost of multi-lingual MashZone NextGen deployments.

MashZone NextGen is entirely web browser-based to facilitate configuration of widgets. A custom widget framework enables the ability to create bespoke visualisations. MashZone NextGen also offers geo mapping widgets (that is, maps with markers and maps with vector overlay) you can use to do native mapping based on latitude and longitude or geographic areas defined by a GeoJSON file.

MashZone NextGen provides direct connectivity to consume Software AG product data (for example, ARIS PPM graphs and Apama and Digital Event Services events). MashZone NextGen can generate Apama events and send them to an Apama correlator for further processing.

MashZone NextGen also provides direct connectivity to consume data at rest (for example, JDBC, XML, CSV, JSON and Excel files, BigMemory Max and Terracotta). You can refine and condense data from BigMemory Max and Terracotta using Real-Time Analytics Query Language (RAQL) before visualizing it. You can store the results of complex and long-running data feeds into Terracotta for reuse within your dashboards, enabling the push-down of complex queries from MashZone NextGen to Terracotta. With Terracotta you can share the results of data feeds across multiple MashZone NextGen servers. Administrators can trigger materialization of feeds manually or externally via a REST API call.

Mining Process Data and Analyzing Root Causes

ARIS Process Performance Manager

ARIS Process Performance Manager, also known as ARIS PPM, is a tool for process analytics that draws data from various 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, ARIS PPM enables you to evaluate process-independent data using data analytics.

You specify the KPIs to investigate (for example, cycle times, frequencies, costs, adherence to deadlines, and quality guidelines). ARIS PPM automatically reconstructs the entire business process via process mining and visualizes the results in various process views. ARIS PPM provides a wide selection of chart types and tables in which to display the results of your analyses clearly and in detail. ARIS PPM data can be analyzed in different user interfaces, such as light-weight dashboards. The illustration below shows the comparison of the aggregated fast-running processes versus the aggregated slow-running processes described in .
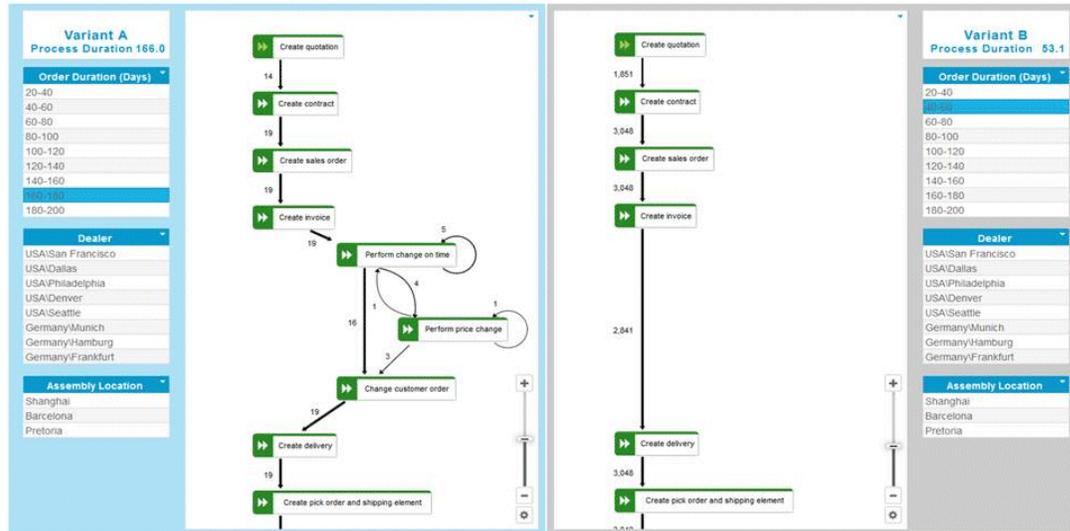
Detecting and Analyzing Complex Patterns in Real Time

Apama

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.

■ 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) and use Apama's Java and C++ plug-in interfaces to use logic written in these languages..

- 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 BigMemory Max's 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.

Apama's Data Player accelerates the development/deployment cycle of EPL applications 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 several ways to do this:

- Implement connectivity plug-ins. These are adapters that run inside the Apama correlator using a mixture of built-in and user-defined "transports" for message sending and receiving, and "codecs" for message transformation. Apama includes a built-in connectivity plug-in transports for use with protocols and systems such as MQTT, HTTP/S, Kafka, and Universal Messaging.

- Create applications that use Java Message Service (JMS) integrated into the Apama correlator.

- Develop client applications with Apama Java, .NET, or C++ APIs.

- 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 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.

You can use MashZone NextGen to create dashboards for Apama data views.

Making Predictions

Zementis Predictive Analytics

Zementis Predictive Analytics allows you to execute predictive models built by machine learning and data mining tools to make real-time predictions. The predictive algorithm classifies data into categories (for example, low water level or high water level) and performs regressions to predict the value that will come next. These classifications and predictions can be processed and sent as events to other Software AG products for the purpose of raising alerts, kicking off automated processes, and so on.

The open, standards-based approach of Zementis Predictive Analytics leverages Predictive Modeling Markup Language (PMML) to execute models created in all leading third-party or open source modeling platforms. The execution engine is highly performant, allowing new predictions to be made at the speed with which new data arrives.

Understanding Software AG Integration & API Products in the Cloud and On Premises