Microsoft BizTalk Server 2006 Part-V

Table of Contents

Module 7: Considerations for BAM Applications
    Performance Considerations for BAM Event Publishing
   BAM Code Maintenance Considerations

Module 8: Performance Considerations for BAM Event Publishing
    Synchronous Business Event Tracking
   Asynchronous Business Event Tracking

Module 9: BAM Code Maintenance Considerations
    Hard Code BAM Event Format
   One-Time Instrumentation

Module 10: BAM Dynamic Infrastructure
    What Is a BAM Definition Schema?
   Activity Data Storage
   Aggregations
   Querying BAM Data

Module 11: Aggregations
    Scheduled Aggregations
   Real-Time Aggregations

Module 12: Querying BAM Data
    Querying Instance Data
   Querying Real-Time Aggregated Data
   Querying Scheduled Aggregated Data
   Querying Activity Relationships

Module 13: Using Lookups to Enrich BAM Data
    Displaying Localized BAM Search Results
   Deploying Localized BAM XML Files

Module 14: Tracking Profile Editor
    What is the Tracking Profile Editor?
   Components of the Tracking Profile Editor
   Using the Tracking Profile Editor
   Tracking Profile Editor Keyboard Shortcuts
   Best Practices for Tracking Profiles
   Security Considerations for Tracking Profile Editor
   Walkthrough: Creating a Tracking Profile

Module 15: Components of the Tracking Profile Editor
    TPE Menu Options
   What Is an Activity View?
   What is the Source Event View?
   TPE Activity View Nodes

Module 16: What is the Source Event View?
    Orchestration Schedule View
   Messaging Payload View
   Context Property View
   Messaging Property View

Module 17: TPE Activity View Nodes
Activity and ActivityID Nodes
Continuation and ContinuationID Nodes
Relationship Nodes
Business Event Nodes
Data Item Nodes
Document Reference URL Nodes

Module 18: BAM End-to-End Walkthroughs
Walkthrough: Creating a BAM Activity with ODBA
Walkthrough: Creating a BAM Observation Model with the BAM Add-In for Excel
Walkthrough: Deploying the BAM Observation Model
Walkthrough: Creating a Tracking Profile
Walkthrough: Consuming BAM Data

Module 19: BAM Portal
Components of the BAM Portal
Planning for the BAM Portal
Customizing the BAM Portal Configuration
What Is an Activity Search?
What Is an Aggregation?
What Is an Alert?
Integrating BAM and HAT
Accessibility for the BAM Portal
Security Considerations for the BAM Portal
Walkthrough: Consuming BAM Data

Module 20: Components of the BAM Portal
BAM Portal Page Layout
Activity Search on the BAM Portal Page
Aggregations on the BAM Portal Page
Alert Manager on the BAM Portal Page

Module 21: What Is an Activity Search?
How to Create a Query in Activity Search
How to Set an Alert
How to View the Results of an Activity Search

Module 22: How to View the Results of an Activity Search
Activity Status
Related Documents
Related Activities
Request Assistance

Module 23: What Is an Aggregation?
How to Set an Alert
How to Modify an Aggregation

Module 24: Monitoring Business Activities with BAM
About Business Activity Monitoring
Business Activity Monitoring Workflow
Defining Business Activities and Views in Excel

Viewing Live BAM Data

Module 25: Defining Business Activities and Views in Excel
Defining a Business Activity
Defining a Business Activity View

Module 26: Defining a Business Activity View
Using the BAM Excel Template
Renaming Activity View Items
Defining Durations
Defining Milestone Groups
Defining Aggregations

Module 27: Defining Aggregations
Defining Measures
Defining Dimensions
Working with Preview Data
Defining Real-Time Aggregations

Module 28: Defining Dimensions
Progress Dimension
Data Dimension
Time Dimension
Numeric Range Dimension

Module 29: Viewing Live BAM Data
Using the BAS Collaboration Site (deleted 1/17/06)
Viewing BAM Aggregations
Searching and Navigating Activities

Module 30: Viewing BAM Aggregations
What are Real-time Aggregations
What are Scheduled Aggregations

Module 31: Searching and Navigating Activities
Viewing an Individual Activity
Viewing Related Activities and Related Documents

Module 32: Managing Business Activity Monitoring
BAM Scenarios for Administrators
Prerequisites for Using BAM Workbooks
Scheduling the DTS Packages
Archiving Primary Import Database Data
Creating a Partitioned View in the Archiving Database
How to Deploy Tracking Profiles with the BizTalk Tracking Profiles Management Utility
Managing the Tracking Data Decode Services (TDDS)
BAM Configuration Schema
Updating OLAP and Real Time Aggregation Connection String Properties
Business Activity Monitoring Security Recommendations
Managing the BAM Dynamic Infrastructure

Module 33: Managing the Tracking Data Decode Services (TDDS)
BAM Performance Counters

Module 34: Changing BAM Runtime Settings
BAM Configuration Schema
Updating OLAP and Real Time Aggregation Connection String Properties
Business Activity Monitoring Security Recommendations

Module 35: BAM Configuration Schema
BAM DTS Packages
Defining the Time Window and Time Slice Properties
Modifying the BAM Configuration Schema

Module 36: Managing the BAM Dynamic Infrastructure
Managing BAM Definitions
Managing BAM Security
Managing Aggregations
Managing BAM Databases

Module 37: Managing BAM Definitions
How to Deploy BAM Definitions
How to Undeploy BAM Definitions
How to List Changes to the BAM Infrastructure
How to List BAM Activities
How to Remove BAM Activities
How to List BAM Views
How to Remove BAM Views
How to Enable Alerts
How to Disable Alerts
How to Remove BAM Alerts
How to Update BAM Artifacts
How to Remove Deployed Artifacts

Module 38: Managing BAM Security
How to List Accounts in a View Role
How to Add Accounts to a View Role
How to Remove Accounts from a View Role
How to List Subscribers to an Alert
How to Add Subscribers to an Alert
How to Remove Subscribers from an Alert

Module 39: Managing Aggregations
How to Create an Index
How to Get the Indexes on an Aggregation
How to Delete an Index
How to Set the Duration on an Activity Window
How to Get the Duration on an Activity Window
How to Set the Duration on an RTA Window
How to Get the Duration on an RTA Window
Defining BAM Aggregations

Module 40: Managing BAM Databases

How to Set Up the BAM Databases Using the BAM Management Utility
How to Retrieve the BAM Configuration File Using the BAM Management Utility
How to Update the BAM Configuration Using the BAM Management Utility
How to Reference Additional BAM Primary Import Databases
How to Disable a BAM Primary Import Database Reference
How to List All Referenced Databases

Module 41: Human Workflow Services [BPM]

What Is Human Workflow Services (HWS)?
Managing Human Workflow Services
Developing Human Workflow Services Solutions

Module 42: What Is Human Workflow Services (HWS)?

WS Scenario: Health Risk Assessment
HWS Scenario: Proposal Review
HWS Workflow Model
HWS Architecture
HWS Components
HWS User Roles

Module 43: Managing Human Workflow Services

Using the HWS Administration Console
Enabling an Action for Use in Human Workflow Services
Adding Trusted Users
Editing Human Workflow Services Properties
Starting and Stopping the Human Workflow Services Web Service
Changing the Web Service User Context and Password
Registering and Unregistering Actions
Adding, Editing, and Deleting Constraints
Installing, Editing, and Removing Fact Retrievers
Working with Activity Flows
Configuring Human Workflow Services
Optimizing Human Workflow Services
Human Workflow Services Security Recommendations

Module 44: Using the HWS Administration Console

How to Open the HWS Administrative Console

Module 45: Developing Human Workflow Services Solutions

Creating an HWS Project
About Actions
Creating Actions
About Action Templates
About Action Constraints
About Activity Models
About Fact Retrievers
About Action Template Schemas
Creating Action Template Schemas

Module 46: Creating Actions
Modifying the Activation Message
Modifying the Task Message
Adding Action Logic and Tasks
Creating and Binding HWS Ports
Deploying and Registering Actions
Creating Dependent Actions
Correlating Actions with Tasks

Module 47: About Action Templates
Action Template Layout
Action Template Correlation Types
Action Template Ports
Action Template Exception Handling

Module 48: About Action Constraints
Multiple Constraints
Positive and Negative Constraints
Activity Model Constraints
Constraints Interaction with Fact Retrievers
Constraints Data Types

Module 49: About Activity Models
Developing Activity Models
Dependent and Independent Transitions
Metadata
Saving and Retrieving Activity Models
Step Constraints
Role Constraints
Loops
Default Parameters

Module 50: About Action Template Schemas
Hws_Activate Schema
Hws_ActivateResponse Schema
Hws_Synchronize Schema
Hws_Task Schema
Hws_Finish Schema
Hws_Interrupt Schema
Tasks for Defining and Updating Schemas

Module 51: Tasks for Defining and Updating Schemas
Defining/Changing the Schema Target Namespace
Updating the Action Name
Updating the Action Assembly Name
Defining a Human Target for an Action

Module 52: Defining Business Processes with ODBA
What Is a Business Process Orchestration Diagram?
How to Install and Start ODBA
Creating a Business Process Orchestration Diagram
How to Add Non-ODBA Shapes and Connectors

How to Validate a Business Process Orchestration Diagram
How to Export a Business Process Orchestration Diagram to BizTalk Server
 How to Use ODBA to View a BizTalk Orchestration
 Creating a BAM Activity Definition with ODBA

Module 53: Creating a Business Process Orchestration Diagram
 How to Add a Begin shape (ODBA)
How to Add an Action Shape (ODBA)
How to Add a Fork Shape (ODBA)
How to Add a Join Shape (ODBA)
How to Add a Decide Shape (ODBA)
How to Add a Group Shape (ODBA)
How to Add a Loop Shape (ODBA)
How to Add an End Shape (ODBA)
How to Add a Connector (ODBA)
How to Fix ODBA Shape Layout
How to Save a Business Process Orchestration Diagram

Business Process Management

This section contains information about using the BizTalk Server 2006 business process management tools.

In This Section

- Business Activity Services

- Using Business Activity Monitoring

- BAM Management Utility

- Human Workflow Services [BPM]

- Defining Business Processes with ODBA

Business Activity Services

This section contains information about using the Business Activity Service Feature.

In This Section

- Developing Business Activity Services Solutions

- Managing Business Activity Services

- Managing Partner Relationships with BAS

Developing Business Activity Services Solutions

Business Activity Services (BAS) provides a single point from which you can easily manage all trading partners and run their associated business processes. Before you read this section, you should be familiar with the following information:

- Business Activity Services (BAS)

- Managing Business Activity Services

This section contains:

- About Trading Partner Management Roles

- About Profiles

- About Profile Templates

- About Partner Schemas

- Extending Partner Schemas

- About Agreements

- Business Document Schemas and Templates Creation

- Creating BAS-Enabled Orchestrations

- **Business Activity Services Solutions Packaging and Deployment**

About Trading Partner Management Roles

The process of developing and using BAS trading partner management solutions may involve several people within your organization. This section contains a summary of the tasks each role completes for Business Activity Services.

Developer role

A Developer performs the following tasks.

Task	Description
Creates XML schemas	Uses the BizTalk Editor to create XML schemas for any document that you must exchange with a trading partner, such as a purchase order or an invoice.
Creates BAS-enabled orchestrations	Uses the Orchestration Designer to create orchestrations that implement the business processes within a business relationship.
Creates user forms	Uses Microsoft® Office InfoPath™ to create form templates used by the business user. These templates are based on the XML schema files (.xsd) created by the BizTalk Editor.
Configures the tracking information	Uses the Tracking Profile Editor to map the BAM views in the BAS-enabled orchestrations. This is an optional step.

Administrator roleAn Administrator performs the following tasks.

Business user role

A Business User performs the following tasks.

Task	Description
Manage profiles and documents	Use Microsoft Windows SharePoint Services to create partner profiles and documents.
Communicate with	Use Microsoft Windows SharePoint Services to execute the

partners	business processes created by the developer through the Inbox and Outbox.

Business manager role

A Business Manager performs the following tasks.

Task	Description
Manage Agreements	Use Microsoft Windows SharePoint Services to create, edit, activate/deactivate, and browse agreements.
Manage Profiles	Use Microsoft Windows SharePoint Services to create, edit, classify, deploy, un-deploy, and browse partner profiles.
Manage Groups	Use Microsoft Windows SharePoint Services to create and delete groups.
Create Partners	Use Microsoft Windows SharePoint Services to create partner profiles with advanced properties.

Business analyst role

A Business Analyst performs the following task.

Task	Description
Configure Business Activity Monitoring (BAM) views	Use the BAM configuration wizard to define the data you want to track and display in near real-time. This is an optional step.

About Profiles

Profiles contain information about a business. There are two types of profiles, a self profile and a partner profile. You can create these profiles manually using the Microsoft® Office InfoPath™ provided templates, or you can receive them from a partner electronically in a BizTalk application.

Self profiles

Self profiles contain information about your own business. A self profile acts as a business card for your organization, which can be sent to partners to start business interactions. Partner profiles represent the business cards of partners.

Partner profiles

A partner profile represents a trading partner and contains information about them. Partner profiles contain both business data and technical data:

- **Business data** - includes business information such as contact names and addresses.

- **Technical data** - the basic information needed to start conducting e-business with the partner, such as a URL at which the partner receives electronic messages.

BAS uses this information to create the required ports and configurations to interact with partners electronically. In addition to partner profiles, Business Activity Services allows creation of profiles to represent internal entities. This is especially useful when different internal departments interact with different partners.

Though the schema is identical for self and partner profiles, there are separate self and partner profile InfoPath templates. This is because a self profile can be packaged and sent to a partner in a BizTalk application, where it acts as the partner profile. While they both use the same schema, they expose different parts of the schema through the InfoPath user interface.

The business data part of Profile Schema is extensible, having both required and optional fields. You can remove all of the optional fields and you can add any number of additional fields as needed.

About Profile Templates

Profile templates provide an easy way to create and update profiles. Business Activity Services provides three different templates for self and partner profiles:

- Self profile template

- Partner Profile template (for business users)

- Partner profile template (for administrators)

When navigating the Microsoft® Windows® SharePoint™ Services site, users will only see the templates that are applicable to their role. For example, a user in the business manager or business administrator role sees an advanced view of the templates that allows modifications to the global ports, but a business user does not see the advanced view.

You access partners and partner groups using the Microsoft Office InfoPath™ templates provided with Business Activity Services. InfoPath provides the default templates. To secure the partner templates, you use Microsoft Windows SharePoint Services roles and security features.

About Partner Schemas

Partner schemas define partner profiles and provide an easy way to customize profiles to meet the individual needs of any organization. Customizing a partner profile involves making changes to the partner schema file (**partner.xsd**) associated with that profile. You can modify the following elements of the partner schema file:

- **PartnerObject**

- **Address**

- **Contacts**

Although you can modify these elements, you should not change the element fields listed in the following table.

Parent element	Field name
PartnerObject	- PartnerId - Description (Optional) - LastModified - ActiveStatus - PartnerIdCrossReference - Properties - TechnicalInfo
Address	- AddressId
Contacts	- ContactId

You can use the Profiling System available in Microsoft Commerce Server 2002 to modify these schemas. For more information about the Commerce Server 2002 Profiling System, go to the MSDN Web site(Commerce Server 2002 Profiling System). The Profiling System Microsoft Management Console (MMC) allows an administrator to modify the **PartnerObject**, **Address**, or **Contact** schema elements. You can remove any non-required fields, and you can add additional fields as needed.

After you modify the Profiling System schema for a profile, you also make corresponding updates to the partner .xsd file. You must deploy the updated partner schema file in the Trading Partner Management (TPM) Web service folder.

After you update both the Profiling System and partner .xsd files, you must update the Microsoft InfoPath template for your partner to reflect these changes. You must deploy the new InfoPath template in the templates folder under the Business Activity Services Windows SharePoint Services Web site.

Extending Partner Schemas

You can extend the BAS Partner schema. This process involves modifying:

- SQL tables

- User Profile Management (UPM) profile schema files

- Partner template files on Partner, and the Address and Contact objects.

To extend the partner schema template

1.   First, extend the Partner, AddressObject and ContactObject tables in the TPM database to include new properties. For more information about changing SQL database tables, go to the Microsoft TechNet Web site (SQL Server 2000 Books Online).

2.   In the Commerce Server 2002 Profiling System Microsoft Management Console (MMC), add new data members in Data Objects (Partner, AddressObject and ContactObject). For more information about adding new data members, go to the MSDN Web site (Running the Profiles Resource).

3.   In the Commerce Server 2002 Profiling System Microsoft Management Console (MMC), add new properties in Profile Definitions (PartnerObject, AddressObject and ContactObject) to map new data members.

4.   Add the new properties to the Partner.xsd and PartnerNoWsdl.xsd files in the *<drive>*:\Program Files\Microsoft BizTalk Server 2006\Business Activity Services\TPM\Schemas folder.

5.   In Microsoft InfoPath window, click **File**, and click **Open** and navigate to and double click *<drive>*:\Program Files\Common Files\Microsoft Shared\Web Server Extensions\60\TEMPLATE\LAYOUTS\InfoPath\PartnerProfileAdmin.xsn to open the partner template.

6.   In the Microsoft InfoPath window, click **File** and click **Extract Form Files** while in design mode. This extracts all of the form files from the form template file and saves them to a specified folder on your hard disk.

7.   Add new properties into schema.xsd file (extracted from the .xsn file). For more information about how to add new properties to a schema, go to the MSDN Web site (A Quick Guide to XML Schema).

8.   Add new properties into sampledata.xml and template.xml (extracted from the .xsn file).

9.   In the Microsoft InfoPath window, click **File**, and click **Open**. Navigate to and double click the manifest.xsf file (extracted from the .xsn file) to open it in design mode and add new properties.

10.   In the Microsoft InfoPath window, click **File** and click **Publish** to publish the updated manifest.xsf file and use it overwrite the PartnerProfileAdmin.xsn file, as an .xsn file.

11.   Upload PartnerProfileAdmin.xsn from BAS site.

12.   Add new properties into *<drive>*:\Program Files\Common Files\Microsoft Shared\Web Server Extensions\60\template\layouts\1033\BAS\Xml\PartnerObject.xml.

Important   You may need to repeat steps 4-9 for PartnerProfileStandard.xsn and PartnerProfileSelf.xsn.

About Agreements

You use agreements to configure business relationships. An agreement is defined between your self profile and your partners profile, and it consists of one or more addendums. Addendums define the business process used, the role of your organization, your partners role, and the parameters used in the relationship.

Agreements can also refer to partner group profiles instead of individual partner profiles. This facilitates management of partners that have identical trading relationships with your organization. When you create an agreement for a group, all of its members automatically inherit that agreement.

Each addendum defines the configuration of a business process. You implement the business processes as orchestrations with specific public service link types and roles used to associate the appropriate profiles. You can also write them to provide the appropriate mechanisms to accept parameters and enforce business parameters.

Creating agreements involves configuring business processes by associating roles with the appropriate profiles and setting up the appropriate business parameters. This enables business users to participate in business processes, because business rule changes no longer require modifications to the lower-level business process code, which generally requires a developer. This lets business users change the business parameters without needing to recompile the orchestrations and involve developers or consultants in the process.

Agreement Schemas

Business Activity Services provides an agreement schema that allows business users and managers to define partner relationships. You can create the partner agreement from scratch using the provided schemas or receive them from partners electronically. For information about receiving BAS artifacts (such as profiles) from partners, see Managing BizTalk Applications and Managing BAS Artifacts.

Every agreement specifies the two profiles used in the business relationship. One is the self profile and the other is the partner or partner group profile. If you specify a partner group, BizTalk creates an agreement for each partner within that group. An agreement consists of one or more addendums. Each addendum is based on a particular business process, and specifies what roles will be played by their

organization (self), and the partner's role. It also contains information about the parameters used in this particular relationship.

Business Document Schemas and Templates Creation

You import the XML schemas used in the orchestrations into the Microsoft® Office InfoPath™ designer to create a solution template for business users. These schemas define the documents exchanged between partners in a business transaction, such as a purchase order or an invoice.

Partner profiles and agreements used by Business Activity Services list as documents within the Microsoft Windows® SharePoint™ Services site. When the user needs to edit these documents, they open them in InfoPath as a solution template.

BAS provides a feature to map Inbox XML documents with a deployed InfoPath solution template. By doing this, business users see InfoPath forms instead of raw XML when they open the documents located in the Inbox.

To develop InfoPath solution templates for your business documents, import the XML schemas used in the orchestrations. These schemas define the documents to exchange between partners in a business transaction, such as a purchase order or an invoice.

After you develop an InfoPath solution for a specific business document schema, you upload the template to the partners document library. If you want your solution templates to be global for all partners, upload the template to your BAS Windows SharePoint Services sites Templates document library.

When you upload the documents to your partners document library or the global Templates document library, you need to specify Namespace for business document schema, which is target namespace of the XSD.

When the orchestration saves a message to the Inbox, BAS looks up the partners document library and the global Templates document library to find a matching InfoPath solution template and inserts an InfoPath specific processing instruction to the Inbox messages if it finds one.

Creating BAS-Enabled Orchestrations

You implement automated business processes through BizTalk Server 2006 orchestrations. Developers must create Business Activity Services-enabled orchestrations that provide business users with the basic underlying structures for creating electronic business agreements. The Microsoft® Windows® SharePoint™ Services user interface exposes these basic structures as part of creating an agreement.

In Business Activity Services, you exchange documents between partners using a model similar to an e-mail server, but built on a Windows SharePoint Services server. Incoming documents are delivered to an Inbox, users save outgoing documents in an Outbox for sending, and a copy of the sent document is

automatically placed in the Sent Items folder if it is successfully processed. By creating an orchestration that adheres to the Business Activity Services requirements, you are enabling your business processes to run within the Business Activity Services environment and exchange messages using the Inboxes and Outboxes on your Microsoft Windows SharePoint Services collaboration site.

On a Windows SharePoint Services extended virtual server, by default, Windows SharePoint Services handles the entire URL namespace. To create a virtual directory for your own use, for example, you can create a virtual directory as a receive location for your Microsoft BizTalk® Server schedule. To do this, you must exclude the URL for this virtual directory from Microsoft Windows SharePoint Services managed paths. The same is true to create a Web form or Web service application on the Microsoft Windows SharePoint Services extended virtual server.

See the "Managing Paths" section in Microsoft Windows SharePoint Services Administrator's Guide for details on how to manage paths from the command line.

Refer to the Microsoft SharePoint Products and Technologies v.2.0 Software Development Kit for more information on creating web applications on a Microsoft Windows SharePoint Services extended virtual server.

This section contains:

- Defining Business Relationships

- Business Activity Services Messaging Infrastructure

- Parameterized Orchestrations

- Preventing Duplicate Inbox Messages

Defining Business Relationships

You define business relationships in the Orchestration Designer using role links that contain two roles. Each of these roles represents a partner in the business relationship the orchestration is implementing. For example, you might implement an automated purchasing relationship in a purchasing role link that contains both a buyer and a seller role. You use the role link to implement two different orchestrations for both sides of the transaction. For example, the orchestration representing the buyer process implements the buyer role and uses the seller role. Conversely, the orchestration representing the seller process orchestration implements the seller role and uses the buyer role. After you create these roles, an administrator can easily package the partner orchestration and associated information in a BizTalk application for quick deployment on a partner running BizTalk Server 2006.

To start building Business Activity Service-enabled orchestrations, you must first decide which business relationship and the associated roles that you want to model. For example, to build business processes for an order management relationship that consists of two roles, including a buyer and a seller. Ensure that you implement each

side of the relationship in a separate orchestration using a different BizTalk Server Project. In the order management relationship, separate projects are required for the seller and the buyer. For the purposes of this example, you use SellerProcess and BuyerProcess. Typically, you only need to implement one side of the relationship, either SellerProcess or BuyerProcess, based upon the role that you play in the relationship. However, for BizTalk application scenarios both implementations are required, as you need to provide implementation for your partners as well as yourself.

Implementing the Order Management business processes

The following steps show how you implement the order management business processes:

Seller

1. Create a new BizTalk Server 2006 project named **SellerProcess**.

2. Add XML schemas to the project for the business messages that will be interchanged. For example, a purchase order, confirmation, invoice, etc. For more information, see Creating Schemas for XML Messages.

3. Add a new orchestration file (.odx) to the project, and name it **Definitions**. This file will contain all the common orchestration types, which will be used by both the SellerProcess and BuyerProcess implementations.

4. In the **Definitions** orchestration:

    -   Add port types for **SellerPort** and **BuyerPort**, using the orchestration type window. Under **SellerPort**, specify operations for messages that the seller will be receiving. For example, an operation named **PO** that uses the **PO** XML schema. Under the **BuyerPort**, specify operations for messages that the buyer will be receiving, for example confirmation and invoice messages.

    -   Add a public role link type called **OrderManagement**. Add two roles called **Buyer** and **Seller** under this role link type. Add **BuyerPort** under the **Buyer** role and **SellerPort** under the **Seller** role.

5. Add a new orchestration file (.odx) to the project named **SellerSvc**. This orchestration implements the seller side of the **OrderManagement** process.

6. In the **SellerSvc** orchestration create a new role link called **ordermanagement** of the existing type created in step 4, called **SellerSvc.OrderManagement**. Specify that you will be implementing the **Seller** role and using the **Buyer** role.

7. Create the custom implementation of the business process as needed.

8. To use the Microsoft Windows SharePoint Services Inbox and Outbox, follow the requirements outlined in Business Activity Services Messaging Infrastructure.

9.   If parameters are required, parameterize your orchestration as described in Parameterized Orchestrations.

10.   Build the orchestration project in Microsoft Visual Studio®.

Buyer

1.   Create a new BizTalk Server 2006 project named **BuyerProcess**.

2.   Add the XML schemas to the project that you created in the seller process. It is important that schema namespaces of the common XML messages are identical in both projects. Use the **Add Existing Item** option on the Visual Studio Project menu for this purpose.

3.   Add the **Definitions** orchestration file created in the seller process to this project.

4.   Add a new orchestration (.odx) file to the project called **BuyerSvc**. This orchestration will implement the buyer side of the order management process.

5.   In the **BuyerSvc** orchestration, create a new role link called **ordermanagement** of existing type created in the Definitions.odx file, called **SellerSvc.OrderManagement**. Specify that you will be using the **Seller** role and implementing the **Buyer** role.

6.   Create the custom implementation of the business process as needed.

7.   To use the Microsoft Windows SharePoint Services Inbox and Outbox, follow the requirements outlined in Business Activity Services Messaging Infrastructure.

8.   If parameters are required, parameterize your orchestration as described in Parameterized Orchestrations.

9.   Build the orchestration project in Microsoft Visual Studio.

Business Activity Services Messaging Infrastructure

When creating a Business Activity Services-enabled orchestration to define a process, you must use role links to identify the relationships of the organizations using this process. You use role links in BizTalk Server 2006 Orchestration Designer to represent the relationship and roles of organizations within a specific business transaction, such as a supplier, customer, or shipper. This mechanism provides a layer of abstraction in the implementation that allows configuration modifications without needing to recompile and redeploy your orchestration source files.

Business Activity Services Orchestration Requirements

Business Activity Services-enabled orchestrations must use predefined names for the role link types, roles, and port types that interact with the Windows SharePoint Services collaboration site and the trading partner publishing Web service (TPPubWS). This section identifies these requirements.

- The orchestration must contain two role links for communicating with Microsoft Windows SharePoint Services using these predefined role link types:

  - **receiveBusinessDocumentsLT**

  - **sendBusinessDocumentsLT**

  Note You define role link types as ProjectName.RoleLinkType. Therefore, in a project named MyProject, the receive role link type would be defined as MyProject.receiveBusinessDocumentsLT.

- Within the **sendBusinessDocumentsLT** role link, you must define:

  - A role named **sender**.

  - Two send port types named **sendToInboxPT** and **sendToSentItemsPT** within the sender role.

- Within the **receiveBusinessDocumentsLT** role link, you must define:

  - A role named **receiver**.

  - One receive port type named **receiveFromOutboxPT** within the receiver role.

- For each logical port, you must add one port operation per message type. For example, if the orchestration sends a purchase order and a payment to your partner, the **sendToInboxPT** port would contain two send operations. You define one send operation for the purchase order document and one for the payment.

- Implicitly, a Business Activity Services orchestration must have at least one role that you use, not just implemented by your organization, which indicates you are communicating with a partner.

Depending on the orchestration you develop, you may have the received document as the first operation, rather than the sent document. If this is the case, you must set the receive shapes **Activate** property to **True**.

Parameterized Orchestrations

When creating Business Activity Services-enabled orchestrations, you may need to design these in a flexible way that allows for variances between different partners that use the roles you defined. For example, suppose you have created a role link representing a shipper in your orchestration. You may need to pass specific information to the shipper in your orchestration instance, such as an account number or a discount code. You accomplish this by creating parameterized schedules that you can reuse for the same role.

You must know these points of variance between orchestration instances when you create the orchestration. This allows you to expose the parameter to the business

user who sets the value in the Microsoft Office InfoPath template. At run-time, the parameter value should be unique to the context of the partner associated with that orchestration instance. Using the previous example, the account number used with one shipper should be unique when compared with the account number of another shipper. This also implies that every instance of the orchestration associated with a specific partner will have the same parameter value.

Parameterized orchestration requirements

You can write Business Activity Services-enabled orchestrations to accept parameters. This section identifies these requirements.

- The orchestration must contain a role link with a type ending with the string **PolicyType**. The unqualified name of this type, excluding the **PolicyType** string, is exposed as the parameter name in the agreement. For example, a role link type you define as MyRole.AccountNumberPolicyType would expose a parameter named **AccountNumber** on the agreement.

- The role link must contain a single role called **PolicyConsumer**, which is a role being used, not implemented. In addition, this role should be initiated by sending data, not receiving data.

- The **PolicyConsumer** role must contain a single port with an unqualified port type value of TPPubWS. This port should contain two operations, one for sending the request and one for receiving the parameter value.

  Before you can assign this port type, you must add a Web reference to the trading partner publishing Web service (tppubws.wsdl) to the orchestration. You do not need to create the "TpPubWs" port. You create this port and add it to the list of port types automatically simply by adding the Web Reference.

Preventing Duplicate Inbox Messages

To guarantee that a particular message appears in the Inbox only once, set the filename property for the message before you send it to the Inbox. If the BizTalk host happens to send this message more than once to the Inbox, in an error condition in which the host retries the send for example, then the new message replaces the existing message with the same filename in the Inbox. If you do not create this filename and instead rely on a unique message ID that BizTalk creates, then the same message could appear more than once in the Inbox (but with different filenames).

Setting the filename property

To set the filename property for messages sent to the Inbox, set the Microsoft.BizTalk.KwTpm.StsDefaultPipelines.OutboxFileName context property to a unique filename, for example a filename derived from a unique ID contained in the message.

To access the Microsoft.BizTalk.KwTpm.StsDefaultPipelines.OutboxFileName property, add a reference to the Microsoft.BizTalk.KwTpm.StsDefaultPipelines.dll assembly. You can find the assembly in the BAS Runtime Components folder of the BizTalk Server installation directory.

Managing Business Activity Services

Business Activity Services (BAS) provides information workers with a simplified way to interact with trading partners and business processes. The BAS interface enables information workers to interact with business processes in a familiar Microsoft Office environment.

The BAS environment consists of a Web site hosted in Microsoft Windows SharePoint Services and InfoPath templates. Windows SharePoint Services and InfoPath provide a common user interface for all of the services included in BAS.

BAS works with Microsoft Office to provide collaborative functionality to business users. BAS enables information workers to be proactively involved with business processes by providing a shared workspace that correlates business events and information and provides the tools for working with them.

Administering BAS includes:

- Installing and configuring the Windows SharePoint Services collaboration site and Microsoft BizTalk Server 2006

- Associating a BizTalk host and a configuration database with BAS by creating a BizTalk registration

- Running synchronization or repair tasks to synchronize data between the BAS database and the BizTalk Management or Windows SharePoint Services content databases

This section contains the information an administrator needs to effectively manage BAS.

In This Section

- Business Activity Services Security Considerations

- How to Manage BAS Rights and Access Control

- Business Activity Services Role Membership

- Changing BAS Service Account Passwords

- Defining Receive Locations

- Removing the Business Activity Services Site

- Removing Permissions on Temporary ASP.NET Files on Uninstall

- Enabling Trace in Business Activity Services

- Improving Business Activity Services Performance

Business Activity Services Security Considerations

Business Activity Services (BAS) is a layer on top of Microsoft BizTalk Server 2006 that provides additional services and exposes a limited set of management functions to business users. Business-sensitive data travels from the client computers used by information workers to the BAS Web services server. BAS supports using Secure Sockets Layer (SSL) to encrypt the data.

BAS uses roles to define the security infrastructure. All BAS users must be assigned to a BAS role. For information about the BAS roles, see <u>Business Activity Services Role Membership</u> .

BAS provides the following three Web services that work with the BAS Microsoft Windows SharePoint Services site and the Microsoft InfoPath templates:

- **TpPubWS**. Business users, business managers, and business administrators use this Web service to create, edit, and delete artifacts in the BAS Trading Partner Management (TPM) database.

- **TpMgmtWS**. Business managers and business administrators use this Web service to create artifacts in the BizTalk Management database and create, edit, and delete artifacts in the BAS TPM database.

- **StsWebService**. Only internal BAS components, including the TpPubWS and TpMgmtWS Web services, can access this Web service to update the BAS Windows SharePoint Services site.

For more information about BizTalk Server Web services, see <u>Using Web Services</u> .

The following table describes the four groups BizTalk Server 2006 Setup creates in the BAS Windows SharePoint Services site.

Business Activity Services site groups	Description
BizTalk BAS User	Business users perform publishing operations that do not invoke BizTalk Server. They have the fewest privileges.
BizTalk BAS Managers	Business managers perform management operations that make changes in BizTalk Server, such as deploying partners and activating agreements.

BizTalk BAS Administrators	Business administrators perform operations that could severely impact BizTalk Server, such as creating a BizTalk registration and synchronizing and repairing databases with the Trading Partner Management database
BAS Web Service Accounts	The BAS Web Services group is an internal group used by the BAS components. No interactive users should be added to this group.

BizTalk Server 2006 setup creates Windows NT group accounts and adds them as members to these Windows SharePoint Services groups to provide end users access to the BAS Windows SharePoint Services site. For information about the Windows NT group accounts BAS uses,

BizTalk Server 2006 setup creates application accounts that have the correct permissions over the site. Setup removes the default Windows SharePoint Services groups—Reader, Contributor, and Web Designer—for the BAS site. For information about the Reader, Contributor, and Web Designer groups, see Windows SharePoint Services   Administrator's   Guide,   located   at http://go.microsoft.com/fwlink/?LinkId=26132.

For information about the BAS service roles, see Business Activity Services Role Membership .

Configuring BAS automatically adds the Windows NT group accounts BAS uses to the corresponding group on the Business Activity Services SharePoint Services site. You add a user to a BAS Windows group, to grant the user access to the Business Activity Services collaboration site with the related permissions of the corresponding BAS site group. You do not need to manually add a particular user to a site group unless more site access rights are needed for that user.

Business Activity Services Role Membership

You map each local group or domain group that you create to a role in Business Activity Services (BAS). BAS has the following four roles in addition to existing Microsoft BizTalk Server roles:

- **BizTalk BAS Users.** Users in the business user group role can access the BAS site and have permission to use some aspects of the trading partner publishing Web service. For example, these users can create partner profiles, but they cannot deploy the profiles to a BizTalk Server computer.

- **BizTalk BAS Managers.** Users in the business manager group role can access the BAS site and have permission to use Trading Partner Management (TPM) tools. They can perform tasks in BAS that configure business processes in BizTalk Server such as deploying partners or activating agreements.

- **BizTalk BAS Administrators.** Users in the business administrator group role can access the BAS site and perform administrative tasks on the site, such as registering BizTalk Server computers.

The Microsoft Windows SharePoint Services administrator must be a member of the specified business administrator group.

- **BizTalk BAS Web Services.** The technical group role contains the different service accounts used to run BAS; it does not contain individual user accounts. Service accounts have access to the TPM database and to the Business Activity Service Windows SharePoint Services Web service.

If your technical group is a domain group, you must add two service accounts to the technical group: the BizTalk BAS Publishing Web Service account and the BizTalk BAS Management Web Service account. Both accounts must have read/write access to the TPM database. If your technical group is a local group, BizTalk adds the two service accounts to the group automatically during configuration.

The trading partner publishing Web service runs under the BizTalk BAS Publishing Web Service account. You must add the BizTalk BAS Management Web Service account to the BizTalk Administrators group.

Changing BAS Service Account Passwords

You may need to change the passwords for the Business Activity Services (BAS) service accounts. You can change the passwords for the following service accounts in the configuration framework:

- BAS Publishing Web service

- BAS Management Web service

- BAS Web Services application pool

To update the password for the BAS Publishing Web service account

1.   Update the password for the BizTalk Sharepoint Messaging Adapter Windows NT service. Open the Services console. Update the Log On property to set the new password.

2.   Update the credentials for the Publishing Web service. The credentials are stored encrypted in the registry. Do the following:

   - Use aspnet_setreg.exe to update the **Software\Microsoft\BizTalk Server\3.0\TPPubWS\identity** key.

   - Set proper access control lists (ACLs) on the **ASPNET_SETREG** registry key under the **identity** key.

   - Grant read permissions to the service account running your TpmWSAppPool application pool.

3.   Update the credentials for the StsWebReceive default.aspx page. The credentials are stored encrypted in the registry. Do the following:

- Use aspnet_setreg.exe to update the **Software\Microsoft\BizTalk Server\3.0\STSWebReceive\identity** key.

- Set proper ACLs on the **ASPNET_SETREG** registry key under the **identity** key.

- Grant read permissions to the service account running your TpmWSAppPool application pool.

To update the password for the BAS Management Web service account

1.   Update the credentials for the Management Web service. The credentials are stored encrypted in the registry. Do the following:

- Use aspnet_setreg.exe to update the **Software\Microsoft\BizTalk Server\3.0\TPMgmtWS\identity** key.

- Set proper ACLs on the **ASPNET_SETREG** registry key under the **identity** key.

- Grant read permissions to the service account running your TpmWSAppPool application pool.

To update the password for the BAS Web Services application pool account

1.   Update the password for the STSWebReceiveAppPool application pool. Do the following:

- Launch INETMGR.

- Find the **STSWebReceiveAppPool** application pool.

- Under **Properties**, click the **Identity** tab.

- Set the new password for this service account and then click **OK**.

2.   Update the password for the TpmWSAppPool application pool. Do the following:

- Launch INETMGR.

- Find the **TpmWSAppPool** application pool.

- Under **Properties**, click the **Identity** tab.

- Set the new password for this service account and then click **OK**.

Defining Receive Locations

By default, on a Windows SharePoint Services-extended virtual server, Windows SharePoint Services handles the entire URL namespace.

To create a virtual directory for your own use, for example, you can create a virtual directory as a receive location for your BizTalk schedule. To do this, you must exclude the URL for this virtual directory from Windows SharePoint Services managed paths. The same is true to create a Web form or Web service application on a virtual server running Windows SharePoint Services.

For information about how to manage paths from the command line, see the "Managing Paths" section in Microsoft Windows SharePoint Services Administrator's Guide, located at http://go.microsoft.com/fwlink/?LinkId=26169.

For information about creating Web applications on a virtual server running Windows SharePoint Services, see the Microsoft SharePoint Products and Technologies 2003 Software Development Kit (SDK), located at http://go.microsoft.com/fwlink/?LinkId=26174.

Removing the Business Activity Services Site

You may want to remove the Business Activity Services (BAS) site. For example, if you are uninstalling Microsoft BizTalk Server 2006, you might also want to remove all related components.

If you configured BizTalk Server 2006 for BAS, and you uninstall BizTalk Server 2006 without removing the BAS site, you may have problems performing partner-related operations.

To remove the Business Activity Services site

1.   Click **Start**, double-click **Control Panel**, double-click **Administrative Tools**, and double-click **SharePoint Central Administration**. The SharePoint Services administration site opens.

2.   On the **Central Administration** page, in the **Virtual Server Configuration** section, click **Delete site collection**.

3.   On the **Delete Site Collection** page, enter the complete URL of the site you want to delete. For example, if you used the default URL during the BizTalk Server configuration for BAS, you would type, **http://<*servername*>/sites/BASSite**, and then click **OK**.

4.   On the **Delete Site Collection** page, click **Delete**.

Removing Permissions on Temporary ASP.NET Files on Uninstall

While configuring Business Activity Services (BAS) the following folders are granted permissions to \WINDOWS\Microsoft.NET\Framework\v1.1.4322\Temporary ASP.NET Files\. When you uninstall Microsoft BizTalk Server, the uninstall process does not automatically remove these files. After you uninstall BizTalk Server and the BAS site, you must manually remove the following permissions:

- BizTalk BAS Web Services group

- BizTalk BAS Management Web Service account

- BizTalk BAS Publishing Web Service account

- BizTalk Server BAS Application Pool account

Enabling Trace in Business Activity Services

To assist you in troubleshooting issues in your Business Activity Services (BAS) deployment, you can enable trace in the BAS components.

The following table describes the BAS components in which you can enable trace.

BAS component	Description
TpPubWS	The trading partner publishing Web service
TpMgmtWS	The BAS trading partner management Web service
StsWebService	The BAS Windows SharePoint Services Web service
StsHandlers	The BAS Windows SharePoint Services Web application
StsWebParts	The BAS Windows SharePoint Services custom Web parts
StsBizTalkAdapter	The BAS BizTalk Messaging adapter
OfficeImporters	The BAS messaging transformation component
StsWebServicePerf	The BAS Biztalk Messaging adapter performance counters
StsAdapterEventHandler	The BAS BizTalk Windows SharePoint Service event handler
StsWebReceive	The BAS BizTalk message receiving Web application

Tracing output

Tracing output for all BAS components appears as follows:

The following table describes the elements of the tracing format.

Tracing element	Description
<BizTalk2004>	All trace statements from our components use "<BizTalk2004>" as the first item in the line.
[TAB]	A tab character. All items are tab separated to make it easier to write a tool that automatically parses the log files.
DATETIME	The date and time that the log entry was written.
ThreadHashID	Integer that represents the hash code of the thread ID.
ClassName	The name of the class from which the trace statement was called.
MethodName	The method name from which the trace statement was called.
LogType	One of the following: Enter, Exit, Info, Warning, or Error. Note that these strings are not localized.
Enter	Indicates that this is a log of a method entry point. A tab-separated list of parameters follows.
Exit	Indicates that this is a log of the method exit. The return value (if any) follows. • Info – Indicates a general informational trace entry. An informational message follows. • Warning – Indicates a general warning trace entry. A warning message follows. • Error – Indicates an error trace entry. An error message follows.
ParametersOrMessage	Either a tab-separated list of parameters or a message depending on the log type as described above.

Enabling Tracing for TpPubWS and TpMgmtWS

To enable tracing for the Publishing and Management Web services, use a text editor such as Notepad to edit the TpPubWS and TpMgmtWS Web.config files.

By default, the Management Web service Web.config file is located at %PROGRAMFILES%\Microsoft BizTalk Server 2006\Business Activity Services\TPM\Management\Web.config.

By default, the Publishing Web service Web.config file is located at %PROGRAMFILES%\Microsoft BizTalk Server 2006\Business Activity Services\TPM\Publishing\Web.config.

To enable tracing for the TpPubWS and TpMgmtWS Web.config files

1. Set the value for the **Microsoft.BizTalk.KwTpm.TpmWebService** switch to **1**. The value is set to 0 (disabled), by default.

2. Add a file listener. The listener code is included in the Web.config files, but is commented out. To add a file listener, remove the comment code around the **trace** element and update the file path with the path of the location where you want the trace file to be written.

Enabling Tracing for StsWebService

To enable tracing for the StsWebService, use a text editor such as Notepad to edit the StsWebService Web.config file.

By default, the StsWebService Web.config file is located at %PROGRAMFILES%\Microsoft BizTalk Server 2006\Business Activity Services\KWSTS\KWStsWebService\Web.config.

To enable tracing for the StsWebService

1. Change the value of the **Microsoft.BizTalk.KwTpm.StsWebService** switch to **1**. The value is set to 0 (disabled), by default.

2. Add a file listener. Copy the listener code from the TpPubWS Web.config file, paste it into the STSWebService Web.config file, and update the file path with the path of the location where you want the trace file to be written.

Enabling Tracing for StsHandlers

To enable tracing for the StsHandlers, use a text editor such as Notepad to edit the StsHandlers Web.config file.

By default, the StsHandlers Web.config file is located at %PROGRAMFILES%\Common Files\Microsoft Shared\Web Server Extensions\60\TEMPLATE\LAYOUTS\1033\BAS\Web.config.

To enable tracing for the StsHanlders

1. Change the value of the **Microsoft.BizTalk.KwTpm.StsHandlers** switch to **1**. The value is set to 0 (disabled), by default.

2.   Copy the listener code from the TpPubWS web.config file, paste it into the StsHandlers web.config file, and update the file path with the path of the location where you want the trace file to be written.

Enabling Tracing for StsWebParts

To enable tracing for the StsWebParts, use a text editor such as Notepad to edit the root Web.config file for the Web server that Windows SharePoint Services runs on—for example, c:\inetpub\wwwroot\web.Config on the default Web site.

To enable tracing for the StsWebParts

1.   Change the value of the **Microsoft.BizTalk.KwTpm.StsWebParts** switch to **1**. The value is set to 0 (disabled), by default.

    The following code shows the area of the Web.config files that you change:

2.   Add **EventLogListener System.Diagnostics.EventLogTraceListener** to the Web.config file.

    Use the following code to add EventLogListener
    System.Diagnostics.EventLogTraceListener to the Web.config file:

Alternatively, you can implement a custom listener by inheriting from the **System.Diagnostics.TraceListener** class. For information, see the .NET Framework Class Library TraceListener Class located at http://go.microsoft.com/fwlink/?LinkId=26172.

Enabling Tracing for OfficeImporters, StsBizTalkAdapter, and StsWebServicePerf

To enable tracing for OfficeImporters, StsBizTalkAdapter, and StsWebServicePerf, use a text editor such as Notepad to edit the OfficeImporters, StsBizTalkAdapter, and StsWebServicePerf Web.config files.

By default, the Web.config file is located at %PROGRAMFILES%\Microsoft BizTalk Server 2006\Business Activity Services\KWSTS\Shared\application.config.

To enable tracing for OfficeImporters, StsBizTalkAdapter, and StsWebServicePerf

1.   Change the values of the Microsoft.BizTalk.KwTpm.OfficeImporters, Microsoft.BizTalk.KwTpm.StsBizTalkAdapter, and Microsoft.BizTalk.KwTpm.tsWebServicePerf switches to 1. The values are set to 0 (disabled), by default.

    The switch names have the same name as the respective assemblies.

2.   Add a file listener. The listener code is included in the Web.config files, but is commented out. To add a file listener, remove the comment code around the **trace** element and update the file path with the path of the location where you want the trace file to be written.

Enabling tracing for StsAdapterEventHandler

To enable tracing for the StsAdapterEventHandler, use a text editor such as Notepad to edit the root Web.config file for the Web server that Windows Sharepoint Service runs on—for example, c:\inetpub\wwwroot\web.Config on the default Web site.

To enable tracing for for StsAdapterEventHandler

1.   Add a switch named **Microsoft.BizTalk.KwTpm.StsAdapterEventHandler** (assembly name) to the root Web.config file for the Web server that Windows SharePoint Services is running on. For example, it could be c:\inetpub\wwwroot\web.config.

2.   Add a file listener. Add **EventLogListener System.Diagnostics.EventLogTraceListener** to the Web.config file.

Enabling Tracing for StsWebReceive

To enable tracing for the StsWebReceive, use a text editor such as Notepad to edit the root Web.config file for the Web server that Windows Sharepoint Service runs on—for example, c:\inetpub\wwwroot\web.Config on the default Web site.

To enable tracing for StsWebReceive

1.   Change the value of the **Microsoft.BizTalk.KwTpm.StsWebReceive** switch to 1. The value is set to 0 (disabled), by default.

2.   Add a file listener. The listener code is included in the Web.config file, but it is commented out. To add a file listener, remove the comment code around the trace element and update the file path with the path of the location where you want the trace file to be written. By default, the Web.config file is located at %PROGRAMFILES%\Microsoft BizTalk Server 2006\Business Activity Services\KWSTS\Shared\application.config.

Business Activity Services Security Recommendations

With Business Activity Services (BAS), business analysts can create relationships with trading partners, define the role of a partner, create business agreements, and other aspects of the relationship with the partner. For more information, see Managing Business Activity Services. It is recommended you follow these guidelines for securing and deploying BAS in your environment.

•   You must install standard Windows Message Queuing on the BAS computer whenever BAS and the BizTalk Server runtime (processing server) are on separate computers.

- If the BAS computer and the BizTalk runtime computer (processing computer) are in different domains:

  - You must disable standard Windows Message Queuing message signing.

  - There must be a BizTalk Message Queuing receive adapter in the processing domain to pick up messages from the BAS computer in the corporate domain.

- BizTalk Server does not support BizTalk Message Queuing and standard Windows Message Queuing on Network Address Translation (NAT) firewalls. Therefore, for the BAS runtime computer, the firewall acts as a router. The firewall works as usual for all other computers on the corporate domain.

- The BAS Web services need to be able to authenticate users from the corporate domain. Therefore, it is recommended that you place the BAS computer in the corporate domain, thus ensuring the security of the other BizTalk computers in the services, processing, and data domains.

- The service account running the BAS Management Web service needs to be a member of the BizTalk administrators group.

- All accounts running services on the BAS computer must be accounts from the data domain, as these accounts need access to resources in the processing, data and services domains. However, all the accounts for the BAS user groups should be accounts from the intranet domain.

For additional security considerations, see <u>Business Activity Services Security Considerations</u>.

Managing Partner Relationships with BAS

Business users use the BizTalk Server 2006 Business Activity Services (BAS) site to manage information about trading partners and the way in which BizTalk Server exchanges messages with those partners.

The BAS site is installed when you configure BizTalk Server with BAS, or you can install BAS on a separate computer with network connectivity to your BizTalk Server installation.

You use the BizTalk Server Business Activity Services (BAS) Web site to manage trading partner profiles and agreements:

- **Trading partners** are any entity with which your organization interacts or transacts business.

- **Trading partner profiles** store business and technical data about partners.

- **Agreements** are documents that describe how your organization conducts business with a trading partner.

The following table provides a workflow for using the BAS site to establish and maintain partner relationships.

Task	Topic
A systems administrator grants business users access to the BAS site by adding the business users to the appropriate Windows user group.	Managing Access to the BAS Site
A BAS administrator creates a BizTalk Server registration on the BAS site to connect the site to the BizTalk Server Management database (by default, BizTalkMgmtDb).	Connecting the BAS Site to BizTalk Server
A business user creates a profile for a trading partner and a profile for the home organization. The profiles contain business data for each organization. A BAS business manager adds credential and communication data to the trading partner profile. A business manager deploys the trading partner profile.	Creating Profiles for Trading Partners [BTS2006]
A business user creates profile groups to organize partner profiles.	Organizing Partner Profiles with Partner Groups
A business user creates a partner agreement. A BAS business manager activates the agreement to begin the trading partner relationship.	Establishing Partner Relationships with Agreements
A business user can access messages received from the trading partner and send messages to the trading partner.	Exchanging Messages with Partners and Processes
A BAS administrator makes sure that the data in the BAS site and in the Trading Partner Management database is the same.	Managing Partner Data
Business users manage partner profiles on an ongoing basis.	Modifying Partner Profiles
Business users manage agreements on an ongoing basis.	Modifying Trading Partner Agreements

In This Section

- Managing Access to the BAS Site

- Connecting the BAS Site to BizTalk Server

- Creating Profiles for Trading Partners [BTS2006]

- Organizing Partner Profiles with Partner Groups

- Establishing Partner Relationships with Agreements

- Exchanging Messages with Partners and Processes

- Managing Partner Data

- Modifying Partner Profiles

- Modifying Trading Partner Agreements

- BAS Keyboard Shortcuts

Managing Access to the BAS Site

The Business Activity Services (BAS) site provides tools for business users to easily manage trading partner information and establish relationships with trading partners to exchange business documents.

There are two layers of security in BAS. The first layer is the site access control provided by the Microsoft Windows SharePoint Services roles. BAS uses Windows SharePoint Services to control access to the site for browsing.

You can also create custom roles for your organization to grant or deny users of other configurable sites the rights provided by Windows SharePoint Services. Such rights include Lists, Web part pages, and sub-Web management. BAS creates four Windows SharePoint Services roles during installation

The second layer of security management relates to the business tasks access control performed by BAS components. The BAS site uses roles to define site security. All business users who access BAS must be assigned to a BAS role. The BAS roles control access to the different areas of the BAS Web site.

Each role is represented by a Windows user group. A systems administrator controls membership in the BAS Windows user groups.

The BAS roles are:

- **BAS business user**. Business users are granted minimum privileges on the BAS site.

- **BAS business manager**. Business managers are granted medium privileges on the BAS site.

- **BAS administrator**. BAS administrators are granted high privileges on the BAS site.

Members of the BAS business user, BAS business manager, and BAS administrator are information workers.

Because the BAS site surfaces tasks such as writing trading partner data to BizTalk databases and establishing message exchange relationships with internal and external partners, we recommend that you take care when assigning users to the BAS roles.

In addition to the BAS user roles, an additional role, BAS Web Services, exists for non-interactive user accounts under which BAS Web services run. This Windows group is called the BizTalk BAS Web Services Group.

BAS Site User Groups

A systems administrator uses membership in the BAS user groups to manage access to the BAS site. Each BAS user group is associated with a BAS role and a set of privileges on the BAS site. The systems administrator can customize role membership and privileges of users by adding and removing user group members.

Systems administrators can view the detailed permissions of the BAS user roles on the group management page in Windows SharePoint Services. For information about managing user access for Microsoft Windows SharePoint Services, see "Managing Site Groups and Permissions" and "Managing Users and Cross-Site Groups" in the security section of the Windows SharePoint Services Administrator's Guide at http://go.microsoft.com/fwlink/?LinkId=26132.

There are four BAS user roles and one system role. Each user role has a set of access rights, and has permission to perform a set of designated tasks.

The business activity permissions are checked against the user groups in the following table:

Windows SharePoint Services role	Windows NT group membership	Site rights and responsibilities
Business User	BizTalk BAS Users	• Manage profiles: creating, editing, and deleting • Manage documents: creating, sending, and browsing • Manage agreements: creating, editing, and browsing • Manage groups: creating and deleting partner groups and adding and removing partner profile membership in partner groups. • Site rights: Can view pages • List rights: Can add, edit, delete, and view

		items
Business Manager	BizTalk BAS Managers	• Manage agreements: creating, editing, activating, deactivating, and browsing • Manage profiles: creating, editing, classifying, deploying, undeploying, and browsing • Manage groups: creating and deleting • Manage documents: creating, sending, and browsing • Site rights: Can view page • List rights: Can add, edit, delete, and view items

Business Administrator	BizTalk BAS Administrators	• Installation and setup of:
		
		   • Windows SharePoint Services
		
		   • Business Activity Services site
		
		• Site rights: Can view, add, and customize pages
		
		• List rights: Can add, edit, delete, and view items
		
		• Register and unregister the site with the BizTalk Management database (BizTalkMgmtDb).
		
		• Synchronize the TPM database with Windows SharePoint Services
		
		• Repair discrepancies between the TPM database and the BizTalk Management database

BAS Web Services	BizTalk BAS Web Services Group	• For noninteractive user accounts under which BAS Web services run.

Connecting the BAS Site to BizTalk Server

BAS administrator registers the BAS site with an installation of BizTalk Server. Registering the BAS site with BizTalk Server connects the BAS site to the BizTalk management database and the Trading Partner Management database.

In This Section

- How to Connect the BAS Site to BizTalk Server

- How to View a BizTalk Server Registration Summary

- How to Modify a BizTalk Server Registration

- How to Remove a BizTalk Server Registration from BAS

How to Connect the BAS Site to BizTalk Server

business administrator creates a Microsoft BizTalk Server registration on the Business Activity Services (BAS) site to associate the BizTalk host and the BizTalk Management database with the BAS site. You can only register one BizTalk Server in BAS. For failover support, you can cluster your BizTalk host computers in a Network Load Balancing cluster and create the BizTalk registration in BAS by using the virtual machine name for the cluster.

Prerequisites

Note the following requirement:

- You must be a member of BizTalk BAS Administrators group to register BizTalk Server with BAS.

To create a BizTalk Server registration

1. In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site, and then press ENTER.

2. On the **Business Activity Services Site Home** page, click **BizTalk Servers**.

3. On the **BizTalk Servers** page, click **Register BizTalk Server**.

4. On the **BizTalk Server Registration** page, do the following:

Use this	To do this
Registration Name	Type the name of the BizTalk registration.
BizTalk Management Database Server Name	Type the name of the SQL Server computer that hosts the BizTalk Management database (BizTalkMgmtDb).
BizTalk Management Database	Type the name of the BizTalk Management database that contains configuration information for the BizTalk host computer.
Refresh Host Lists	Click to refresh the **Outbox Receive Location Host** list and the **Parameter Service Host** list.

Outbox Receive Location Host	This field is automatically populated when you click **Refresh Host Lists**. Or, you can use the drop-down list to select a host to use for the **Outbox Receive Location Host**.
Parameter Service Host	This field is automatically populated when you click **Refresh Host Lists**. Or, you can use the drop-down list to select a host to use for the **Parameter Services Host**.

5.   Click **Save and Close**.

How to View a BizTalk Server Registration Summary

You can view the details of a Microsoft BizTalk Server registration. The summary page lists information such as the name of the server and the database on which the server is running.

Prerequisites

Note the following requirement:

- You must be a member of BizTalk BAS Administrators group to view a BizTalk Server registration.

To view a BizTalk Server registration

1.   In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site, and then press ENTER.

2.   On the **Business Activity Services Site Home** page, click **BizTalk Servers**.

3.   On the **BizTalk Servers** page from the **<BizTalk Server registration>** drop-down list, select **Summary View**.

The **BizTalk Servers: <BizTalk Server registration name>** page lists the BizTalk registration properties and the deployed partners.

How to Modify a BizTalk Server Registration

You can modify the Outbox Receive Location and Parameter Services hosts in a Microsoft BizTalk Server registration.

Prerequisites

Note the following requirement:

- You must be a member of BizTalk BAS Administrators group to modify a BizTalk Server registration.

To modify a BizTalk Server registration

1.   In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site, and then press ENTER.

2.   On the **Business Activity Services Site Home** page, click **BizTalkServers**.

3.   On the **BizTalk Servers** page, from the <BizTalk Server registration name> drop-down list, select **Edit Item**.

4.   On the **BizTalk Servers: <BizTalk Server registration name>** page, do the following:

Use this	To do this
Outbox Receive Location Host	From the drop-down list, select a host.
Parameter Service Host	From the drop-down list, select a host.

5.   Click **Save and Close**.

How to Remove a BizTalk Server Registration from BAS

You can unregister a Microsoft BizTalk Server. By unregistering a server you delete it from the Business Activity Services (BAS) site. You can unregister a BizTalk Server only if it has no deployed partners or active agreements and if the corresponding BizTalk receive location is not bound to any orchestration.

Prerequisites

Note the following requirement:

•   You must be a member of BizTalk BAS Administrators group to remove a BizTalk Server registration.

To unregister a BizTalk Server

1.   In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site, and then press ENTER.

2.   On the **Business Activity Services Site Home** page, click **BizTalk Servers**.

3.   On the **BizTalk Servers** page, from the **<BizTalk Server registration>** drop-down list, select **Unregister BizTalkServer**.

4.   In the message box, click **OK**.

Creating Profiles for Trading Partners [BTS2006]

Ramkumar JD   IntSol

Business users create trading partner profiles to store information about trading partners:

- **Trading partners**. Trading partners are any entity with which your organization interacts. For example, a trading partner could be an external company or a department within your company.

- **Trading partner profiles**. You use trading partner profiles to store both business data and technical data for trading partners:

  - Examples of business data that business users can store in a partner profile include the partner's name, address, and contact information. You can store multiple addresses and contacts for each partner.

  - Examples of technical data that business managers can store in partner profiles include security and communication details that enable the exchange of messages with the partner.

  For information about the BAS roles, see Managing Access to the BAS Site .

  You can create multiple profiles for each trading partners. For example, if you buy products from a company, you could create one profile for the department that receives your purchase orders, and a separate profile for the department that receives your payments.

- **Self profiles**. In addition to creating profiles for your trading partners, you must also create profiles for your organization that you share with trading partners. Profiles that you create for your organization are called self profiles. By creating multiple profiles for your organization, you can customize the information you share with each of your trading partners. In the BAS site, self profiles are referred to as "My Profiles."

The following table provides a workflow for creating trading partner profiles using the BAS site.

Task	Topic
A business user creates the initial profile.	How to Create a Profile
A business user uses a custom property to add additional information to the profile.	How to Add a Custom Property to a Profile
A business user adds one or more addresses to the profile.	How to Add an Address to a Profile
A business user adds one or more contacts to the profile.	How to Add a Contact to a Profile
A business manager adds signature certificate information to the trading partner profile.	How to Add Partner Credential Checking to a Profile

A business manager deploys the trading partner profile.	How to Deploy a Profile

In This Section

- How to Create a Profile

- How to Add a Custom Property to a Profile

- How to Add an Address to a Profile

- How to Add a Contact to a Profile

- How to Add Partner Credential Checking to a Profile

- How to Add a Send Port to a Profile

- How to Deploy a Profile

How to Create a Profile

Business users use the **Partner Profiles** page on the BAS site to create a partner profile containing information about a trading partner. You can create multiple partner profiles for each partner. For example, you can create one partner profile for a partner's headquarters, and a second partner profile for a partner's subsidiary.

You use the **My Profiles** page on the BAS site to create a My profile containing information about your company that you want to share with partners. You can create multiple My profiles. For example, you can create one My profile to share with partners who buy products from you, and one for partners from whom you buy products.

For information about trading partners, trading partner profiles, and My profiles, see Creating Profiles for Trading Partners [BTS2006].

Prerequisites

Note the following requirements:

- You must use the 32-bit version of Internet Explorer to create trading partner profiles. If you installed the 64-bit version of Windows on the computer you use to create profiles, you must install the 32-bit version of Internet Explorer.

- A BAS administrator must connect the BAS site to BizTalk Server so that the profile information you enter in this step is stored in the Trading Partner Management database. For information about connecting the BAS site to BizTalk Server, see Connecting the BAS Site to BizTalk Server .

- You must logon as a member of one of the following Windows user groups:

  - BizTalk BAS Users

  - BizTalk BAS Managers

  - BizTalk BAS Administrators

To create a profile

1. In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site, and then press ENTER.

2. On the **Business Activity Services Site Home** page, click one of the following:

Use this	To do this
My Profiles	Click to create a self profile and then on the **My Profiles** page, click **New Profile**.
Partner Profiles	Click to create a partner profile, and then on the **Partner Profiles** page, click **New Partner**.

3. In the Profile template, on the **General Information** tab, do the following:

Use this	To do this
Name	Type a unique name to identify the profile.
Description	Type a description for the profile.
Insert Alternate Names	Type any alternate names for the profile.
Duns Number	Type the Dun & Bradstreet DUNS number for the profile. The DUNS number is a unique nine-digit identification sequence that provides a unique identifier for a single business entity and links corporate family structures together.
Internal ID	Type the ID you use to refer to the profile internally in your organization.
Currency	From the drop-down list, select the preferred currency for this profile.
Payment Method	Type the payment method for the profile. For example, check or credit card.

Tax Exempt	Select **Yes** or **No** to indicate whether the company is tax exempt or not.
Tax Id	Type in a Tax ID for this profile.
Term Id	Type in a Term ID for this profile. For example, your organization may have a standard set of terms such as "Net 30" or "Cash on delivery (COD)."

4.   Click **Submit Profile** to save the profile.

Next Steps

After you submit the profile, you can add one or more of the following to the profile:

- **Custom properties**. For instructions, see How to Add a Custom Property to a Profile.

- **Addresses**. For instructions, see How to Add an Address to a Profile.

- **Contacts**. For instructions, see How to Add a Contact to a Profile.

- **Signature certificates**. For instructions, see How to Add Partner Credential Checking to a Profile.

- **Send ports**. For instructions, see How to Add a Send Port to a Profile .

How to Add a Custom Property to a Profile

You can add a custom property to a profile. You use custom properties to add additional information to the profile.

You can choose a property from the list that BAS provides, or you can create a property that is not on the list. For each property, BAS uses a unique identifier. You assign a description and a value for the property.

BAS provides the name and unique identifier (property qualifier) for the list of custom properties in the **Select Property** dialog box. Or, you can create your own.

Prerequisites

Note the following requirements:

- You must use the 32-bit version of Internet Explorer to create trading partner profiles. If you installed the 64-bit version of Windows on the computer you use to create profiles, you must install the 32-bit version of Internet Explorer.

- You must log on as a member of one of the following Business Activity Services (BAS) user groups:

  - BAS business users

  - BAS business managers

  - BAS business administrators

To add a custom property to a profile

1. In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site, and then press ENTER.

2. On the **Business Activity Services Site Home** page, click one of the following:

Use this	To do this
My Profiles	Click to add a custom property to a self profile.
Partner Profiles	Click to add a custom property to a partner profile.

3. On the **My Profiles** page or the **Partner Profiles** page, click the profile to which you want to add a custom property.

4. In the Profile template, on the **General Information** tab, click **Insert custom property**.

    In the **Custom properties** section, you can either select a predefined property from a list, or create a new property.

To select a predefined property, do the following

1. In the **Custom properties** section, click the **Property qualifier** ellipse **[...]**.

2. In the **Select Property** dialog box, select the property that you want to add to the profile, and then click **OK**.

    In the profile, the property name appears in the **Property name** box, and the property qualifier appears in the **Property qualifier** box.

3. In the **Custom properties** section, do the following:

Use this	To do this
Property description	Type a description for the property.
Property value	Type the value for the property.

4. Click **Submit profile** to save your changes to the profile.

To create a new property, do the following

1. In the **Custom properties** section, do the following:

Use this	To do this
Property name	Type a unique name for the property.
Property qualifier	Type a unique identifier for the property.
Property description	Type a description for the property.
Property value	Type the value for the property.

2. Click **Submit profile** to save your changes to the profile.

How to Add an Address to a Profile

You can add one or more addresses to both self profiles and partner profiles. You use the profile template **Address** tab to add addresses to profiles. For example, you may add both a billing and a shipping address to a profile. For each address, you can add one or more of the following:

- Telephone number

- Facsimile (Fax) number

- E-mail address

- Web address

Prerequisites

Note the following requirements:

- You must use the 32-bit version of Internet Explorer to create trading partner profiles. If you installed the 64-bit version of Windows on the computer you use to create profiles, you must install the 32-bit version of Internet Explorer.

- You must log on as a member of one of the following Business Activity Services (BAS) user groups:

  - BAS business users

  - BAS business managers

- BAS business administrators

To add an address to a profile

1.   In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site, and then press ENTER.

2.   On the **Business Activity Services Site Home** page, click one of the following:

Use this	To do this
My Profiles	Click to add an address to a self profile.
Partner Profiles	Click to add an address to a partner profile.

3.   On the **My Profiles** page or the **Partner Profiles** page, click the profile to which you want to add an address.

4.   In the Profile template, on the **Address** tab, click **Insert Address**, and then do the following:

Use this	To do this
Address type	Type the address type. For example, headquarters, warehouse, shipping, or billing.
Address	Type the street address.
City	Type the city name.
State/Province	Type the state or province name.
ZIP/Postal Code	Type the ZIP or postal code.
Insert country	Click to type a country/region name. The address, for example, may be for a foreign subsidiary.
Insert telephone	Click to type a telephone number and description (for example, main or personal). You can add multiple telephone numbers to an address.
Insert Fax	Click to type a facsimile number and description. You can add multiple fax numbers to an address.
Insert e-mail	Click to type an e-mail address and description. You can add multiple e-mail addresses to an address.
Insert Web address	Click to type a company URL and description. You can add multiple Web addresses to an address.

5.   To add additional addresses, click **Insert address** for as many addresses as you want to add.

6.   Click **Submit Profile** to save the profile.

After you add an address to a profile, you can add additional addresses, custom properties, and contacts to it. You can also view summary information about the profile, make changes to the profile, set up specific ways to look at the profile information, or delete the profile.

How to Add a Contact to a Profile

You can add one or more contacts to both self profiles and partner profiles. You use the profile template **Contacts** tab to add contacts to profiles. For example, you may add both a sales and a billing contact to a profile. For each contact, you can add one or more of the following:

- Telephone number

- Facsimile (Fax) number

- E-mail address

Prerequisites

Note the following requirements:

- You must use the 32-bit version of Internet Explorer to create trading partner profiles. If you installed the 64-bit version of Windows on the computer you use to create profiles, you must install the 32-bit version of Internet Explorer.

- You must log on as a member of one of the following Business Activity Services (BAS) user groups:

  - BAS business users

  - BAS business managers

  - BAS business administrators

To add a contact to a profile

1.   In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site, and then press ENTER.

2.   On the **Business Activity Services Site Home** page, click one of the following:

Use this	To do this
My Profiles	Click to add a contact to a self profile.
Partner Profiles	Click to add a contact to a partner profile.

3.   On the **My Profiles** page or the **Partner Profiles** page, click the profile to which you want to add a contact.

4.   In the Profile template, on the **Contacts** tab, click **Insert Contact**, and then do the following:

Use this	To do this
Name	Type a name for the contact.
Description	Type a description for the contact.
Type	Type the type for the contact type, for example, "primary."
Insert Telephone	Click to type a telephone number and description (for example, main or personal).
Insert Fax	Click to type a facsimile number and description.
Insert Email	Click to type an e-mail address and description.

5.   To add additional contacts, click **Insert contact** for as many sets of contact information as you want to add.

6.   Click **Submit Profile** to save the profile.

After you create the profile, you can add custom properties, addresses, and contacts to it. You can also view summary information about the profile, make changes to the profile, set up specific ways to look at the profile information, or delete the profile.

How to Add Partner Credential Checking to a Profile

You can select a signature certificate for a partner. BizTalk Server supports signing documents (including messages) with a digital signature certificate issued from companies that are official Certificate Authorities (CA). Digital certificates are electronic credentials that identify parties online, so the recipient of the digitally signed document can "trust" that the document is indeed from the sender — making the digital signature as legally binding as if a written signature is on the document.

BizTalk Server verifies the validity of the certificates associated with the incoming signed messages by validating the certificate authority (CA) chain of trust on the

certificate until a root certificate authority is reached. This validates that the certificate used to sign a message is indeed from the identified party. This validation occurs at runtime for each and every signed message.

In addition, BizTalk Server can verify that the certificate authority has not revoked the certificate used to sign or encrypt the message. To do this, you must download the certificate revocation list (CRL) from the certificate authority.

For more information about signature certificates, see the topic "Sending and Receiving Secure Messages" in BizTalk Server Help.

Prerequisites

Note the following requirements:

- You must use the 32-bit version of Internet Explorer to create trading partner profiles. If you installed the 64-bit version of Windows on the computer you use to create profiles, you must install the 32-bit version of Internet Explorer.

- You must log on as a member of one of the following Business Activity Services (BAS) user groups:

  - BAS business managers

  - BAS business administrators

To configure the signature certificate

1. In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site, and then press ENTER.

2. On the **Business Activity Services Site Home** page, click **Partner Profiles**.

3. On the **Partner Profiles** page, click the profile name. The Microsoft Office InfoPath profile template opens.

4. In the Profile template, on the **Advanced** tab, expand **Insert Signature Certificate**, and then click the ellipse.

5. In the **Select Certificate** dialog box, from the **Select certificate** list, select the signature certificate for the partner, and then click **OK**.

6. Click **Submit Profile**.

How to Add a Send Port to a Profile

This section describes associating logical send ports with trading partner profiles.

A send port is the location to which Microsoft® BizTalk® Server sends messages. The name of the port uniquely identifies the location. The send port provides the

technology that BizTalk Server uses to implement the communication action. When you associate a send port with a trading partner profile, it is the port used to send messages to the trading partner.

The send port you associate with a trading partner profile is a logical port. When you activate an agreement for the trading partner, you map the logical port in the profile with a physical port.

In BizTalk Server 2006, there are three ways to create logical send ports:

- A solutions developer uses BizTalk Explorer to create send ports and bind them to an orchestration.

- A BizTalk administrator uses the BizTalk Administration console to create send ports and bind them to an orchestration.

- Business managers use BAS to configure a new send port for a profile.

For every partner profile, you can specify an existing send port, or you can configure a new send port. The following table lists the different types of send ports you can associate with a trading partner profile and the topics that provide instructions for each type.

Send port	Topic
Existing send port	How to Add an Existing Send Port to a Profile
New HTTP send port	How to Configure an HTTP Send Port
New FILE send port	How to Configure a FILE Send Port
New SOAP send port	How to Configure a SOAP Send Port

In This Section

- How to Add an Existing Send Port to a Profile

- How to Configure an HTTP Send Port

- How to Configure a FILE Send Port

- How to Configure a SOAP Send Port

How to Add an Existing Send Port to a Profile

business manager can add an existing send port to a trading partner profile. Solutions developers use BizTalk Explorer to create send ports, and system administrators use the BizTalk Administration console to create send ports.

You can associate one logical send port with one profile. You can associate multiple logical send ports with one physical send port.

Prerequisites

Note the following requirements:

- You must use the 32-bit version of Internet Explorer to create trading partner profiles. If you installed the 64-bit version of Windows on the computer you use to create profiles, you must install the 32-bit version of Internet Explorer.

- You must log on as a member of one of the following Business Activity Services (BAS) user groups:

  - BAS business managers

  - BAS business administrators

To add an existing send port to a profile

1. In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site, and then press ENTER.

2. On the **Business Activity Services Site Home** page, click **Partner Profiles**.

3. On the **Partner Profiles** page, click the profile name.

4. In the **Partner Profiles** template, on the **Advanced** tab, expand **Insert Global Port**, and then do the following:

Use this	To do this
Global port name	Type a friendly name for the send port.
Choose BizTalk send port	Click the ellipse **[…]** to open the **Select Send Port** dialog box.

5. In the **Select Send Port** dialog box, from the **Select send port** list select the port, and then click **OK**.

6. Click **Submit Profile**.

Next Steps

Now that you have associated a send port with the partner profile, you can deploy it or add it to a partner agreement. If you add an undeployed partner profile to an agreement, you must deploy the partner profile before you can activate the agreement.

For information about deploying a partner profile, see How to Deploy a Profile .

For information about partner agreements, see Establishing Partner Relationships with Agreements .

How to Add an Existing Send Port to a Profile

A business manager can add an existing send port to a trading partner profile. Solutions developers use BizTalk Explorer to create send ports, and system administrators use the BizTalk Administration console to create send ports.

You can associate one logical send port with one profile. You can associate multiple logical send ports with one physical send port.

Prerequisites

Note the following requirements:

- You must use the 32-bit version of Internet Explorer to create trading partner profiles. If you installed the 64-bit version of Windows on the computer you use to create profiles, you must install the 32-bit version of Internet Explorer.

- You must log on as a member of one of the following Business Activity Services (BAS) user groups:

  - BAS business managers

  - BAS business administrators

To add an existing send port to a profile

1.   In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site, and then press ENTER.

2.   On the **Business Activity Services Site Home** page, click **Partner Profiles**.

3.   On the **Partner Profiles** page, click the profile name.

4.   In the **Partner Profiles** template, on the **Advanced** tab, expand **Insert Global Port**, and then do the following:

Use this	To do this
Global port name	Type a friendly name for the send port.
Choose BizTalk send port	Click the ellipse **[…]** to open the **Select Send Port** dialog box.

5.   In the **Select Send Port** dialog box, from the **Select send port** list select the port, and then click **OK**.

6.   Click **Submit Profile**.

Next Steps

Now that you have associated a send port with the partner profile, you can deploy it or add it to a partner agreement. If you add an undeployed partner profile to an agreement, you must deploy the partner profile before you can activate the agreement.

For information about deploying a partner profile, see How to Deploy a Profile .

For information about partner agreements, see Establishing Partner Relationships with Agreements .

How to Configure an HTTP Send Port

You can use the BAS site to configure a new HTTP send port for a profile. An HTTP send port specifies the location to which messages are sent or from which messages are received, and that files of the type HTTP are used to implement the communication action.

Prerequisites

Note the following requirements:

- You must use the 32-bit version of Internet Explorer to create trading partner profiles. If you installed the 64-bit version of Windows on the computer you use to create profiles, you must install the 32-bit version of Internet Explorer.

- You must log on as a member of one of the following Business Activity Services (BAS) user groups:

    - BAS business managers

    - BAS business administrators

To create an HTTP send port

1.   In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site, and then press ENTER.

2.   On the **Business Activity Services Site Home** page, click **Partner Profiles**.

3.   On the **Partner Profiles** page, click the profile name.

4.   In the **Partner Profiles** template, on the **Advanced** tab, expand **Insert Global Port**, and then do the following:

Use this	To do this
Global port name	Type a friendly name for the port.
Choose BizTalk send port	Point to the **Choose BizTalk send port** label, click the control that appears to the left of the label, and then from the menu select **Configure send port**.
Pipeline Name	From the drop-down list, select the pipeline you want to use for the port.
Retry Count	Type the number of retries for sending messages.
Retry Interval	Type the amount of time in seconds the port should wait before resending messages.
Transport Type	From the drop-down list, select HTTP.
HTTP Binding Address	Type the address to send HTTP requests. Query strings appended to the base URL might be required.
Insert direction information	From the drop-down list, select Solicit-Response or One-way.
Insert Request Timeout	Type the timeout, in seconds, for the HTTP/HTTPS transmission. If the response is not received within this time, the service logs the error and resubmits the message.
Insert Maximum Redirects	Type the maximum number of times the request can be redirected. However, this property is not implemented by the HTTP send port.
Insert Kerberos Authentication	Select this check box to indicate that the port uses Kerberos Authentication.
Insert Content Type	Type the content type of the request messages.
Insert SSL Client Certificate	Type the thumbprint of the client certificate to use for establishing a Secure Sockets Layer (SSL) connection.
Insert Service Window: From	Type the time at which the send port should begin sending messages. You must type the time in the format hh:mm:ss.
Insert Service	Type the time at which the send port should stop sending

Window: To	messages. You must type the time in the format hh:mm:ss.
Insert Encryption Certificate	Click the ellipse [...] to open the Select Certificate dialog box. In the Select Certificate dialog box, from the Select certificate list, select the certificate for Outbound message encryption, and then click OK.

5.   Click **Submit Profile**.

The send port is added to the profile. You must activate an agreement associated with the profile to establish the business relationship and begin the exchange of messages. For information about creating agreements, see How to Create an Agreement . For information about activating an agreement, see How to Activate an Agreement .

How to Configure a FILE Send Port

You use BAS to configure a new FILE send port for a profile. A FILE send port specifies the location to which messages are sent or from which messages are received, and that files of the type FILE are used to implement the communication action.

Prerequisites

Note the following requirements:

- You must use the 32-bit version of Internet Explorer to create trading partner profiles. If you installed the 64-bit version of Windows on the computer you use to create profiles, you must install the 32-bit version of Internet Explorer.

- You must log on as a member of one of the following Business Activity Services (BAS) user groups:

  - BAS business managers

  - BAS business administrators

To create a File send port

1.   In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site, and then press ENTER.

2.   On the **Business Activity Services Site Home** page, click **Partner Profiles**.

3.   On the **Partner Profiles** page, click the profile name.

4.   In the **Partner Profiles** template, on the **Advanced** tab, expand **Insert Global Port**, and then do the following:

Use this	To do this
Global port name	Type a friendly name for the port.
Choose BizTalk send port	Point to the **Choose BizTalk send port** label, click the control that appears to the left of the label, and then from the menu select **Configure send port**.
Pipeline Name	Select the pipeline from the list of available pipelines on your server
Retry Count	The number of retries for sending the messages
Retry Interval	Interval after which messages will be resent
Transport Type	The transport type to be used: HTTP, File, or SOAP
Destination location	Specify the path to the location on the file system or public share to write the output messages.
File Name	Specify the name of the file where the file send handler writes the message.
Copy Mode	Define the copy mode to use when writing a message to a file. Valid values are: • Append. The file send handler opens a file if it exists and appends a message to the end of the file. If the file does not exist, the file send handler creates a new file. • Overwrite. The file send handler opens a file if it exists and overwrites its content. If the file does not exist, the file send handler creates a new file. • Create new. If file does not exist, the file send handler creates a new file and writes to it. If the file already exists, the file send handler reports an error and then follows common adapter retry logic for send ports. This is a default copy mode for the file send handler.
Allow Cache on Write	Define whether to use file system caching when writing a message to a file. Options: • **False**. Do not use the file system cache.

	• **True**. Use the file system cache. • **Default Value**: False
Insert Service Window	The service window configures the send port to send only at specified times of the day. If not specified, the send port can send messages at any time.
Insert Encryption Certificate	Select the certificate for Outbound message encryption from the list of certificates on your server.

5.   Click **Submit Profile**.

The send port is added to the profile. You must activate an agreement associated with the profile to establish the business relationship and begin the exchange of messages. For information about creating agreements, see How to Create an Agreement . For information about activating an agreement, see How to Activate an Agreement .

How to Configure a SOAP Send Port

You use BAS to configure a new SOAP send port for a profile. A SOAP send port specifies the location to which messages are sent or from which messages are received, and that files of the type SOAP are used to implement the communication action.

Prerequisites

Note the following requirements:

• You must use the 32-bit version of Internet Explorer to create trading partner profiles. If you installed the 64-bit version of Windows on the computer you use to create profiles, you must install the 32-bit version of Internet Explorer.

• You must log on as a member of one of the following Business Activity Services (BAS) user groups:

  • BAS business managers

  • BAS business administrators

To create a SOAP send port

1.   In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site, and then press ENTER.

2.   On the **Business Activity Services Site Home** page, click **Partner Profiles**.

3.   On the **Partner Profiles** page, click the profile name.

4.   In the **Partner Profiles** template, on the **Advanced** tab, expand **Insert Global Port**, and then do the following:

Use this	To do this
Global port name	Type a friendly name for the port.
Choose BizTalk send port	Point to the **Choose BizTalk send port** label, click the control that appears to the left of the label, and then from the menu select **Configure send port**.
Pipeline Name	Select the pipeline from the list of available pipelines on your server
Retry Count	The number of retries for sending the messages
Retry Interval	Interval after which messages will be resent
Transport Type	The transport type to be used: HTTP, File, or SOAP
Address	Specify the address of the Web service you want to call.
Direction	Specify the direction of the port. Options: • Solicit–Response • One–Way
Receive Pipeline	When this is a two-way (i.e., Solicit-Response) port, select the receive pipeline from the list of available pipelines on your server.
Authentication Scheme	Indicate the authentication method used by the Web service you are calling. Options: • Anonymous • NTLM
Insert SSL Client Certificate	Specify the thumbprint of the client certificate to use for establishing a secure connection.
Insert Service Window	The service window configures the send port to send only at specified times of the day. If not specified, the send port can send messages at any time.

Insert Encryption Certificate	Select the certificate for Outbound message encryption from the list of certificates on your server.

5.   Click **Submit Profile**.

The send port is added to the profile. You must activate an agreement associated with the profile to establish the business relationship and begin the exchange of messages. For information about creating agreements, see How to Create an Agreement . For information about activating an agreement, see How to Activate an Agreement .

How to Deploy a Profile

You deploy a profile so that you can activate an agreement for that profile. You must activate an agreement for the profile before you can exchange messages with the partner the profile represents. Deployment makes your system ready to talk to your partner.

Prerequisites

Note the following requirement:

- You must logon as a member of one of the following Windows user groups:

  - BizTalk BAS Managers

  - BizTalk BAS Administrators

To deploy a partner profile

1.   In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site, and then press ENTER.

2.   On the **Business Activity Services Site Home** page, click **Partner Profiles**.

3.   On the **Partner Profiles** page, from the drop-down list box of the partner profile you want to deploy, click **Deploy Partner**.

4.   On the **Deploy Partner** page, select the BizTalk Server registration to which you want to deploy the partner profile, and then click **Deploy**.

Organizing Partner Profiles with Partner Groups

You use partner groups to manage relationships with multiple trading partners. You add partner profiles that you want to manage together to a group. You can associate an agreement with a partner group to apply it to all partner profiles in the group.

All group members inherit the group agreements. Adding a partner profile to a group with an agreement in place automatically creates an identical agreement with the newly added partner profile. Group agreements simplify management of trading partners involved in the same business process.

In This Section

- How to Create a Partner Group

- How to Add and Remove Partner Group Members

- How to View Partner Group Summary Information

- How to Modify a Partner Group

- How to Filter the Partner Groups List

- How to Delete a Partner Group

How to Create a Partner Group

You can group partner profiles. For example, if you have partners in several different locations you might want to group their profiles by region.

Prerequisites

Note the following requirements:

- You must use the 32-bit version of Internet Explorer to create partner groups. If you installed the 64-bit version of Windows on the computer you use to create partner groups, you must install the 32-bit version of Internet Explorer.

- You must log on as a member of one of the following Windows user groups:

  - BAS business users

  - BAS business managers

  - BAS business administrators

To create a profile group

1. In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site.

2. On the Business Activity Services site home page, click **Partner Groups**.

3. On the **Partner Groups** page, click **New Group**.

4. On the **Partner Group Profile** page, do the following:

Ramkumar JD   IntSol

Use this	To do this
Group Name	Type a name for the group.
Description	Type a description for the group.
Alternate Names	Type an alternate name for the group. You can add multiple names by pressing Enter after each name.

How to Add and Remove Partner Group Members

In the BAS Windows SharePoint Services collaboration site, you can add members to or remove members from a partner group.

Prerequisites

Note the following requirement:

- You must log on as a member of one of the following Windows user groups:

  - BAS business users

  - BAS business managers

  - BAS business administrators

To add or remove partner group members

1. In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site, and then press ENTER.

2. On the **Business Activity Services Site Home** page, click **Partner Profiles**.

3. On the **Partner Profiles** page, in the list of partner profiles, point to the partner you want to add or remove from a partner group, and from the drop-down list, click **Join/Leave Groups**.

4. In the **Join/Leave Groups: <partner name>** page, select the check boxes of the groups you want the profile to belong to and clear the check boxes of the groups you do not want the profile to belong to.

5. Click **Submit**.

How to View Partner Group Summary Information

In the BAS Windows SharePoint Services collaboration site, you can view a summary of information about a partner group.

Microsoft BizTalk Server 2006   63

Prerequisites

Note the following requirement:

- You must log on as a member of one of the following Windows user groups:

  - BAS business users

  - BAS business managers

  - BAS business administrators

To view partner group information

1. In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site, and then press ENTER.

2. On the **Business Activity Services Site Home** page, click **Partner Groups**.

3. On the **Partner Groups** page, from the drop-down list box of the group for which you want to view summary information, select **Summary View**.

    The **Partner Groups: <group name>** page contains key information about the partner group. It shows you the name and a description of the partner group. The bottom half of the page displays the parent groups this partner group belongs to, the members (partners and groups) of this group, and the agreements this group is a part of.

How to Modify a Partner Group

the BAS Windows SharePoint Services collaboration site, you can make changes to an existing partner group.

Prerequisites

Note the following requirements:

- You must use the 32-bit version of Internet Explorer to modify partner groups. If you installed the 64-bit version of Windows on the computer you use to modify partner groups, you must install the 32-bit version of Internet Explorer.

- You must log on as a member of one of the following Windows user groups:

  - BAS business users

  - BAS business managers

  - BAS business administrators

To modify a partner group

1.   In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site, and then press ENTER.

2.   On the **Business Activity Services Site Home** page, click **Partner Groups**.

3.   On the **Partner Groups** page, from the drop-down list box of the partner you want to modify, click **Edit Item**.

4.   In the **Partner Group Profile** template, do the following:

Use this	To do this
Group name	You cannot modify the group name. If you want to change a group name, you must delete the group and then recreate it with the new name. For information about deleting a group, see <u>How to Delete a Partner Group</u> . For information about creating a group, see <u>How to Create a Partner Group</u> .
Description	Type a new description for the partner group or modify the existing one.
Alternative names	Type alternative names for the group.

5.   Click **Submit Profile**.

How to Filter the Partner Groups List

You use the filter option on the **Partner Groups** page to query partner groups. The filter you set determines the partner groups that appear in the partner group list on the **Partner Groups** page. You can use one filter criterion to query the partner groups at a time. You can filter partner groups by:

- Type

- Name

- Description

- Member Type

Prerequisites

Note the following requirement:

- You must log on as a member of one of the following Windows user groups:

    - BAS business users

- BAS business managers

- BAS business administrators

To filter your partner groups

1.   In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site, and then press ENTER.

2.   On the **Business Activity Services Site Home** page, click **Filter**, and then do one of the following:

Use this	To do this
Type	From the drop-down list, select the type of partner group for which you want to query the partner groups.
Name	From the drop-down list, select the partner group name for which you want to query the partner groups.
Description	From the drop-down list, select the description for which you want to query the partner groups.
Member Type	From the drop-down list, select the member type for which you want to query the partner groups.

4.   After you have created a filter, to go back to seeing all of your groups, click **Change Filter** and select **All** from the drop-down lists.

How to Delete a Partner Group

In the BAS Windows SharePoint Services collaboration site, you can delete a partner group.

Prerequisites

Note the following requirement:

- You must log on as a member of one of the following Windows user groups:

  - BAS business users

  - BAS business managers

  - BAS business administrators

Considerations

Note the following when you delete partner groups

- You cannot delete a group that has members. For information about removing members from a group, see <u>How to Add and Remove Partner Group Members</u>.

- You cannot delete a group associated with an active agreement. You must deactivate the agreement before you can delete the partner group. For information about deactivating an agreement, see <u>How to Deactivate an Agreement</u>.

To delete a partner group

1. In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site, and then press ENTER.

2. On the **Business Activity Services Site Home** page, click **Partner Groups**.

3. On the **Partner Groups** page, from the drop-down list box of the group you want to delete, click **Delete Item**.

4. In the **Are you sure you want to delete this item** message box, click **OK**.

Establishing Partner Relationships with Agreements

An agreement represents the relationship between partners based on business processes. You can create an agreement between a self profile for your organization and a trading partner profile or partner group. (Partner groups contain trading partner profiles you manage together).

Agreements contain the following types of information:

- **Agreement name**. The agreement name uniquely identifies the agreement.

- **Self profile**. The profile that represents your organization in the agreement.

- **Trading partner profile**. The profile that represents the trading partner or group of partners (partner group) in the agreement. You specify a partner group in an agreement when the trading partners in the group have identical trading relationships with your organization. When you associate a partner group with an agreement, all of the members of the group inherit that agreement.

- **Addendum terms**. The addendum terms field is a text field you use to store information for the agreement.

- **Business relationship**. The business relationship is the representation of a Role Link Type in the BAS site. A Role Link Type contains technical information about how two roles (such as buyer and seller) exchange information. A solutions developer defines Role Link Types when enabling an orchestration to work with BAS.

- **Legal terms**. The legal terms field is a text field you use to store information for the agreement.

Each agreement must have a least one addendum. For each addendum you must specify a unique name and a business relationship. Additionally, you can specify a friendly name for the addendum and addendum terms. You specify one or more addendums for each agreement.

A business relationship (Role Link Type) can add additional parameters to an agreement. The solutions developer defines these parameters in the project that contains the Role Link Type. The parameters are used to collect additional information in the agreement.

After you create the agreement, a business manager must activate the agreement. When you activate an agreement, you map the logical send ports specified in the partner profile (or partner group) with a physical port. Activating an agreement starts the trading relations and enables you to exchange messages with the partner or partners specified in the agreement. You can create agreement templates by saving an agreement to a local file and then copying and making the changes as needed.

The following table provides a workflow for creating agreements using the BAS site.

Task	Topic
A business user creates a new agreement and specifies the agreement name, the self profile, and the trading partner profile for the agreement.	How to Create an Agreement
A business user adds one or more addendums to the agreement, specifying the addendum name and business relationship for each addendum. Additionally, the business manager may add a friendly name and addendum terms to each addendum.	How to Add an Addendum to an Agreement
A business user adds legal terms to the agreement.	How to Add a Legal Term to an Agreement
A business manager activates the agreement.	How to Activate an Agreement

For information about changing, deactivating, or deleting partner agreements, see Modifying Trading Partner Agreements.

In This Section

- How to Create an Agreement

- How to Add an Addendum to an Agreement

- How to Add a Legal Term to an Agreement

- How to Activate an Agreement

How to Create an Agreement

Business users use the BAS **Agreement** template to set up and manage partner relationships. Agreements define the relationship between your organization and partner organizations. You can define an agreement between your organization and a partner, or between your organization and a partner group.

A solutions developer configures business processes for an agreement by associating roles with the appropriate profiles and setting up the appropriate business parameters. This enables you, the business user, to participate in business processes. You can make changes to business rules without modifying the lower-level business process code, which requires a developer. You can change the business parameters without recompiling the orchestrations, which would involve developers or consultants in the process.

Every agreement specifies the two profiles used in the business relationship. One is the self profile and the other is the partner or partner group profile. If you specify a partner group, BizTalk creates an agreement for each partner within that group. An agreement consists of one or more addendums. Each addendum is based on a particular business process, and specifies what roles will be played by their organization (self), and the partner's role. It also contains information about the parameters used in this particular relationship.

Prerequisites

Note the following requirements:

- You must use the 32-bit version of Internet Explorer to create agreements. If you installed the 64-bit version of Windows on the computer you use to create profiles, you must install the 32-bit version of Internet Explorer.

- You must create at least one self profile and one trading partner profile before you can create a partner agreement. For information about creating profiles see Creating Profiles for Trading Partners [BTS2006].

- You must log on as a member of one of the following Windows user groups:

    - BAS business users

    - BAS business managers

    - BAS business administrators

To create an agreement

1. In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site, and then press ENTER.

2. On the **Business Activity Services Site Home** page, click **Agreements**.

3.   On the **Agreements** page, click **New Agreement**.

4.   In the **Agreement** template, on the **General** tab, do the following:

Use this	To do this
Agreement name	Type a name for this agreement.
Description	Type a description of the agreement.
Agreement date	Click the calendar icon, and then click the date of the agreement.
Insert Duration	Click to insert the duration of the agreement. Type a start date and an end date for the duration, or use the calendar picker to choose the start and end dates. .
Select self profile	Click the ellipse **[…]**, and in the **Select My Profile** dialog box, from the **Select profile** list, select the self profile for the agreement, and then click OK.
Select partner profile	Click the ellipse **[…]**, and in the **Select Partner Profile** dialog box, from the **Select profile** list, select the partner profile for the agreement, and then click **OK**.

5.   Click **Submit Agreement**.

After you create an agreement, you can add an addendum and a business rule to it. For information about adding an addendum, see How to Add an Addendum to an Agreement. For information about adding a business rule, see How to Add a Legal Term to an Agreement.

How to Add an Addendum to an Agreement

You use the **Agreement** template, **Addendums** tab to add addendums to an agreement. An agreement consists of multiple segments called addendums. The different addendums define what business process is used, your role, your partner's role, and what policies or parameters are used in that relationship along with documentation such as business and legal terms.

Prerequisites

Note the following requirements:

- You must use the 32-bit version of Internet Explorer to create agreements. If you installed the 64-bit version of Windows on the computer you use to create profiles, you must install the 32-bit version of Internet Explorer.

- You must log on as a member of one of the following Windows user groups:

  - BAS business users

  - BAS business managers

  - BAS business administrators

⊟Procedure Title

1.   In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site, and then press ENTER.

2.   On the **Business Activity Services Site Home** page, click **Agreements**.

3.   On the **Agreements** page, click the agreement to which you want to add an addendum.

4.   In the **Agreement** template, on the **Addendums** tab, do the following:

Use this	To do this
Addendum Name	Type a name for the addendum.
Friendly Name (Optional)	Type an easy-to-remember name for the specified addendum.
Addendum Terms (Optional)	Add terms for your agreement, for example, for a purchase order "Void after 30 days" for addendum
Business Relationship Name	Click the ellipse **[…]** to open the **Select Relationship** dialog box. From the **Select relationship** list, select the role link type that implements the trading relationship electronically, and then click **OK**.
Parameters	Define the value of parameter that you want to use for this partner and this relationship. (Parameters are shown only if the underlying business process defines them)

5.   Click **Submit Agreement**.

After you add an addendum to an agreement, you can add legal terms to the agreement. For information about adding legal terms to an agreement, see How to Add a Legal Term to an Agreement. The Agreement does not go into effect until a

BAS Business Manager activates the agreement. For information about activating an agreement, see How to Activate an Agreement .

How to Add a Legal Term to an Agreement

You use the **Legal Terms** tab of the agreement template to add legal terms to an agreement. You can use the Legal terms field to type any relevant information relating to the agreement.

Prerequisites

Note the following requirements:

- You must use the 32-bit version of Internet Explorer to create agreements. If you installed the 64-bit version of Windows on the computer you use to create profiles, you must install the 32-bit version of Internet Explorer.

- You must log on as a member of one of the following Windows user groups:

    - BAS business users

    - BAS business managers

    - BAS business administrators

To add legal terms to an agreement

1. In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site, and then press ENTER.

2. On the **Business Activity Services Site Home** page, click **Agreements**.

3. On the **Agreements** page, click the agreement to which you want to add legal terms.

4. On the **Legal Terms** tab, in the **Legal Terms** box, type the legal terms for the agreement.

5. Click **Submit Agreement**.

After you add legal terms to an agreement, a BAS business manager must activate the agreement to begin the business relationship defined in the agreement. For information about activating an agreement, see How to Activate an

How to Activate an Agreement

When you activate an agreement, it activates a trading relationship for that partner or group of partners whose profiles are associated with the agreement.

Prerequisites

Note the following requirements:

- You must use the 32-bit version of Internet Explorer to activate agreements. If you installed the 64-bit version of Windows on the computer you use to create profiles, you must install the 32-bit version of Internet Explorer.

- You must log on as a member of one of the following Windows user groups:

  - BAS business managers

  - BAS business administrators

To activate an agreement

1. In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site, and then press ENTER.

2. On the **Business Activity Services Site Home** page, click **Agreements**.

3. On the **Agreements** page, from the drop-down list for the agreement you want to activate, click **Activate Agreement**.

4. On the **Agreement Mapping** page, for the first **Operation port**, click the ellipse […].

5. In the **Select Port** dialog box, from the **Select Port** list, select the port for the operation, and then click **OK**.

6. Repeat steps 4 and 5 for each **Operation port**.

7. Click **Submit**.

The agreement is active. If you want to edit the agreement, you must first deactivate it. For information about deactivating an agreement, see Deactivating an agreement.

Exchanging Messages with Partners and Processes

In the BAS Microsoft® Windows® SharePoint™ Services collaboration site, business users interact with partners and business processes through messages in the inbox and outbox document libraries.

Business users use the **inbox** document library to access messages received from a partner or process.

Business users use the **outbox** document library to send messages to partners and processes.

Business users use the **sent items** document library to access messages sent to a partner or process.

Every trading partner profile has an associated inbox document library and an associated outbox document library.

The sent items document library is associated with the BizTalk Server installation registered with the BAS site.

Opening a document from the inbox loads a pre-defined template associated with a message so business users see InfoPath forms instead of raw XML when they open the documents located in the Inbox.

A solutions developer creates InfoPath solution templates for business documents, and imports the XML schemas used in the orchestrations to the BAS site. These schemas define the documents exchanged by partners in a business transaction, such as a purchase order or an invoice.

When an orchestration saves a message to the Inbox, the BAS site looks up the associated partner document library and the global Templates document library to find a matching InfoPath solution template and inserts an InfoPath specific processing instruction to the Inbox messages if it finds one.

A business user responds to a business process by saving Microsoft Office documents in a specific document library such as the Outbox. When you save a document into the Outbox document library, the business process picks it up as a response to the previous Inbox message.

In This Section

How to Access Messages Exchanged with a Partner

How to Send Messages to Business Processes

How to Filter the Messages List

How to Access Messages Exchanged with a Partner

You use the partner inbox document library to view documents received from the partner. You use the partner sent items document library to view all documents sent to the partner.

A solutions developer must enable a business process for BAS before documents exchanged with partners will appear in the inbox and sent items document libraries. For information about enabling business processes for BAS, see the "Creating BAS-Enabled Orchestrations" topic in BizTalk Server 2006 Help.

Prerequisites

Note the following requirement:

- You must log on as a member of one of the following Windows user groups:

  - BAS business users

  - BAS business managers

  - BAS business administrators

To view Business Messages received from a partner

1. In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site, and then press ENTER.

2. On the **Business Activity Services Site Home** page, click **Partner Profiles**.

3. In the list of partner profiles, on the same line as the partner for which you want to see the inbox documents library, click **Inbox**.

4. To view a particular message, click the message title.

To view Business Messages sent to a partner

1. In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site.

2. From the **Quick Launch** navigation pane, click **Partner Profiles**.

3. In the list of partner profiles, on the same line as the partner for which you want to see the sent items document library, click **Sent Items**.

4. To view a particular message, click the message title.

How to Send Messages to Business Processes

You use the BizTalk server outbox document library to send messages to partners. You can send one message at a time, or you can send multiple messages to the server outbox document library.

A solutions developer must enable a business process for BAS before BAS forwards to partners documents sent to the server outbox document library. For information about enabling business processes for BAS, see the "Creating BAS-Enabled Orchestrations" topic in BizTalk Server 2006 Help.

A BAS administrator must register the BizTalk server before you can add messages to the outbox document library. For information about registering a BizTalk server, see How to Connect the BAS Site to BizTalk Server .

Prerequisites

Note the following requirement:

- You must log on as a member of one of the following Windows user groups:

  - BAS business users

  - BAS business managers

  - BAS business administrators

To send a single message to a server outbox document library

1.   In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site, and then press ENTER.

2.   On the **Business Activity Services Site Home** page, click **BizTalk Servers**.

3.   In the **BizTalk Servers** page, on the same line as the server, click **Outbox**.

4.   In the **<server name> Outbox** document library, click Upload Document.

5.   In **<server name> Outbox: Upload Document** page, do the following:

Use this	To do this
Overwrite existing files	Select this check box if you want the file you add to the document library to overwrite a file currently in the outbox document library if the two files have the same name.
Name	Type the name of the file, or click Browse to select the file from a list.
Upload multiple files	Do not click Upload multiple files if you are adding a single file to the document library.
Status	View the status of the file upload process for the document library.
Description	View the description of the status.

6.   Click **Save and Close** to save the outbox document library with the new document.

To send multiple messages to a server outbox document library

1.   In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site.

2.   From the **Quick Launch** navigation pane, click **BizTalk Servers**.

3.   In the **BizTalk Servers** page, on the same line as the server, click **Outbox**.

4.   In the **<server name> Outbox** document library, click Upload Document.

5.   In **<server name> Outbox: Upload Document** page, click **Upload multiple files**, and then select the files you want to add to the outbox document library.

6.   Click **Save and Close** to save the document library with the new files and return to the outbox document library.

How to Filter the Messages List

You use the filter option to query messages in the inbox and sent items document libraries. The filter you set determines the documents that appear in the document list in the document libraries. You can use one filter criterion to query the messages at a time. You can filter your messages by:

*   Type

*   Modified

*   Modified By

*   Checked Out To

*   Template

*   Namespace

*   Business Process

*   Status

*   Description

The filter options automatically provide you with details from the documents in the library. For example, if you choose to query for all files uploaded on the same day, the **Modified** drop-down list lists the dates on which documents in the library were uploaded.

Prerequisites

Note the following requirement:

*   You must log on as a member of one of the following Windows user groups:

    *   BAS business users

    *   BAS business managers

- BAS business administrators

To filter your business process messages

1. In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site, and then press ENTER.

2. On the **Business Activity Services Site Home** page, click **Partner Profiles**.

3. For the partner, click **Inbox** or **Sent items**.

4. In the partner inbox, or the partner outbox, click **Filter**, and then do one of the following:

Use this	To do this
Type	From the drop-down list, select the type of document for which you want to query the documents.
Modified	From the drop-down list, select the date on which you want to query the documents.
Modified by	From the drop-down list, select the person for which you want to query the documents.
Checked out to	From the drop-down list, select the person for which you want to query the documents.
Template	From the drop-down list, select the template for which you want to query the documents.
Namespace	From the drop-down list, select the namespace for which you want to query the documents.
Business process	From the drop-down list, select the business process for which you want to query the documents.
Status	From the drop-down list, select the status for which you want to query the documents.
Description	From the drop-down list, select the description for which you want to query the documents.

6. After you have created a filter, to see all of your messages, click **Change Filter** and select **All** from the drop-down lists.

Managing Partner Data

Business users use the Business Activity Services (BAS) site to create objects (such as profiles) and data about objects (such as the name of a profile). The data that

business users add to the BAS site is stored in the Trading Partner Management (TPM) database.

The data stored in the TPM database is also stored in the BizTalk Management database (BizTalkMgmtDb). Systems administrators and Solutions developers can change BAS data in the BizTalk Management database. Systems administrators use the BizTalk Administration console to change BAS data. Solutions developers use BizTalk Explorer to change BAS data.

When systems administrators and solution developers change the BAS data in the BizTalk Management database, these changes are not made in the TPM database. In this case, the BAS data in the TPM database is not the same as the BAS data in the BizTalk Management database.

Additionally, sometimes the data shown in the BAS site does not match the data stored in the TPM database. This can happen, for example, if a systems administrator updates the TPM database when Internet Information Services (IIS) is stopped.

To maintain BAS data on the BAS site, a BAS administrator periodically performs two database maintenance tasks to make sure that the data in the BAS site, the BAS data in the TPE database, and the BAS data in the BizTalk Management database is the same:

- **Synchronize**. The synchronize command (called Resync) refreshes the BAS site with the data from the TPM database.

- **Repair**. The repair command refreshes the BizTalk Management database with data from the TPM database.

In This Section

- Why Synchronize BAS Data?

- Why Repair BAS Data?

- How to Synchronize BAS Data

- How to Repair BAS Data

Why Synchronize BAS Data?

A BAS administrator runs the Sync command to refresh the Business Activity Services (BAS) Windows SharePoint site with data from the Trading Partner Management (TPM) database. The data users enter through the Business Activity Services (BAS) site is stored in the TPM database. Sometimes, the data shown in the BAS site does not match the data stored in the TPM database.

The data shown in the BAS site can be out of date. For example, if a systems administrator updates the TPM database when Internet Information Services (IIS) is stopped. An example of out-of-date data is when you have created an agreement,

the agreement is active in the TPM database, you have activated it, and the site displays it as inactive.

We recommend that you periodically synchronize the BAS site with the TPM database as part of your database maintenance strategy. Additionally, you would synchronize the BAS site with the TPM database as the first step in troubleshooting any unexpected behavior that occurs when BAS processes a request from a user on the site. For information about running the Sync command.

When you run the Synch command, the synchronization process runs the repair process automatically if any partner profile or BizTalk registration URL changes. For information about the repair process,

The following table describes how the synchronization process affects items on the BAS Windows SharePoint Services site.

BAS site item	Synchronize process actions
My profiles list	• Refreshes my profiles list items and adds any that are missing
Partner profiles list	• Refreshes partner profiles list items and adds any that are missing • Updates status • Refreshes inbox document library and re-creates it if necessary • Refreshes sent items document library and re-creates it if necessary
Partner groups list	• Refreshes partner groups list items and adds any that are missing • Updates agreement name • Updates description • Updates status
Agreements list	• Refreshes agreements list items and adds any that are missing • Updates description • Updates status
BizTalk Servers	• Refreshes BizTalk Servers list items and adds any that are

list	missing
	• Refreshes registration name and re-creates it if necessary
	• Refreshes outbox document library and re-creates it if necessary
	• Refreshes BizTalk Management database server name
	• Refreshes BizTalk Management database
	• Refreshes BizTalk Server computer name

Why Repair BAS Data?

A BAS administrator runs the Repair command to refresh the BizTalk Management database (BizTalkMgmtDb) with data from the Trading Partner Management (TPM) database. You must repair the data in the BizTalk Management database because sometimes the data in the BizTalk Management database does not match the data in the TPM database.

Data from the Business Activity Services (BAS) site is stored in both the TPM and the BizTalk Management database. The BizTalk Management database contains artifacts that result from user actions on the Business Activity Services (BAS) Web site. For example, when you deploy a partner profile, BizTalk creates a corresponding party in the BizTalk Management database. When you activate an agreement, BizTalk creates the corresponding send ports, enlists the send ports under parties, and enlists the parties into roles in the BizTalk Management database.

Developers using BizTalk Explorer in Microsoft Visual Studio or administrators in the Administration console can modify BAS artifact properties or states stored in the BizTalk Management database without updating the data in the TPM database. This may cause business processes (orchestrations) configured through the BAS site to function incorrectly.

For example, a developer might change a send port address originally created by BAS agreement activation. As a result, the orchestration routes messages to an invalid URL.

We recommend that you periodically repair the data in the BizTalk Management database with the data in the TPM database. For information about running the Repair command, see How to Repair BAS Data .

We recommend that you do not delete a profile Inbox or Outbox, because the Inbox and Outbox are not repaired when you repair BAS data. For information about the Inbox and Outbox, see Exchanging Messages with Partners and Processes . For information about repairing BAS data,

Some BizTalk artifact properties are stored in the BizTalk Management database, but are not stored in the TPM database. If a property is not stored in the TPM database, the repair process will not restore it in the BizTalk Management database. The following properties (accessible from BizTalk Explorer in Visual Studio .NET) are not repaired because they are not stored in the TPM database:

- SendPort/Transport/Primary/Address/Proxy/ (everything in this properties group)

- SendPort/Transport/Secondary/ (everything in this properties group)

- SendPort/Filters & Maps/ (everything in this properties group)

- SendPort/Send/Tracking Type

- SendPort/Send/Priority

If you stop the Microsoft SQL Server service running on the computer on which the Windows SharePoint Services database or TPM database is located, while the repair process is running, the repair will fail. You must first restart Internet Information Services (IIS) and then restart SQL Server before repairing the database.

The following table describes how the repair process affects BizTalk artifacts in the BizTalk Management database.

BizTalk artifact	Repair process action
STS outbox receive port	Ensures that this port exists
STS outbox receive location	Ensures that the properties are correct
Parties (deployed partners)	Ensures that there is a corresponding party for each deployed partner, and that the party alias and send port values are correct
Send ports (for agreements)	Ensures that ports created during agreement activation exist and their values are correct
Party role enlistments (for agreements)	Ensures that party role enlistments created during agreement activation exist and their operation mappings are corr

How to Synchronize BAS Data

BAS administrator synchronizes the Business Activity Services (BAS) site with the Trading Partner Management (TPM) database to refresh the site with data from the TPM database. You synchronize the BAS site as part of your overall database

maintenance strategy and as the first step in troubleshooting any unexpected behavior that occurs when BAS processes a request from a user on the site. For information about what happens when you synchronize BAS data, see <u>Why Synchronize BAS Data?</u>

Prerequisites

Note the following requirement:

- You must log on as a member of the following Windows user group:

    - BAS business administrators

To synchronize the BAS site with the TPM database

1. In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site, and then press ENTER.

2. On the **Business Activity Services Site Home** page, click **TPM Admin** at the top of the page.

3. On the **TPM Administration** page, click **Resync**.

If an error occurs during the synchronization process, an error page appears on the site. Use the Event Viewer on the Windows SharePoint Services server to check for any errors from KwTpm that occur during the synchronization process.

How to Repair BAS Data

A BAS administrator runs the Repair command to refresh the BizTalk Management database (BizTalkMgmtDb) with data from the Trading Partner Management (TPM) database. For information about what happens when you repair BAS data, see <u>Why Repair BAS Data?</u>

Prerequisites

Note the following requirement:

- You must log on as a member of the following Windows user group:

    - BAS business administrators

To repair BAS artifacts in the BizTalk Management database

1. In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site, and then press ENTER.

2. On the **Business Activity Services Site Home** page, click **TPM Admin** at the top of the page.

3. On the TPM Administration page, **Repair**.

4.   If an error occurs, an error page appears. Check the Event Viewer on the Windows SharePoint Services server for any errors from KwTpm that occur during the synchronization process.

Modifying Partner Profiles

You may want to modify a partner profile. The changes you can make to a partner profile depend on the following:

- The permissions you have on the BAS site

- Whether the profile you want to modify is deployed or not

- Whether any agreement associated with the profile is activated or not.

If you are a member of the BAS business users group, you can change the data on the General Information, Addresses, and Contacts tabs of the profile template. Additionally, you can delete profiles that are not deployed.

If you are a member of the BAS business managers group, you can change the data on all of the tabs of the profile template, including the Advanced tab, and you can deploy and undeploy profiles.

In This Section

- How to Filter the Profile List

- How to View Profile Summary Information

- How to Edit a Profile

- How to Undeploy a Profile

- How to Delete a Profile

How to Filter the Profile List

You use the filter option to query profiles. The filter you set determines the profiles that appear in the profile list on the My Profiles and Partner Profiles pages. You can use one filter criterion to query the profiles at a time. You can filter your profiles by:

- Type

- Name

- Description

- Member Type

Prerequisites

Note the following requirement:

- You must log on as a member of one of the following Windows user groups:

  - BAS business users

  - BAS business managers

  - BAS business administrators

To filter the profile list

1.   In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site, and then press ENTER.

2.   On the **Business Activity Services Site Home** page, click one of the following:

Use this	To do this
My Profiles	Click to access the list of self profiles.
Partner Profiles	Click to access the list of partner profiles.

3.   On the **My Profiles** page or the **Partner Profiles** page, click **Filter**, and then do one of the following:

Use this	To do this
Type	From the drop-down list, select a profile type for which you want to query the profile list.
Name	From the drop-down list, select a profile name for which you want to query the profile list.
Description	From the drop-down list, select a profile description for which you want to query the profile list.
Member Type	From the drop-down list, select a profile member type for which you want to query the profile list.

4.   After you have created a filter, to remove the filter, click **Change Filter**, and then from any of the filter drop-down lists, select **All**.

How to View Profile Summary Information

You can view a summary of a profile. The summary view displays the profile name and what partner groups and agreements are associated with the profile.

Prerequisites

Note the following requirements:

- You must use the 32-bit version of Internet Explorer to view profile data. If you installed the 64-bit version of Windows on the computer you use to create profiles, you must install the 32-bit version of Internet Explorer.

- You must log on as a member of one of the following Windows user groups:

  - BAS business users

  - BAS business managers

  - BAS business administrators

To view a profile summary

1. In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site, and then press ENTER.

2. On the **Business Activity Services Site Home** page, click one of the following:

Use this	To do this
My Profiles	Click to view a summary of a self profile.
Partner Profiles	Click to view a summary of a partner profile.

3. On the **My Profiles** page or the **Partner Profiles** page, point to the profile name, and from the drop-down list, select **Summary View**.

The summary page displays information about the profile, including the Parent Groups and Agreements with which the profile is associated.

How to Edit a Profile

You can edit both self profiles and partner profiles. You cannot edit profile names. If you want to change a profile name, delete the profile and create a new profile with the new name. For information about deleting a profile.

Prerequisites

Note the following requirements:

- You must use the 32-bit version of Internet Explorer to update profiles. If you installed the 64-bit version of Windows on the computer you use to create profiles, you must install the 32-bit version of Internet Explorer.

- You must log on as a member of one of the following Windows user groups:

  - BAS business users

  - BAS business managers

  - BAS business administrators

To edit a profile

1. In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site, and then press ENTER.

2. On the **Business Activity Services Site Home** page, click one of the following:

Use this	To do this
My Profiles	Click to edit a self profile.
Partner Profiles	Click to edit a partner profile.

3. On the **My Profiles** page or the **Partner Profiles** page, click the profile name.

4. In the profile template, do the following:

Use this	To do this
Name	View the profile name.
Description	Edit the profile description.
Insert alternate names	Edit the profile alternate names.
Duns number	Edit the partner Dun & Bradstreet DUNS number. The DUNS number is a unique nine-digit identification sequence that provides a unique identifier for a single business entity and links corporate family structures together.
Internal ID	Edit the ID you use to refer to the profile internally in your organization.
Currency	From the drop-down list, select the preferred currency for this profile.
Payment method	Edit the payment method for the profile.

Tax exempt	Select **Yes** or **No** to indicate whether the company is tax exempt or not.
Tax ID	Edit the Tax ID for this profile.
Term ID	Edit the Term ID for this profile.
Insert custom property	Edit the custom property if any, or click to insert a custom property. For information about adding a custom property to a profile, see How to Add a Custom Property to a Profile.

5.   Click **Submit Profile** to save the profile.

How to Undeploy a Profile

A BAS business manager can undeploy a profile to stop doing business with the partner. You can redeploy an undeployed profile if you want to resume doing business with the partner.

Prerequisites

Note the following requirement:

- You must log on as a member of one of the following Windows user groups:

  - BAS business managers

  - BAS business administrators

Considerations

Note the following considerations:

- A partner profile must be deployed before you can undeploy it.

- When you undeploy a partner profile, the undeployment process recreates the Inbox and Outbox for the partner profile. We recommend that you do not delete a profile Inbox or Outbox, because the Inbox and Outbox are not repaired when you repair BAS data. For information about the Inbox and Outbox, see Exchanging Messages with Partners and Processes . For information about repairing BAS data, see Why Repair BAS Data?

- If a profile is associated with an active agreement, you must deactivate the agreement before you can undeploy the profile. You must redeploy the partner profile or update the agreement with a deployed profile before you can reactivate the agreement. You cannot activate an agreement if any of the associated partner profiles is not deployed. For information about deactivating an agreement, see How to Deactivate an Agreement .

- If a profile is a member of a partner group and the partner group is associated with an active agreement, you must deactivate the agreement, and then redeploy the profile or remove it from the partner group, before you can reactivate the agreement. For information about removing a profile from a partner group, see How to Add and Remove Partner Group Members .

- If you want to remove a partner profile altogether, you must first undeploy it, and then delete it. For information about deleting a partner profile, see How to Delete a Profile .

To undeploy a partner profile

1. In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site, and then press ENTER.

2. On the **Business Activity Services Site Home** page, click **Partner Profiles**.

3. In the **Partner Profiles** page, point to the profile, and from the drop-down list box click **Undeploy Partner**.

The profile is undeployed. If you want to resume exchanging messages with the partner, you must deploy the profile again and reactivate the agreement it is associated with. For information about deploying a profile, see How to Deploy a Profile . For information about activating an agreement, see How to Activate an Agreement .

How to Delete a Profile

You can delete a profile that no agreement references. For information about agreements,

Prerequisites

Note the following requirements:

- You must log on as a member of one of the following Windows user groups:

  - BAS business users

  - BAS business managers

  - BAS business administrators

- Before you delete a profile, you must first undeploy it, and then delete it. For information about undeploying a partner profile,

To delete a profile

1.   In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site, and then press ENTER.

2.   On the **Business Activity Services Site Home** page, click one of the following:

Use this	To do this
My Profiles	Click to delete a self profile.
Partner Profiles	Click to delete a partner profile.

3.   On the **My Profiles** page or the **Partner Profiles** page, point to the profile name, and from the drop-down list, select **Delete Item**.

4.   In the **Are you sure you want to delete this item?** message box, click **OK**.

Modifying Trading Partner Agreements

Business users can update and delete agreements. Business managers can deactivate active agreements. If an agreement is active, a business manager must deactivate it before a business user can delete it.

In This Section

- How to Filter the Agreements List

- How to View Agreement Summary Information

- How to Deactivate an Agreement

- How to Delete an Agreement

How to Filter the Agreements List

In the BAS Windows SharePoint Services collaboration site, when you have a list of agreements, you can set up specific views of the agreements using a filter. You can filter your agreements by:

- Name

- Description

- Status

Prerequisites

Note the following requirements:

- You must use the 32-bit version of Internet Explorer to view agreement information. If you installed the 64-bit version of Windows on the computer you use to create profiles, you must install the 32-bit version of Internet Explorer.

- You must log on as a member of one of the following Windows user groups:

  - BAS business users

  - BAS business managers

  - BAS business administrators

To filter your agreements

1. In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site, and then press ENTER.

2. On the **Business Activity Services Site Home** page, click **Agreements**.

3. On the **Agreements** page, click **Filter**, and then do one of the following:

Use this	To do this
Name	To filter agreements by name, from the drop-down list, select the name that you want to use to filter the agreements.
Description	To filter agreements by description, from the drop-down list, select the description you want to use to filter the agreements.
Status	To filter agreements by status, from the drop-down list, select the status you want to use to filter the agreements.

5. After you have created a filter, to go back to seeing all of your agreements, click **Change Filter** and select **All** from the drop-down lists.

How to View Agreement Summary Information

In the BAS Windows SharePoint Services collaboration site, you can view information about your agreement in the summary view.

Prerequisites

Note the following requirements:

- You must use the 32-bit version of Internet Explorer to view agreement information. If you installed the 64-bit version of Windows on the computer you use to create profiles, you must install the 32-bit version of Internet Explorer.

- You must log on as a member of one of the following Windows user groups:

  - BAS business users

  - BAS business managers

  - BAS business administrators

To view agreement summary information

1. In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site, and then press ENTER.

2. On the **Business Activity Services Site Home** page, click **Agreements**.

3. On the **Agreements** page, from the drop-down list of the agreement you want to view, click **Summary View**.

This page contains key information about this agreement.

How to Modify an Agreement

In the BAS Windows SharePoint Services collaboration site, you can make changes to your agreement. For information about agreements, see Establishing Partner Relationships with Agreements .

Prerequisites

Note the following requirements:

- You must use the 32-bit version of Internet Explorer to modify agreements. If you installed the 64-bit version of Windows on the computer you use to create profiles, you must install the 32-bit version of Internet Explorer.

- You must log on as a member of one of the following Windows user groups:

  - BAS business users

  - BAS business managers

  - BAS business administrators

To modify an agreement

1. In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site, and then press ENTER.

2.   On the **Business Activity Services Site Home** page, click **Agreements**.

3.   On the **Agreements** page, from the drop-down list box for the agreement, click **Edit Item**.

4.   On the **Agreement** template on the **General Information** tab, do the following:

Use this	To do this
Agreement Name	You cannot modify the agreement name. To change the name of an agreement, you must delete the agreement and then recreate it with the new name. For information about deleting an agreement, see How to Delete an Agreement . For information about creating an agreement, see How to Create an Agreement .
Description	Edit the agreement description.
Agreement Date	To modify the agreement date, click the calendar icon, and then click the date for the agreement.
Insert Duration	Click **Insert Duration** to expose the start date and an end date for the agreement. Type a start and end date, or click the calendar icon to pick the dates from the calendar.
My Profile	Click **Select Profile**. In the **Select My Profile** dialog box, select the self-profile you want to have this agreement with, and then click **OK**.
Partner Profile	Click **Select Profile**. In the **Select My Profile** dialog box, select the partner profile you want to have this agreement with, and then click **OK**.

5.   On the **Agreement** template on the **Addendums** tab, do the following:

Use this	To do this
Addendum Name	Type a name for the addendum.
Friendly Name (Optional)	Type an easy-to-remember name for the specified addendum.
Addendum Terms (Optional)	Add terms for your agreement, for example, for a purchase order "Void after 30 days" for addendum
Business Relationship Name	Type a Role Link name that implements the trading relationship electronically. Click and select the appropriate relationship deployed on your registered BizTalk Servers
Parameters	Define the value of parameter that you want to use for this partner and this relationship. (Parameters are shown only if the underlying business process defines them)

6.   On the **Legal Terms** tab, edit the terms in the **Legal Terms** box.

7.   Click **Submit Agreement**.

After you submit the agreement, a BAS business manager must activate the agreement for the changes to affect the business relationship. For information about activating an agreement, see How to Activate an Agreement .

How to Deactivate an Agreement

When you deactivate an agreement, it ends the trading relationship with that partner or group of partners. You must deactivate an agreement before you can undeploy an associated profile.

Prerequisites

Note the following requirements:

- You must log on as a member of one of the following Windows user groups:

  - BAS business managers

  - BAS business administrators

To deactivate an agreement

1.   In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site, and then press ENTER.

2.   On the **Business Activity Services Site Home** page, click **Agreements**.

3.   On the **Agreements** page, from the drop-down list box click **Deactivate Agreement**.

The agreement is no longer active. You can now modify the agreement. For information about modifying the agreement, see How to Modify an Agreement . If you want to resume the trading relationship, you must reactivate the agreement. For information about activating an agreement, see How to Activate an Agreement .

How to Delete an Agreement

You can only delete an inactive agreement. If an agreement is active, a BAS business manager must deactivate the agreement before you can delete it. For information about deactivating an agreement, see How to Deactivate an Agreement .

You can delete an inactive group agreement. When you delete a group agreement, you automatically delete all inherited agreements at the same time. You cannot delete an inherited agreement directly.

If a group agreement is inactive, you can remove an individual inherited agreement indirectly by removing the partner from the group from which the agreement is inherited. This automatically removes the inherited agreement. For information about removing a partner from a group, see <u>How to Add and Remove Partner Group Members</u> .

Prerequisites

Note the following requirement:

- You must log on as a member of one of the following Windows user groups:

  - BAS business users

  - BAS business managers

  - BAS business administrators

To delete an agreement

1.   In Internet Explorer, in the **Address** box, type the URL of the Business Activity Services site, and then press ENTER.

2.   On the **Business Activity Services Site Home** page, click **Agreements**.

3.   On the **Agreements** page, point to the agreement you want to delete, and from the drop-down list click **Delete Item**.

4.   When the **Are you sure you want to delete this item** message appears, click **OK**.

Using Business Activity Monitoring

Business Activity Monitoring (BAM) provides visibility into business processes independent of your IT environments and across a heterogeneous IT implementation. This distinguishes BAM from operational monitoring such as Health and Activity Tracking (HAT). A key aspect of BAM is being able to view business data in as close to real time as possible, referred to as zero latency

What Is BAM?

Business Activity Monitoring (BAM) is a collection of tools that allow you to manage aggregations, alerts, and profiles to monitor relevant business metrics (often called Key Performance Indicators, or KPIs). BAM provides accurate information about the status and results of various operations, processes, and transactions so you can address problem areas and resolve issues within the your business process.

The BAM Framework provides an easy, real-time, transaction-consistent way to monitor heterogeneous business applications, and to present data for SQL queries and aggregated reports (OLAP). Through queries and aggregations you can include

not only the data that is present during the running business process, but also the state and the dynamics of the running business process, independent of how the business is automated.

BAM applies operational business intelligence and application integration technologies to automated processes to continually refine them based on feedback that comes directly from knowledge of operational events. In addition to auditing business processes (and business process management systems), BAM can send event-driven alerts that can be used to alert decision makers to changes in the business that may require action.

Why Use BAM?

The typical enterprise today uses a variety of business applications, such as customer relationship management (CRM), SAP, and order management, purchased or developed internally over time. These applications frequently use disparate technologies and run as heterogeneous operating systems, ranging from COBOL programs using COM and COM+, to C# and Java.

At the same time, many aspects of a typical enterprise are based on human actions, such as phone calls, faxes, and e-mail. It becomes increasingly difficult to see "what is going on in the business" in such a complex environment. Nevertheless, with the increasing speed of the market, it is more crucial for enterprises to make quick decisions to take advantage of market opportunities or to prevent losses.

Business Activity Monitoring (BAM) can be used as a monitoring solution by IT managers who want to cut the cost of their distributed IT environments while improving service quality. BAM provides management oversight of the business-critical servers and applications across your company and offers a vertical view of high-performance workstations and applications within engineering, design, and production divisions.

BAM exposes the visibility requirements of the end-to-end business process. Using an understanding of how the various roles within the business interact with the business process as well as an understanding of the data requirements for the interactions, the business analyst interacts with the BAM design-time surface in Microsoft Office Excel 2003 through the BAM wizards to create activity and view definitions to support these visibility needs of the various roles.

BAM drives interaction through a rich KPI and alerting experience, simplifies creation of end-to-end business process visibility by the business analyst and consumption of the data by the information worker, improves deployment and management of dynamic business tracking environment, and provides broader support for mapping of visibility to process, more support for other business activity sources.

More specifically, BAM addresses challenges facing enterprises as follows:

- Business end user empowerment through the BAM portal

- Business alerts and notifications

- Better visibility creation and consumption experience

- End-to-end visibility of the business process

- Support for pipelines (and, indirectly, adapters and Web services)

- Support for process visibility as MOM role for administrators

- Immediate Activity change-on-the-fly

- Interceptor and tracking profile guidelines

About BAM Activities

The BAM Activity represents a unit of work in the business, such as Purchase Order or Loan Application. The intention of the Activity is to show the history (milestones) and data about this unit of work to the end business users (Information Workers). Thus, the BAM Activity is high-level abstraction that is independent of the actual implementation of your IT infrastructure.

Your job as a developer is to maintain this abstraction by exposing only the relevant milestones and data from the implementation in the context of a specific activity. For a code sample that demonstrates the use of an activity, see BAM API (BizTalk Server Sample).

In This Section

- Using an Activity

- Activity Relationships

- Activity Continuation

- Looping Activities

Using an Activity

The simplest way to use BAM is to send explicit milestones or data, using the BAM API. You can think of the BAM Activity as a record in a SQL table that you are keeping in synchronization with the actual unit of work.

- Call **BeginActivity** for each new unit of work.

- Call **EndActivity** when the work is complete and you expect no more events in the context of this unit of work.

The following code shows how to do use BeginActivity, UpdateActivity, and EndActivity when the unit of work is a Purchase Order. In the example, we assume that the string variable **poid** identifies the current Purchase Order in process:

Activity Relationships

An activity relationship exists when an activity relates to one or more other activities. An example of this is having multiple "Shipment" activities related to a single "Purchase Order" activity, or one "Shipment" containing items from two "Purchase Orders".

To indicate that two activities are related, you need to know both names and have the corresponding ActivityIDs in-memory in order to call AddRelatedActivity. This API creates the link between the corresponding activity records.

In the following diagram, the highlighted lines of code shows how you make a relationship between purchase order activity instance #123, and shipment #1549, 1550, and 1551.

Activity Relationship Diagram

Relationships Code Example

```
string poid="PO#123";
es.BeginActivity("PurchaseOrder",poid);
...
es.UpdateActivity("PurchaseOrder",poid,
     "POReceived",DateTime.UtcNow,
     "POAmount",100,
     "CustomerName","Joe",
     "CustomerCity","Seattle");

...
es.AddRelatedActivity("PurchaseOrder",poid
                    "Shipment","1549");
es.AddRelatedActivity("PurchaseOrder",poid
                    "Shipment",1550");
es.AddRelatedActivity("PurchaseOrder",poid
                    "Shipment",1551");

es.EndActivity("PurchaseOrder",poid);
```
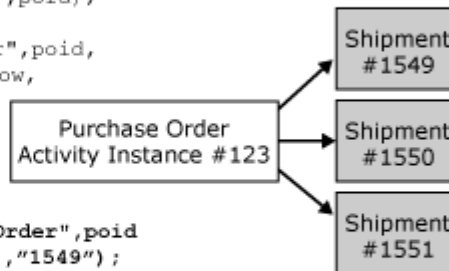
The business user looks at one Web page that shows the history of a purchase order. It may indicate that at 10 A.M. it arrives, two days later it receives approval, and the page provides a link to the actual documents. Because of the code above, the page will also provide hyperlinks that take the business user to the corresponding shipment Web pages.

Activity Continuation

The Business Activity can span across multiple heterogeneous applications (for example, a pipeline, two orchestrations, a line-of-business application, and then another pipeline). The BAM infrastructure can correlate the events from multiple applications, with a little help from the developer – a concept called "Continuation":

BAM Continuation Tokens
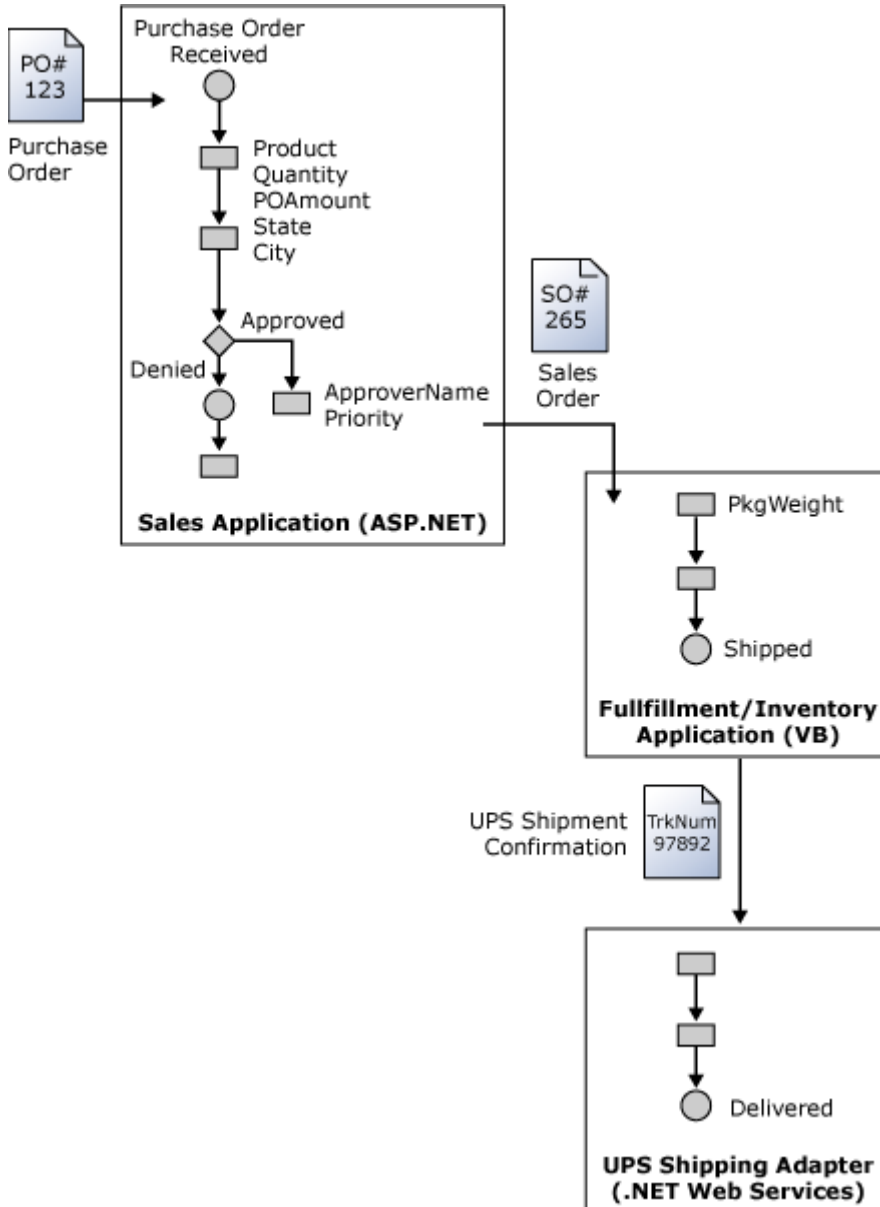
The first part of the activity happens in the Sales application, the second part of the activity happens in the Packaging & Assembly application, and finally, the delivery progress is available in the Shipping application. Each application uses different IDs for the current work unit: purchase order number (PO), sales order number (SO), and shipping order number (UPS). To correlate the events between two different applications, you must:

- Identify the Continuation Token – a unique piece of data that is available to both applications (e.g. part of the message being exchanged).

- Call EnableContinuation in the first application and pass the Continuation Token along with the current ActivityID.

- Do not call BeginActivity in the second application.

- Fire all subsequent events in the second application by using the Continuation Token instead of ActivityID.

The following code example illustrates the use of Activity Continuation among three applications:

Purchase Order Application

Fulfillment Application

Shipping Application

Use the following guidelines regarding activity continuation in your code:

- Only use continuation when the end user must treat the work of different applications as part of the same activity. Use separate activities for each application and create an activity relationship if the end user views the work in each applications as meaningful activities.

- If the work units in the applications do not have a one-to-one relationship, you can use activity relationships but not continuation, for example when multiple shipments exist for a sales order.

- If you send data to BAM synchronously (using DirectEventStream) and the ActivityID is propagated to all involved components, then you do not need to use continuation.

- If you send data to BAM asynchronously (using BufferedEventStream or from orchestrations), then you must use continuation even if the ActivityID is propagated to all components. In this case, you need to use a different ActivityID in each application by prefixing it with unique string (e.g. Application Name). This is necessary because the data from different applications may arrive to BAM in random order and BAM has to hide the out-of-order events to ensure correct query and aggregation results.

- Continuation does not require rewriting your applications to exchange more data.

Looping Activities

Looping activities refers to actions that loop within an orchestration. It is possible to capture the events from actions that loop within an orchestration. To do this, you create another activity and map all of the new activity milestones and data inside the loop. This is necessary because the data processing in the loop will occur more than

once per schedule execution. The following diagram shows an example of this situation.

Looping Activities

Handling Loops and Repeating Data
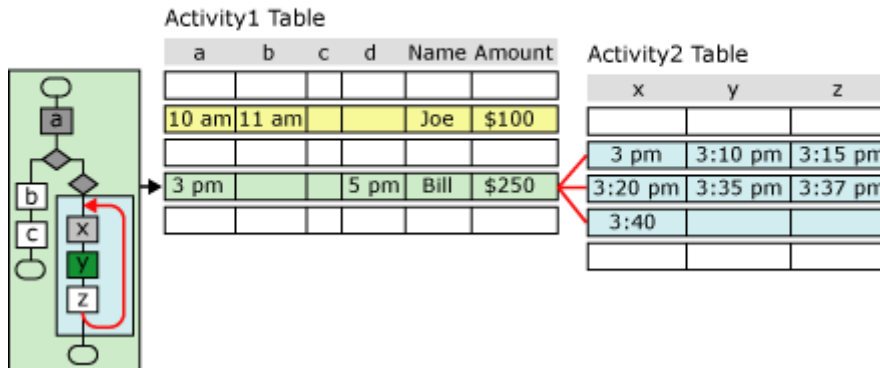
As shown in the diagram, if you have a Purchase Order with multiple Line Items that process in a loop, questions like Which purchase orders have item prices of $100? are ambiguous. The correct questions are:

- Which purchase orders have line items with a price of $100?

- Which purchase orders have Total/Min/Max item prices of $100?

This requires thinking of the Line Items as something separate from the purchase order. In the Tracking Profile Editor, the root activity (purchase order for example) maps to all actions outside the loop. The child Activity (line item for example) maps to the actions inside the loop.

You need to use a payload item as ActivityID for the root activity. Have this payload item available in some of the messages inside the loop. Map the activity to the Relationship Node that displays under the Child Activity and name it as the root activity.

Considerations for BAM Applications

This section describes various performance and code maintenance considerations for developing applications that use BAM.

In This Section

- Performance Considerations for BAM Event Publishing

- BAM Code Maintenance Considerations

Performance Considerations for BAM Event Publishing

BAM supports two forms of business event publishing:

- Synchronous

- Asynchronous

The following diagram illustrates the two models.

BAM Topologies
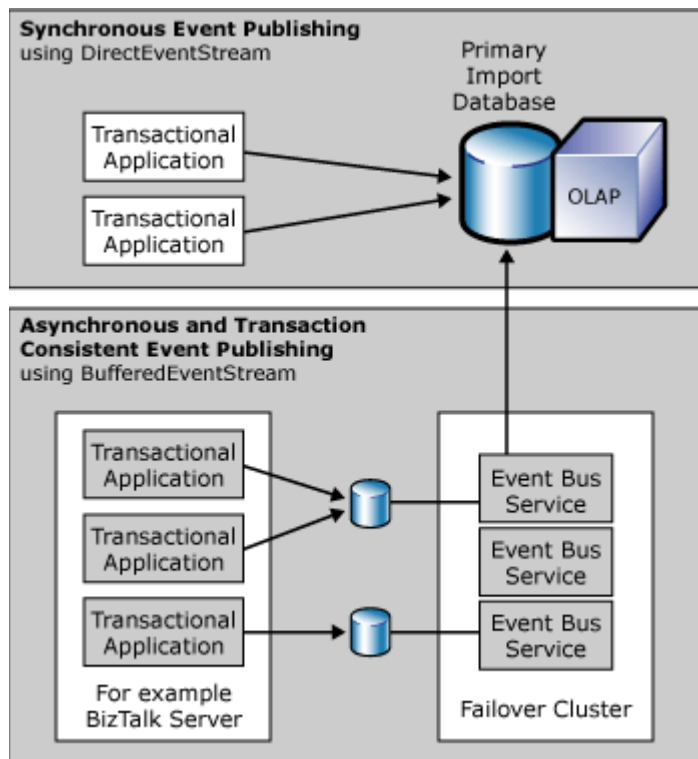
The synchronous approach is much simpler for management and using from code, while the asynchronous approach allows for better performance.

This section contains:

- Synchronous Business Event Tracking

- Asynchronous Business Event Tracking

Synchronous Business Event Tracking

The simplest way to send event data to BAM is to use an instance of the class **DirectEventStream**. This class saves the event data directly into the BAM Primary

Import Database in the context of the current transaction of the application (if present).

If any error happens during this operation, the method call will throw an exception back in the calling application. This will happen for example if the name of an item passed in UpdateActivity mismatches the BAM Activity Definition, or you did not deploy the BAM Definition yet. This allows the calling application to catch and recover from any errors when saving the BAM data, which results in much easier management later.

Saving the data synchronously might have a performance impact, because the calling application has to wait until BAM executes all stored procedures and triggers.

Asynchronous Business Event Tracking

Asynchronous (using **BufferedEventStream**) - This model offers significant performance improvements. This uses a similar API to the synchronous model, using only a different constructor. Instead of pushing the data into the primary import database, BufferedEventStream accumulates the event data in memory in binary form, and then inserts it as a single table record into an interim database (MessageBox). The Event Bus service reads the data queued in the MessageBox database by BizTalk and imports it into the primary import database.

To configure BAM for asynchronous operation, the Event Bus Service and the calling application (e.g. Orchestration Host) should run on different computers. This allows the calling application to get rid of the event data immediately to be processed using the CPU on a different computer.

This BAM topology is also transaction-consistent. You will never get BAM data for transactions that rolled back in the calling application, because the Event Bus Service only reads the committed data from the MessageBox database. The BAM data for the transactions that were committed will show up for query and aggregation with small latency.

If errors occur, the Event Bus Service will retry a few times, use timeouts, crash recovery logic, and so on. If however the data is in an invalid format, it ends up in special tables. An event occurs in the event log to indicate this condition. This additional failure point increases the difficulty of managing the system.

Using the asynchronous approach from code is usually as simple as replacing the DirectEventStream with BufferedEventStream, and passing the connection string to the Message Box in the constructor, instead of the one for BAM Primary Import.

Use the following guidelines regarding asynchronous business event tracking in your code:

- Always use continuation when the Activity spans multiple applications and you send the data asynchronously to BAM, even if you propagate the ActivityID to all components. In this case, use a different ActivityID in each application by prefixing it with a unique string (e.g. Application Name). This is necessary

because the data from different applications might arrive to BAM in random order, and BAM has to manage the out-of-order events to ensure correct Query and Aggregation results.

- The usual methods of event monitoring (e.g. COM+ Loosely Coupled Events or WMI Events) are not applicable for BAM because the event data is independent of the transaction. For example, it may look like you received 10 purchase orders for a total of $10,000 while there was only one incoming purchase order for $1000, but the attempt to save it to the database failed 10 times.

BAM Code Maintenance Considerations

An important consideration to bear in mind is the likelihood of changing requirements about what data is sent to BAM. If the requirements change, and you instrument your application with hard-coded calls to the EventStream classes, you will need to take it offline, change the code, recompile, and put back in production.

An alternative approach is to use the BAMInterceptor class, which allows one-time instrumentation of the code, and then change the format of the events through metadata.

In This Section

- Hard Code BAM Event Format

- One-Time Instrumentation

Hard Code BAM Event Format

This approach is simpler and generally applicable when you are building dedicated applications with specific well-known BAM requirements.

Before deciding to use this approach, you need to have answers to the following questions:

- Where are the BAM Milestones in the code?

- What is the data of interest, and when and where it is available?

If anything is likely to change (such as more Milestones or different data of interest), then you should also consider one-time instrumentation with the BAMInterceptor.

The hard-coded approach means that you simply call the DirectEventStream or BufferedEventStream, depending on your performance requirements.

One-Time Instrumentation

This approach is better when:

- You have to deal with tradeoff between Visibility and Performance, and you want to be able to have control at runtime.

- The application deals with big XML messages, in which any data may become important for monitoring at some point.

- It is unacceptable to stop the business and change the code to extract different data for BAM

In this approach, you instrument the application in a generic way using the BAMInterceptor. By passing different configurations to the Interceptor, you can change what data you actually send to BAM.

The BizTalk Orchestration Engine accommodates interception, which allows changing the BAM-collected data at runtime using the Tracking Profile Editor.

Using the BAM Interceptor

The BAM Interceptor is an object that lets you instrument your application to capture data of interest. The following diagram shows the role of the BAM interceptor and its interaction with the other BAM components:

BAM Interceptor
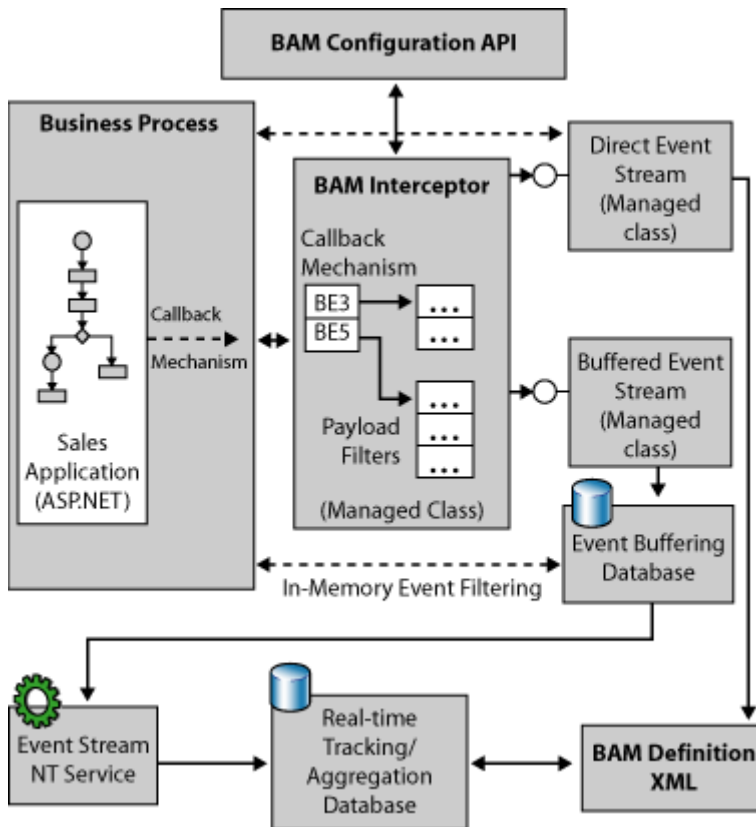
In each step of your application where you could have data of interest, you call Interceptor OnStep, provide an identifier for the step, and provide some data or arbitrary object that you are using in your application.

You must implement a callback function so when the callback occurs, your callback procedure gets the current step ID and your data object. Essentially, the BAM interceptor is simply propagating the data object to the callback. The actual logic of extracting data resides in your application. For example, if your data takes the form of XML messages, then the callback will use XPaths. For more information about XPaths, see Using XPaths in Message Assignment.

The BAM interceptor decides which data to request at each step, based on the configuration that you can create programmatically. The BAM Interceptor then uses the obtained data to call either DirectEventStream or BufferedEventStream that you need to keep around and pass each time as an argument to OnStep.

Calling the interceptor for each step is not a resource-intensive operation. If you call and you register nothing for this step, the interceptor returns immediately. This means that there are no disk operations, no transactions, not even memory allocations, and thus almost no performance impact. At the same time, you have the opportunity to extract any data for BAM if necessary. The performance impact on the steps involving data extraction and the availability of the data will depend on your implementation of **IBAMDataExtractor Interface**.

The following code examples demonstrate the use of the interceptor during configuration and run time.

Configuration time

The following code shows how you configure the Interceptor to stop at step recvPO of the application, and ask for Customer Name and Customer SSN:

After you create an interceptor instance, you can store it for later use at runtime.

You may keep different pre-created interceptors representing different preferences for the data and milestones for BAM. For best performance, serialize the Interceptor instances using the BinaryFormatter class.

Run time

Use this code to use the interceptor at runtime in a production environment:

Where:

- *recvPO* and *approvePO* are arbitrary objects you use to identify the steps in your application.

- *data1* and *data2* are arbitrary objects that you have at that point and may contain interesting data – for example the XML document of the purchase order.

- *es* is either DirectEventStream or BufferedEvent stream depending on your performance requirements.

- *callback* is your implementation of the **IBAMDataExtractor Interface**.

The SDK sample, BAM API (BizTalk Server Sample), demonstrates using the Interceptor, which contains both a configuration tool and example runtime application.

The BizTalk Orchestration Engine accommodates interception, which allows changing what data is collected for BAM at runtime using the Tracking Profile Editor.

BAM Dynamic Infrastructure

Once you extract the data of interest from your applications, you should store it so that it is available for queries. Additionally, you may need to maintain certain pre-created aggregations of the data for faster aggregated queries.

Typically, you achieve such functionality by a heavy development effort to implement a data warehouse. BAM in Microsoft® BizTalk® Server 2006 greatly simplifies this process by automatically generating the SQL and OLAP infrastructure based on your activity and view definitions.

In This Section

- Activity Data Storage

- Aggregations

- Querying BAM Data

- BAM Definition Schema

- BAM Infrastructure Limitations

What Is a BAM Definition Schema?

The BAM definition schema defines the structure of the observation model created by the business analyst.

The schema defines the structure of the BAM definition XML document including the document's elements and sub elements. For example, the root element is the BAM definition, and inside the BAM definition there are the following elements:

- Activities

- Views

- Cubes

The schema defines the document constraints. The constraints dictate that some elements can only reference certain other elements for example.

You can create the BAM definitions (instances of this schema) with any text editor or by using the BAM.xls template spreadsheet available in the *<drive>*:\Program Files\Microsoft BizTalk Server 2006\Tracking folder. This spreadsheet contains macros that enable you to define BAM activities and views, then the spreadsheet will generate the BAM definition XML for you automatically. Once you define the BAM activities and views, you do not have to manually export the XML file, since BAM Manager can read this spreadsheet and get the XML from it to deploy your infrastructure automatically. The result is another copy of the spreadsheet named BAM_Livedata.xls. You can use this new spreadsheet to connect to your live data source and view the aggregations in PivotTable® reports.

Activity Data Storage

The basic idea of activity data storage is to have a separate table for each activity type, in which each record represents a different activity instance (in progress or completed for example). For example, if the activity were Purchase Order, the table would look like the following:

PO#	RecvTime	City	Quantity	ShipTime	DeliveryTime
123	8:00am	Seattle	150	8:24am	12:45pm
124	8:30am	Seattle	234	8:45am	1:20pm
125	8:35am	Redmond	87	9:05am	2:30pm
126	8:45am	Seattle	450	9:20am	3:10pm
127	8:55am	Redmond	200	9:30am	<NULL>
128	8:57am	Seattle	340	9:20am	3:05pm
129	9:12am	Seattle	120	9:45am	<NULL>
130	9:30am	Redmond	25	10:15am	<NULL>
131	9:45	Seattle	250	10:35am	<NULL>
132	10:00am	Redmond	100	<NULL>	<NULL>
133	10:15am	Seattle	230	<NULL>	<NULL>
134	10:25am	Redmond	45	<NULL>	<NULL>

In this table, when BAM receives a new purchase order, it inserts a new row and some sets some of the columns to non-null values (RecvTime, City, Quantity, and so on). Later, when you approve and ship this purchase order, BAM sets ShipTime to a non-null value. Finally, when you receive and confirm the shipment, BAM sets DeliveryTime to a non-null value.

Unfortunately, the performance of this simplistic implementation quickly degrades over time. In the beginning, the performance is limited by the number of the transactions the SQL server can perform (essentially CPU bound), but after some time, it drops drastically. At the same time, the average queue length for disk IO increases beyond the acceptable limits:

BAM Write Performance versus Disk Queue Length
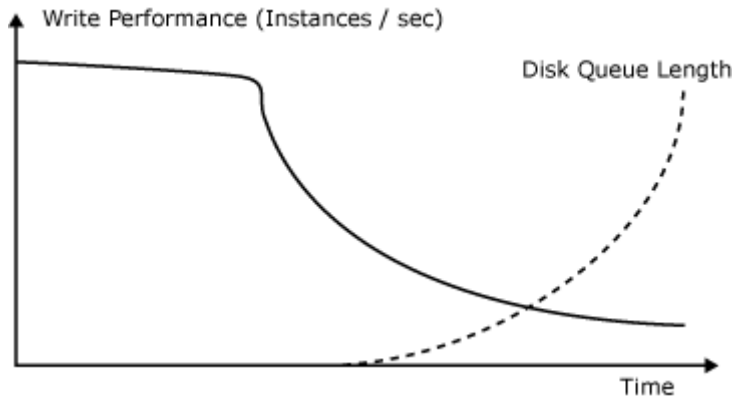
The reason for this is that the size of the table grows as more instances of the business process complete. For example, the first time, the UPDATE statement of the stored procedure causes a search on the clustered index for purchase order number and reads some pages in memory. Since the instances of the purchase order process are independent (some take a long time, but some are short), the next call to the stored procedure may be for some other purchase order instance and therefore will require reading of different data pages in memory. As long as the total number of purchase order records is small, Microsoft SQL Server™ will cache all data pages in memory. When the number of the records grows large enough, the cache-hit ratio decreases and each operation requires a physical disk read. Apparently, in this situation no query activity against the table is possible.

To avoid this problem BAM uses two separate tables – one for the Activities still in progress, and another for the completed ones in the following diagram:

BAM Tables
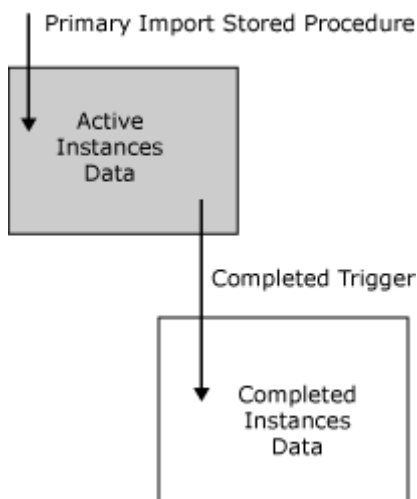
In this diagram, the idea is to keep a relatively small table that updates occur in and another that grows large, but is incrementally accessed (INSERTs only). In the

example, only the orders being processed at the moment will be in the active table, while all orders that were already delivered will go to the completed table.

Because of the trigger, this structure of tables is slower than an INSERT/UPDATE of a single table at the beginning, but maintains stable write performance over time.

Online Window for Activity Data

The Activity Storage primarily handles queries for current or recently completed activities. BAM archives, then purges very old, completed activities from the Primary Import Database. Thus the activity data flows through BAM and is available for queries during a configurable Online Window.

To allow for higher performance and avoid downtime, the Activity Storage uses partitioning based on the timestamp when the activity was completed. BAM achieves this by regular swapping of the completed table with another empty table of exactly the same format. Once BAM does this, the further completed activities go into the new table, while BAM keeps the old one only for queries, as in the following diagram:

BAM Partition Swapping
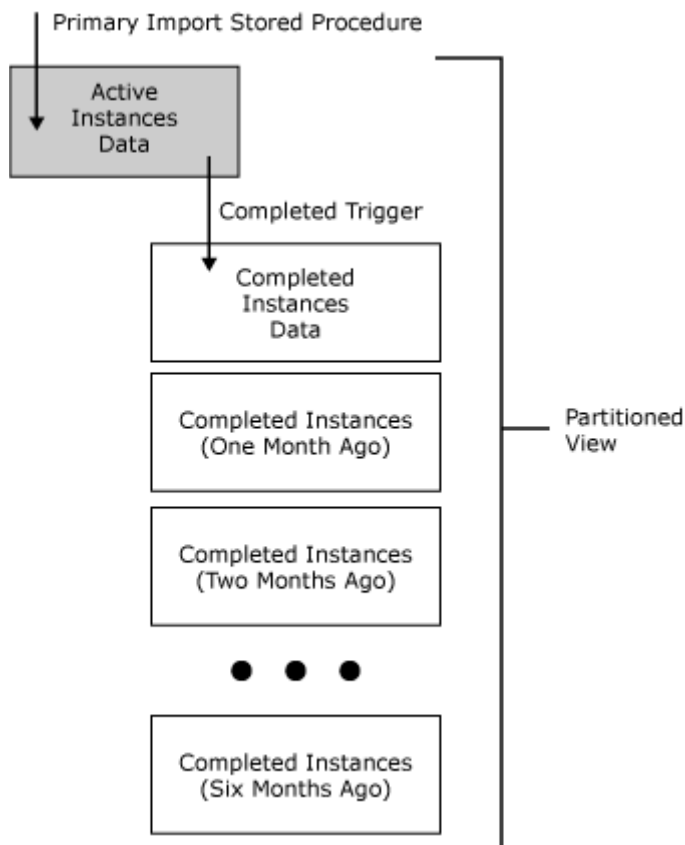
Once a partition is completely outside the online window, BAM archives and then drops it. To hide this complexity from the user, BAM also maintains a partitioned view of the form:

BAM automatically recreates this view each time it creates or drops a partition.

Note the following about BAM partitioning:

- The name of the partitioned view is **bam_<ActivityName>_AllInstances**. This view is not meant for direct queries, but may be useful when troubleshooting the BAM instrumentation. You should query the data from the specific views for each category of business users that you create on top of this view (see <u>Querying Instance Data</u>).

- You set the online window by modifying the values for **OnlineWindowTimeUnit** and **OnlineWindowLength** in the record for the current activity in the table **bam_Metadata_Activities** in the Primary Import database.

- The DTS package, **BAM_DM_<ActivityName>**, performs the partitioning and archiving/purging. Each time this package runs, it truncates another partition and archives/drops all partitions that are outside the online window.

- If you do not have Archiving Database configured, then BAM drops the aged activity data without archiving.

Aggregations

Excel defines aggregations as pre-calculated summaries of data that improve query response time by having the answers ready before the questions are asked. For example, when a data warehouse fact table contains hundreds of thousands of rows, a query requesting the shipping schedules for two particular products can take a long time to answer if the fact table has to be scanned to compute the answer. However, the response can be almost immediate if the summarization data to answer this query has been pre-calculated.

The following figure displays an example of pre-calculated aggregation data.

BAM OLAP Cube
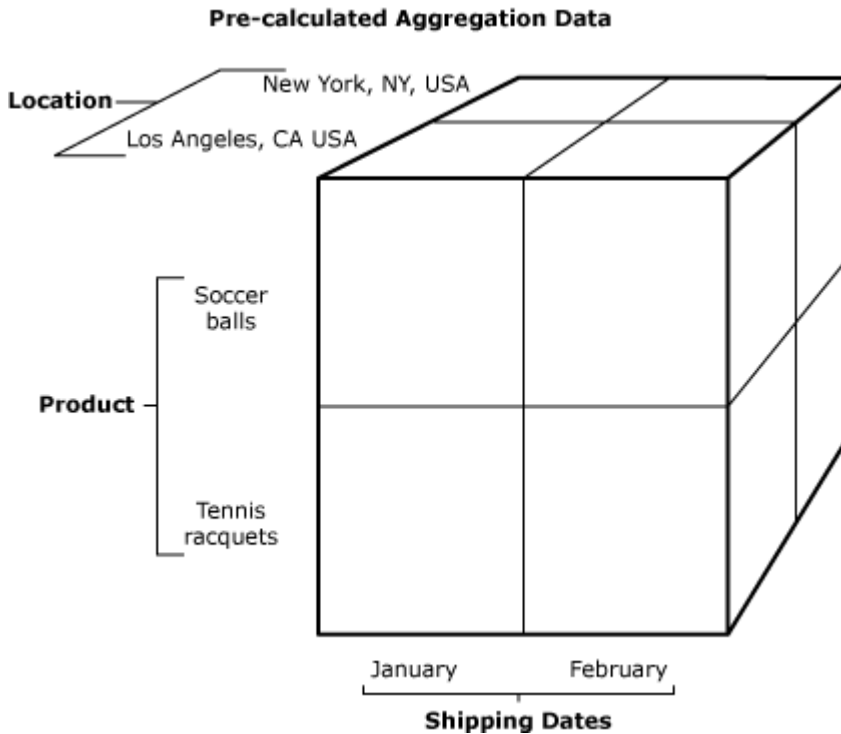
Pre-calculated Aggregation Data

The above figure summarizes the numbers of each product, shipped to specific locations over a two-month time period. Excel typically defines this data as measure. The data used for filtering and categorization, Excel defines as dimension.

For the user experience of browsing multidimensional data, see the Pivot table topic in Excel Help.

You can create multi-dimensional activity aggregations based on the data available in a specific view. Those aggregations contain measures (data to aggregate, such as dollar amount) and dimensions (used for filtering or grouping, such as State and City).

You can create the aggregations in the form of OLAP cubes, which represent a snapshot of the business at a specific time, or as real-time aggregations, which the business scenario determines and you see them via the multi-dimensional structure in real time.

This section contains:

- Scheduled Aggregations

- Real-Time Aggregations

Scheduled Aggregations

BAM bases scheduled aggregations on dynamically generated OLAP cubes and DTS packages. The data in scheduled aggregations represents a snapshot of your business activities when you start your DTS package. To achieve this, the first step of the DTS package for analysis is a call to the stored procedure **bam_Metadata_BeginAnalysis** that will retrieve a snapshot consisting of:

- A snapshot copy of all activity instances in progress

- A view that represents an incremental window on the completed activity instances from the moment that you ran the DTS Package for the last time to the moment of the snapshot

BAM achieves this by taking an exclusive lock on the Activity Storage for a very short time, thus preventing any data writing at the same time. Once BAM takes the snapshot, the DTS package might take a long time to run, but BAM will ignore any new data that arrives during the processing. The following diagram illustrates this activity:

BAM Scheduled Aggregations
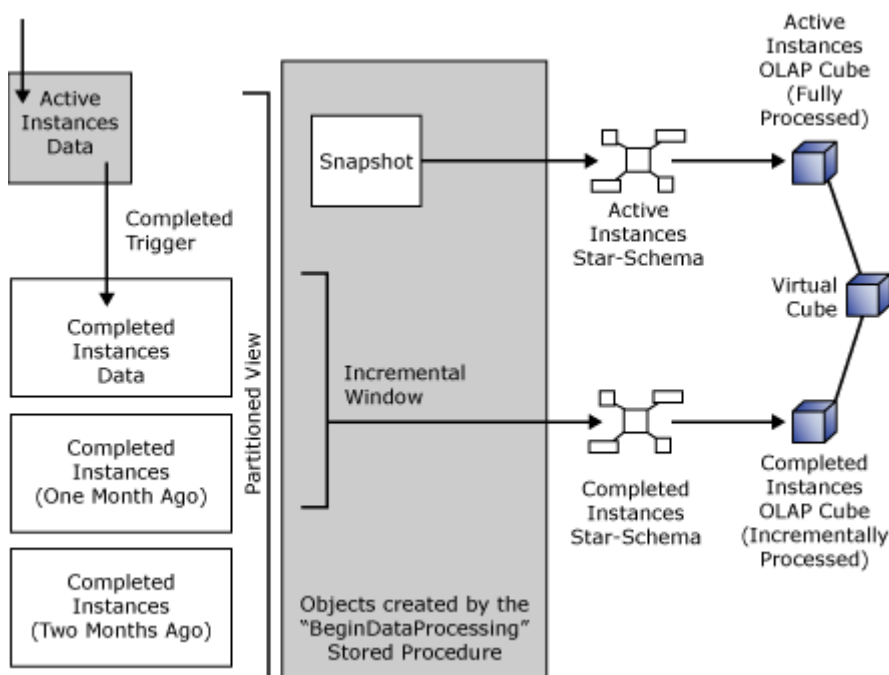
In the diagram, BAM moves data about the completed activity instances to the Completed Instances OLAP cube. BAM incrementally processes this cube.

At the same time, BAM moves the data about the activities still in progress to the Active Instances cube, which the DTS package fully processes. This is acceptable, because BAM assumes that only a relatively small number of activities are in progress at any given moment.

Data for the scheduled aggregations is available from a virtual cube that hides the difference between the completed and current activities. For more information, see Querying Scheduled Aggregated Data.

Real-Time Aggregations

In some cases, specific slices of the multi-dimensional aggregations are so time-sensitive that you want them to be available in real time. For example, your business is selling perishable products and you want the aggregation of product quantity in different stages of delivery to be available in real time. At the same time, you want other aggregations such as the age of your typical customers, but only at the end of the month for business intelligence analysis.

BAM implements Real Time Aggregation (RTA) as a table maintained by triggers from the Activity Storage tables. In the case when your business deals with purchase orders,(PO) the RTA view may look like the example in the following diagram.

BAM Real-time Aggregations

In this diagram, if a new PO of $100 from Redmond is received, BAM adds a contribution to the cells in the corresponding row for {Redmond, In Progress} by performing an operation like `Count=Count+1` and `Amount=Amount+$100.`

Later, if the same order ships, then BAM removes this contribution from the row {Redmond, In Progress} and adds it to the row, {Redmond, Shipped}.

BAM maintains the data inside RTA for a given online window, and then deletes it. You can configure the online window by changing the corresponding row of the table **bam_Metadata_RealTimeAggregations**.

The following statements also apply to real time aggregations:

Real time aggregations significantly affect the speed by which BAM can write data. Thus, you should only define the most important slices of the aggregation structure as RTA.

The limitation of the dimension levels for real time aggregations is 14. If for example you create a Data Dimension Location for State and City, this counts as two levels (State and City.). For Progress Dimensions the number of levels is depth of the tree,

and for Time Dimensions it is the count of all sub-units. For example, a Time Dimension for Year, Month, Day, Hour will count as 4 levels.

BAM does not support aggregations of type **Min** and **Max**. The aggregations that BAM supports are **Count**, **Sum** and **Average**.

You must always create a time dimension for RTA and always use it in all data slices, because the data in RTA is aging based on the server timestamp, and not on any specific business milestone.

Querying BAM Data

This section describes the way you query BAM data, including instance and aggregated data.

This section contains:

- Querying Instance Data

- Querying Real-Time Aggregated Data

- Querying Scheduled Aggregated Data

- Querying Activity Relationships

Querying Instance Data

The data about individual activity instances is available for queries in a dynamically created SQL View in the BAM Primary Import Database.

The name of this view is

bam_<*ViewName*>_<*ActivityName*>_View

Where

<ViewName> is the Name Attribute of the View element in the BAM Definition XML, which is the same as the View Name entered in the related Microsoft Excel Wizards.

<ActivityName> is the Name Attribute of the Activity element in the BAM Definition XML, which is the same as the Activity Name entered in the Excel Wizards.

It is important to note the following conditions when querying instance data:

- If you send activity data to BAM via the DirectEventStream, the instance data has no latency, meaning that it appears instantaneously when the transaction in the calling application commits.

- If the activity data is send to BAM via the BufferedEventStream, the instance data will show up for queries a few seconds later, depending on the load of the BAM Event Bus Service and the SQL Server that hosts the BAM Primary Import Database.

- The actual structure of tables behind this view is more complex to ensure that the data for the current or recent activities is available for queries, while the data for the activities that are completed and aged out is archived or purged without taking the system offline. For more information, see <u>Activity Data Storage</u>.

- Out-of-the-box Web Experience for Queries for this data is available if you install Business Activity Services.

Querying Real-Time Aggregated Data

The real-time aggregation data is available for queries in a dynamically created SQL View in the BAM Primary Import Database.

The name of this view is

bam_<*ViewName*>_<*RTAName*>_RTAView

Where

<*ViewName*> is the Name attribute of the View element in the BAM Definition XML, which is the same as the View Name entered in the related Microsoft Excel Wizards.

<*RTAName*> is the Name attribute of the RealTimeAggregation element in the BAM Definition XML, which BAM generates to be unique based on the view name

It is important to note the following conditions when querying real-time aggregated data:

- The real-time aggregations must be configured to keep the aggregations for given amount of time (default 1 day) and never grow very big. The older aggregations should be available in the OLAP cubes instead.

- Any query against RTA must include filtering on a time dimension that will be inside the online window for the RTA data. This is necessary because BAM bases the data maintenance for RTA on the server time the data gets into BAM and is optimized to drop the data in chunks. Thus if you simply send the Transact-SQL command "select *", the results will fluctuate in unpredictable way.

- If the activity data is end to BAM via the DirectEventStream, the real-time aggregated data has no latency – it appears instantaneously when the transaction in the calling Application commits.

- If the activity data is sent to BAM via the BufferedEventStream, the RTA data will show up for queries a few seconds later, depending on the load of the BAM

Event Bus Service(s) and the SQL Server that hosts the BAM Primary Import Database.

- BAM bases the real-time aggregation on a table that it maintains in synchronization with the changes or insertions in the <u>Activity Data Storage</u> records via triggers. Thus, the real-time aggregation can have a significant performance impact. For more information, see <u>Real-Time Aggregations</u>

- Out-of-the-box Web Experience of Querying the RTA is available if you deploy BAM Definition Workbook instead of XML (see BAM Roles)

Querying Scheduled Aggregated Data

The scheduled aggregation data is available for queries in a dynamically created OLAP Cube BAM analysis database.

The name of this cube is the same as the Name attribute of the View element in the BAM Definition XML, which is the same as the view name entered in the Excel wizards. BAM refreshes this cube when you run the DTS Package BAM_AN_*<ViewName>* where the *<ViewName>* is the name of the view described previously.

It is important to note the following conditions when querying scheduled aggregated data:

- This is a virtual cube containing a snapshot of the business at the exact moment that the DTS Package execution started. It contains aggregations both for the completed activities and for the ones still in progress. You can use the progress dimension to filter or group the aggregations accordingly.

- If you are never interested in the current activities (e.g. if they complete very fast) you can query the corresponding real cube *<ViewName>*#Completed.

- If you also have real-time aggregations, schedule the DTS Package for refreshing the cube more frequently than the online window you set for the real-time aggregations. Otherwise, there will be a window of time for which you do not have aggregations.

- Out-of-the-box Web Experience of Querying the RTA is available if you deploy BAM Definition Workbook instead of XML (see BAM Roles)

Querying Activity Relationships

activity relationship information is available in a dynamically created SQL view for each activity. The name of this view is

bam_<*Activity*>_AllRelationships

Where *<Activity>* is the Name attribute of the Activity element in the BAM Definition XML, which is the same as the Activity name entered in the Excel Workbook.

The relationship events occur in the context of a specific activity. If for example the relationship between Purchase Order and Shipment occurs in the context of the Purchase Order activity (see Activity Relationships) the Relationship record will show up in **bam_PurchaseOrder_AllRelationships** but not in **bam_Shipment_AllRelationships**.

To find all the related activities to a purchase order you need to query both the view **bam_PurchaseOrder_AllRelationships** as well as all views **bam_<OtherActivity>_AllRelationships** where <OtherActivity> is the activity in the same BAM view.

The relationship records are part of the activity instance and they are maintained in synchronization with the instance data as described in Activity Data Storage.

BAM Infrastructure Limitations

The BAM infrastructure has the following design limitations for this release of BizTalk Server 2006:

1.   Real time aggregations (RTA) do not support the MIN or MAX function.

2.   You cannot reference a one-level alias by more than one data dimension in one RTA view.

3.   Neither regular cube nor RTA supports distinct count measures. BAM Manager creates a default count measure for both regular cube and RTA. The default count is the total count of facts (also known as "activity instances"), not distinct count.

4.   BizTalk limitation - checkpoints of **DateTime** type are always identified as business milestones. When BizTalk Server collects business milestones, BizTalk places the timestamp on the message at the tracking point. For data items such as "date of birth", it cannot collect it as a business milestone and hence it is not queryable.

BAM Maximum Number of Objects

The following table lists the maximum number of objects in the BAM infrastructure.

Object	Implementation Limit	Reason
Checkpoints in Activity	1000	SQL supports 1024 SP arguments and we have 3 system ones. Place reserved for extensibility
Indexes in Activity	248	SQL supports 249 non-clustered indexes per table. We always create an index on ActivityID

DataLength	4000 Unicode characters	SQL variables of type NVARCHAR are used
ActivityRef	128	You can obtain the view by joining the PI and Relationship tables (two per Activity). SQL supports 256 tables in SELECT statement
Columns in View (Aliases, durations, plus dimension levels)	1000	SQL Limit of 1024 columns on the Staging Table, and there are two system columns
Dimensions in OLAP cube	128	OLAP limitation - 128 dimensions per Cube
Measures in OLAP	1024	OLAP limitation - 1024 measures per Cube. And the Count measure is created always
Dimension Levels in OLAP	64	OLAP limitation with an additional limit of 256 levels per cube
Dimensions levels in RTA view	14 dimension levels	BAM creates an index on all dimension levels, a SQL index can be created on up to 16 columns, and BAM reserves two for system columns
Measures, hidden measures (default count, hidden SUM measures for AVG) and dimension levels in RTA view	1024	Maximum 1024 columns in SQL table/view

BAM Object Names

The following table lists the limitations of object names in the BAM Definition schema and Microsoft Excel spreadsheet.

Object Names	XSD Limitation	Excel Limitation
Checkpoint name	100 characters	100 characters
Index name	100 characters	100 characters
Duration name	100 characters	100 characters
Alias name	50 characters	50 characters
Activity Name	20 characters	20 characters

ViewName	48 characters	18 characters
ActivityView name	48 characters	N/A (derived from Activity name)
Cube Name	20 characters	N/A (derived from View name)
RealTimeAggregation name	48 characters	N/A (derived from Cube name)
Dimension Name	20 characters	20 characters
Level Name	20 characters	20 characters
Measure Name	20 characters	20 characters

Using BAM with Custom Analysis Tasks

The easiest way to create a custom DTS task for processing BAM data is to start from the package that is auto-generated by BAM and replace all the actual data processing.

To create a custom DTS task

1. Create a BAM Definition that requires an OLAP cube. For example use the Excel wizards, and leave one PivotTable® report as non-RTA view.

2. Open the DTS package for cube processing that BAM creates. BAM creates one such package for each view, known as BAM_AN_<*View Name*>.

3. Open the Package in the DTS Designer and remove all steps except the first two steps and the last step. You may also want to keep the connection to the Primary Import database.

4. Edit the properties of the first ActiveX® task. The script begins with:

Remove all lines that contain DTSGlobalVariables.Parent.Steps, because they refer to the deleted steps.

Note The task "Begin Data Analysis" (the second task in the package) is very important because it gives your package:

- A moving window for incremental processing of the completed Activities (the dynamic SQL view named bam_(BamView)_View(Activity)_CompletedInstancesWindow

- A Snapshot of the Activities that are in progress - a table named bam_(BamView)_View(Activity)_ActiveInstancesSnapshot.

1. Obtain the view and table in a short transaction, during which you insert no data, so that the data represents a real instantaneous snapshot of the Primary

Import database. Implement one or more steps to do the actual data transformations based on the view and table as input data. If the purpose of your analysis task is something other than filling an OLAP cube, remember to keep a timestamp of when your job committed for the last time and replace the first ActiveX task with code that assigns this timestamp to the global variable "CompletedCubeLastProcessTime". The second task uses this variable to make sure that there is no missed data and that no data processes twice in case of a crash and restart of the DTS package.

2.   Finally, you must call the last task, which is "end data analysis". This task releases the completed activities that were processed, so that they can be archived and removed from the primary import once they are outside of the online window.

Using Lookups to Enrich BAM Data

There are cases in which the data that is available at operation time does not contain everything you need for reporting purposes. For example, you may have a ProductID but not a ProductName at runtime. Since the BAM Activity represents an abstraction independent of how the data is actually collected, it should contain an item named as the final data that you want to see in the report "ProductName". Just like any other item, you can use this in interpretive constructs such as milestone groups, durations, dimensions, and measures. Since the ProductName is not available at runtime, you must get some additional data that is sufficient for performing a lookup, such as the ProductID.

You should collect the data in the same column, instead of the data that you need for reports. For example, you should collect the ProductID instead of ProductName at runtime. If more columns are required, you may create more items in the activity but not use them in any View.

To enrich BAM data via lookups

1.   Deploy your BAM definition.

2.   In SQL Enterprise Manager, add your server that contains the data of interest as "Remote Server" under "Security".

3.   Locate the data analysis package named BAM_AN_<*View Name*>. For example if the view is SalesMgr, this will be BAM_AN_SalesMgr.

4.   Set the zoom to magnify the view of the package (e.g. 100%)

5.   Add a SQL connection that you will be using in the lookups.

6.   Locate the transform data task after the step "Cleanup Staging". This is where you move the data from the PrimaryImport to the StarSchema database. There are two instances of this task—one for the completed activities, and another for the one that is in progress. Apply all the rest of the steps to both tasks.

7.   Click the transformation.

8. Select Lookups; add your lookup "LookupProductByID" using the lookup connection (see SQL books online for lookups). If for example the lookup is a simple table "LookupProduct", the with columns ProductID and ProductName, the text of the lookup will be:

9. Click the Transformations tab. Delete the default data transformation "Transform", and create ActiveX transformation instead. Click on Source Columns and add all columns. Click on Destination Columns and add all columns.

10. Click on the General tab, and then Properties. This results in automatic generation of a script that performs the trivial copy transformation as shown:

11. Change the value by using the lookup as shown:

12. Save and then run the package.

13. Ensure that the correct data ends up in the OLAP cube. You should save the package as VBScript or a structured storage file, because it contains your custom code, not just the auto-generated steps from BAM.

Displaying Localized BAM Search Results

In certain localized languages such as Chinese, you must add the appropriate language support to Windows so that Internet Explorer can display the BAM search results correctly. The instructions provided here are for adding Chinese language support.

GB18030 is a Chinese National Standard for encoding characters. Windows 2000, Windows XP, and Windows Server 2003 support GB18030. FortheBAM search pages to display GB18030 characters in Internet Explorer, you need to add GB18030 support for these two fonts: Tahoma and Arial.

To enable G18030 support

1. Install the language support for Simplified Chinese on any non-Simplified Chinese version of Windows. On a computer with Windows XP or Windows Server 2003 installed:

2. Click **Start**, click **Control Panel**, and then select **Regional and Language Options**.

3. On the **Languages** tab, check the **Install files for East Asian languages** check box.

4. Install the GB18030 support package as a download from http://go.microsoft.com/fwlink/?LinkId=26235.

    This support package the font necessary to display GB18030 characters (SimSunGB18030.ttc) as well as a Unicode-to-GB18030 conversion tool (gbunicnv.exe), and a file necessary to support the 4-byte character set

(ms4bsp.dll). The conversion tool is found in C:\Program Files\GB18030Tools folder.

Font linking provides the ability to render non-Latin characters when the font set by the application does not support those particular characters. It is suitable when an application cannot use a single font face to render all the characters required.

To enable GB18030 font linking on your system

1.   Click **Start** and then click **Run.**

2.   In the "Run" text box, type **regedt32** to open the registry. (You can also use "regedit.exe" on Windows XP or later.)

3.   Navigate   to   [HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\FontLink\SystemLink]

4.   Double-click on the "Tahoma" key, in the right panel.

5.   Add "SimSun18030.ttc,SimSun-18030", in a new line, in the edit box.

6.   Double-click on the "Arial" key, in the right panel.

7.   Add "SimSun18030.ttc,SimSun-18030", in a new line, in the edit box.

8.   Click **OK** to restart the computer.

Deploying Localized BAM XML Files

Microsoft SQL Server 2000 Analysis Services supports Unicode mapping. However, there are known character corruption issues with SQL Server Analysis Services and Visual Basic Unicode mapping. Specifically, if the regional settings are not set to the correct localized language when the conversion from ANSI to Unicode occurs, the conversion is performed using the wrong locale information and character corruption occurs.

To deploy the BAM definition XML file successfully, you must switch your system locale to the correct locale before you can actually deploy a BAM definition XML file that contains localized characters.

Tracking Profile Editor

This section describes the Tracking Profile Editor (TPE). The TPE is a tool that developers use to create or modify tracking profiles. The TPE consists of a graphical user interface that allows developers to map a specific view of internal business processes, as well as associated data, to orchestrations or pipelines (ports), or a combination both orchestrations and pipelines.

Business processes such as ordering, invoicing, and inventory tracking are the fundamental actions of e-commerce. Microsoft BizTalk Server 2006 enables knowledge workers to gather specific or aggregated real-time information about the

state of a business process. For example, they can find out why a specific customer order was processed but not shipped, or how many units of a product were sold and shipped in the previous 48-hour period.

The Tracking Profile Editor (TPE) is a key component in the end-to-end business activity monitoring solution that Microsoft BizTalk Server 2006 offers. The Business Activity Monitoring (BAM) Framework provides an easy, real-time, transaction-consistent way to monitor heterogeneous business applications, and present the data for SQL queries and aggregated reports by using online analytical processing (OLAP) technology. Knowledge workers can use this information to assess the current state of their business so that they can quickly respond to changes.

For example, to troubleshoot a specific customer order or analyze trends of specific business events and milestones in the shipping process, a business analyst can specify the view of the data he wants to see. To extract the data the business analyst requires, the developer uses TPE to perform the mapping between the specific business event data and the actual orchestration. After you create, save, test, and deploy the tracking profile, the business analyst can retrieve the data collected to analyze a specific aspect of a business process.

Before you start working with the Tracking Profile Editor (TPE), we recommend that you understand BizTalk Server and how it works to facilitate communications across different platforms and applications.

As a solution developer, you also need to have specialized knowledge about how the business processes of your organization are implemented using Microsoft BizTalk Server orchestration. You must understand how to use the Orchestration Designer to produce orchestrations. For more information about Microsoft BizTalk 2006 orchestration, see About Orchestration . For more information about valid and invalid orchestration shapes using the TPE, see Best Practices for Tracking Profiles.

In This Section

- What is the Tracking Profile Editor?

- Components of the Tracking Profile Editor

- Using the Tracking Profile Editor

- Tracking Profile Editor Keyboard Shortcuts

- Best Practices for Tracking Profiles

- Security Considerations for Tracking Profile Editor

- Walkthrough: Creating a Tracking Profile

What is the Tracking Profile Editor?

The Tracking Profile Editor (TPE) is a mapping tool targeted at developer audiences. Users of the TPE create a map, known as a tracking profile, between items in a BizTalk Activity Monitoring (BAM) activity and the BizTalk Server solution sources for those items.

Once the business user has defined the activity, the developer connects the activity to the implementation. One source of the implementation is a BizTalk Orchestration. The TPE facilitates the connecting of the implementation by providing a drag and drop experience that does not require the developer to provide code.

A profile is a set of characteristics that define a business-related process. A tracking profile is a file with a .btt extension. It contains the mapping between an orchestration or messaging artifact and the activity definition

What is a Tracking Profile?

A profile is a set of characteristics that define a business-related process. A tracking profile is a file with a .btt extension. It contains the mapping of these characteristics from an activity to orchestrations and ports.

Why Create A Tracking Profile?

Users of the TPE create a map, or tracking profile, between items in a BAM activity, such as milestones and data items and the BizTalk Server solution sources for those items.

When you create a tracking profile using the TPE, you are working with the following objects:

- BAM Activities

- BizTalk Orchestrations in deployed Assemblies

- Receive and Send Ports

- Message Schemas in deployed Assemblies

- Context Properties

- Primary Import Database

- BizTalk Management Database

- BizTalk DTA database

You define the data extraction from an orchestration by dropping items from message schemas, orchestration shapes and context properties into business milestone (event) and data item folders.

For example, consider a BAM activity that includes a milestone called PO Received and has a Messaging port through which purchase orders flow to initiate processing. The developer can associate the *PO Received* milestone with a BizTalk Messaging property called *PortEndTime* for the port in their solution. Semantically, this indicates that the PO is successfully received once the receive port concludes its action and populates the *PortEndTime* property. The developer makes this and any other mappings to complete the tracking profile. All items in the activity are mapped if they have a BizTalk Server source, or are left unmapped to be populated by API calls directly if the source of the data or event is from a process outside or BizTalk Server's run-time environment.

Although each pane or view in the TPE has a unique function, all the views and folders have similar navigational features to help you find and manipulate information. This topic describes the layout of the TPE and the purpose of various views and folders.

Who Uses the TPE?

Users involved with Enterprise Integration Development utilize this tool to map BizTalk event sources to BAM target activities. The resulting .BTT file is handed off to IT Implementation for deployment.

IT Implementation users will typically apply tracking profiles using command line tools (BTTDeploy). The IT user can also use the TPE tool directly to apply the tracking profile.

Users in IT Operations may be responsible for periodic updates to tracking profiles on a scheduled basis (including any database operations required, such as backup and restore), especially as the result of the activity change-on-the-fly (additions only) feature.

Components of the Tracking Profile Editor

This section contains information about the TPE user interface and the functions it performs.

In This Section

- TPE Menu Options

- What Is an Activity View?

- What is the Source Event View?

- TPE Activity View Nodes

TPE Menu Options

This topic describes the menu options of the Tracking Profile Editor (TPE).The main menus include File, Tools, and Help.

Additional functionality specific to messaging interception is exposed on the context menu of items displayed in the Activity View in the left pane of the application.

File Menu

The File menu contains the following options:

- **New** – Creates a new tracking profile. This option is always available.

- **Open** – Opens an existing tracking profile. This option is always available.

- **Save** – Saves the tracking profile currently being edited. If no tracking profile is being edited, this option is not available.

- **Save As** – Saves the tracking profile currently being edited to a name you specify. If no tracking profile is being edited, this option is not available.

- **Import BAM Activity Definition** – Imports an Activity Definition that has been applied. This menu item has the same functionality as the watermark link in the left frame, meaning the user can start this action from either place (the watermark being a kind of usability hint). This option is always available.

- **Exit** – Exits the TPE. Exiting during an active edit invokes a Save or Save As per Windows standards. This option is always available.

New Profile

The **New Profile** menu option opens a blank tracking profile source file with no contents against an empty BTT file (a tracking profile file). The default name of the new profile is TrackingProfile1 (...2, 3, etc.).

Open

The **Open** menu option allows the user to locate and select a single profile with which to work. Profiles are dependent on the assemblies they were created against being available in order to render and validate the profile.

Opening the profile automatically re-establishes the mapping source information, the BizTalk Management database, and the assemblies as they existed when the user last edited the profile. If the original deployed assembly is no longer available or you need to change the assembly with which the profile is associated, you have the option of explicitly invoking the **Set Mgmt DB** action to refresh this linkage. You can also edit the profile and apply it.

Save

The **Save** option saves the tracking profile currently being edited. The first time Save is selected on new a tracking profile the option invokes the **Save As** dialog.

Save As

The **Save As** option allows you to specify the folder and the name for the tracking profile. The default extension for the tracking profile file is .btt.

Import BAM Activity Definition

The **Import BAM Activity Definition** option opens the Import BAM Activity Definition dialog box which is populated with a list of BAM activity definitions found in the BAM Primary Import Table.

This feature is also invoked by clicking the jumpstart link in the Activity View of a new tracking profile.

Retrieve Existing Profile Check box

This check box allows you to extract and include existing tracking profile mappings for the activity definition. The check box is not selected by default.

Exit

The **Exit** Option closes the TPE. If there has been a modification to the tracking profile you are prompted to save the profile the Tracking Profile Editorcloses.

Tools Menu

The Tools menu contains the following options:

- **Apply Tracking Profile** – Stores the profile in the BAM Primary Import database.

- **Remove Tracking Profile** – Remove the tracking profile mappings from the BAM Primary Import database.

- **Set Management Database** – Set the management database that the TPE is working against.

- **Options** – Allows you to set TPE options.

Apply Tracking Profile

The **Apply Tracking Profile** menu option stores the tracking profile mapping to the specified assembly as well as to the corresponding orchestrations and ports. All the artifacts that are referred to in the tracking profile should be available in a specified BizTalk Management database. All new instances of the relevant services use the

new tracking profile when starting. If the specified assembly or accompanying artifacts do not exist, then an error is presented which advises the user to reopen the profile and specify a deployed assembly before it can be stored.

You can work against multiple assemblies at the same time. You do not work with the assemblies concurrently, but multiple assemblies can be accessed in a single editing session relative to any single BAM activity definition. In other words, if you import a BAM activity definition and then select three assemblies from which to draw tracked items, the profile that TPE creates is applied to multiple profiles that are associated with the orchestrations and ports that are used in the tracking profile. The tracking profile always contains information about all assemblies associated with any given BAM activity definition.

Remove Tracking Profile

The **Remove Tracking Profile** menu option removes the tracking profile for the loaded activity definition and its corresponding orchestrations and ports. You are prompted for confirmation before this action is completed.

Set Management Database

The **Set Management Database** menu option specifies the BizTalk Management database from which mapping sources can be chosen, and through which BAM activity definitions are found. By default, the tracking profile is set to the default BizTalk Management database.

SQL connection strings can be specified in one of two manners:

- You can specify just the server name, and the TPE will connect to the default port.

- You can specify a server name and port pair, ServerName,port number. The TPE will connect to the computer running SQL Server that is using the specified port.

What Is an Activity View?

An activity view contains the imported BAM activity definition that you create with the BAM Add-In for Excel or the ODBA Wizard. The Activity View is on the left pane of Tracking Profile Editor (TPE) user interface. The BAM activity definition is an abstract of your trace requirements for your business process. An activity can span multiple orchestrations and ports. You import the activity once and map it to each orchestration or messaging artifact that fulfills some part of the definition.

Activity view elements

The activity view displays the overall structure of the tracking profile in tree view and includes the following elements:

- Milestones

- Data items for the activity

- Event sources

- Data sources

Milestones: Milestones are objects which define a point in a given process. Milestones are accessed in one of three ways:

- You can drag a shape from an orchestration schedule, and the end-time for that shape's execution is reported by the engine as the milestone value.

- You can drag a Messaging property, from a schematic representation on the right, to a target milestone.

- You can drag a message payload schema node which contains a milestone value.

Data items: Data items are objects which define a particular element from an xml schema for a message instance, system, or promoted property. You access the data item by expanding the schema to find and select the element you are interested in and dragging the element to the correct data item type folder. Information about the data items (for example, XPath) are stored in the profile.

The context menus of available actions for the activity view dynamically change depending on the node selected in the Orchestration View. For example, if you select an activity folder node, the shortcut menu will contain the shortcut menu items for that activity folder.

You associate events and data to the items in the business activity by dragging them from the source event pane on the right to the event or data node in the Activity view.

Activity view context menus

The context menus for the nodes in the activity view are accessible by right-clicking a node in the tree. The following tables describe the items in the context menus for the different nodes for an activity view.

Activity Definition Tree root node

Menu Item	Usage
New Continuation	Inserts a new Continuation folder into the Activity tree. Used in conjunction with a ContinuationID folder to provide a means to hand off processing between multiple components that populate the same activity. Examples of components are BizTalk orchestrations, ports, BufferedEventStreams, and DirectEventStreams.
New ContinuationID	Inserts a ContinuationID folder into the Activity tree. Used in conjunction with a Continuation folder to provide a means to hand off processing between multiple components that populate the same activity. Examples of components are BizTalk orchestrations, Ports, BufferedEventStreams, and DirectEventStreams.
New Relationship	Inserts a new relationship folder into the Activity tree. Used to publish the relationship between activities that form a view.
New Document Reference URL	Inserts a new Document Reference URL folder into the Activity tree. Used to set a reference URL to a location that contains a document that is related to this activity.

Property Node

Menu Item	Usage
Associate Selected Data	Used to create an association between a and the BAM Activity.

Event node

Menu Item	Usage
Associate with the end of the selected action	Used to create an association between a message payload or context property data item and the BAM Activity data item folder.

What is the Source Event View?

The Event Source View is where the Tracking Profile Editor (TPE) presents the orchestrations or message schemas selected from the assemblies or the context properties which you map to the activity definition.

The Source Event View is presented in the right pane of user interface and the contents of the pane vary based on the data source you selected.

Event Sources

You have four choices for event sources Orchestrations, Messaging Payloads, Context Properties and Message Properties.

Orchestrations and Messaging Payloads require that you select an assembly from which to map your data items. You then select the specific orchestrations or message payload schemas of interest from the assembly. For orchestration schedules you are given a list of the orchestrations in the assembly, messaging payloads allow you to select from a list of messaging property schemas, and context properties presents you a list of schemas in the assembly and publicly available system schemas.

When you select messaging properties you are given a list of the known messaging properties that you can then map to the activity.

When you select context properties, you are first given a list of property schemas that represent the properties. When you select the context property, the related schema of the context property is shown on the right pane. You can then map the context property to your activity by dragging and dropping the property to an activity node.

Orchestration Schedule View

The Orchestration View displays the steps by which BAM transacts the business process contained within the selected orchestration. This view typically contains low-level technical details, such as invocations of custom code and exception recovery logic. The view is located on the right pane of the Tracking Profile Editor (TPE) user interface.

Working with the Orchestration Schedule View

You select your orchestration view by clicking the **Select Event Source** button and clicking the **Select Orchestration Schedule** menu item. You then choose an assembly from which to select an orchestration. Once you select the orchestration, you can drag orchestration shapes that represent orchestration constructs or actions, from the orchestration into the business milestone folders in the Assembly view to do the following:

- Create a mapping between the low-level implementation of the business process and the Business Activity Monitoring (BAM) activity.

- Indicate the timestamp to track when the associated underlying construct or action has completed.

Right clicking a shape in the orchestration schedule view opens a context menu that allows you to message payloads, context properties, or message properties, if any, that are associated with the shape.

For an explanation of the Orchestration shapes displayed in the Orchestration View,

The list of available orchestrations can be extensive. If you know part of the name of the orchestration for which you are searching, you can type it in the **In String** text box and click the **Search** button. This will select only those orchestrations that contain the partial string you have entered.

Messaging Payload View

The Messaging Payload view displays the schema of the XML message associated with the selected action (for example, Message Sent or Received). The view is available from the shortcut menu for some of the shapes in the Orchestration View.

Working with the Messaging Payload View

You select your orchestration view by clicking the Select Event Source button and clicking the Select Messaging Payload Schema menu item. You then choose an assembly from which to select a schema. Once you select the schema, can drag elements from this schema to the data item folders of the activity, thus indicating you want to extract that data from specific XPath expression inside the message at this action.

You can open message payload views from the orchestration schedule view by right clicking a shape that contains a message payload. This will open a context menu from which you can click the Message Payload menu item to retrieve a list of payloads associated with the shape.

Context Property View

The Context Property displays the schema of the XML message associated with the property. The view is available from the shortcut menu for some of the shapes in the Orchestration View.

Working with the Context Property View

You select your context property view by clicking the **Select Event Source** button and clicking the **Select Context Property** menu item. You then choose a context property from the list of known context properties to load the schema for that property. Once you select the property, can drag elements from the associated schema to the data item folders of the activity, thus indicating you want to extract that data from specific XPath expression inside the message at this action.

You can open context property views from the orchestration schedule view by right clicking a shape that contains a context property. This will open a context menu from which you can click the **Context Property Schema** menu item to retrieve a list of context properties associated with the shape.

The list of available context properties can be extensive. If you know part of the name of the property for which you are searching, you can type it in the **In String** text box and click the **Search** button. This will select only those properties that contain the partial string you have entered.

Messaging Property View

The Messaging Property view displays the schema of the XML message associated with the property. The view is available from the shortcut menu for some of the shapes in the Orchestration View.

Working with message properties

You select your message property view by clicking the **Select Event Source** button and clicking the **Select Message Property** menu item. You then choose an message property from which to select a schema. Once you select the schema, can drag elements from this schema to the data item folders of the activity, thus indicating you want to extract that data from specific XPath expression inside the message at this action.

You can open message property views from the orchestration schedule view by right clicking a shape that contains a message payload. This will open a context menu from which you can click the Message Payload menu item to retrieve a list of payloads associated with the shape.

The list of available message properties can be extensive. If you know part of the name of the orchestration for which you are searching, you can type it in the **In String** text box and click the **Search** button. This will select only those message properties that contain the partial string you have entered.

TPE Activity View Nodes

This topics listed below describe the six types of tree nodes used by Tracking Profile Editor to map a specific view of internal business processes, as well as associated data, to an orchestration.

In This Section

- Activity and ActivityID Nodes

- Continuation and ContinuationID Nodes

- Relationship Nodes

- Business Event Nodes

- Data Item Nodes

- Document Reference URL Nodes

Activity and ActivityID Nodes

Pre-defined activity nodes exist in the activity definition file that the developer imports from the knowledge worker. Develops can add or delete Relationship nodes,Continuation folders, ContinuationID nodes, and Document Reference Url nodes as required. All data items and business event nodes are subordinate to and contained within the associated activity node.

Working with Activity nodes

For example, consider the following scenario; the EquityLoan orchestration contains the activity folder LoanProcess. It references business events including the following:

- LoanApplicationReceived

- CHRequest

- CHResponse

- AppraisalRequest

- AppraisalResponse

- Approved

- Denied

The ActivityID node enables the solution developer to extract data that uniquely identifies the activity, such as a purchase order number, or, in the case of the

sample scenario, the SSN field of the message. If you do not drag any data to the ActivityID node, an automatically generated GUID identifies the business activities.

To define the relationship between business events or milestones in different orchestrations, the target orchestration must reference the ActivityID. For more information about how to implement the concept of relationship using TPE, see <u>Relationship Nodes</u>.

Continuation and ContinuationID Nodes

In some cases, the actual implementation of a business activity may involve two or more related orchestrations. With correlated orchestrations, some of the events or data items for each business activity will occur in one orchestration, and some in another.

Working with Continuation nodes

The Continuation node contains data items that indicate a unique instance ID, also called a continuation token. By using the continuation token, developers can link to other activities using the ContinuationID node.

In order for BAM to correlate the activities, the Continuation node and the ContinuationID node must have the same name.

Relationship Nodes

Relationship folders are used whenever an activity definition file contains more than one activity. The names of the folders match the name of the associated activity. You form the link by matching the name of the relationship folder to the activity ID of the related activity and by matching the values for the data items. You define each relationship using a separate node.

Working with relationship nodes

To indicate the unique instance identifier that links the data item between activities:

- Map a data item to the ActivityId node of the main orchestration.

- Drag and drop a data item with the same name as above to the relationship node in the related activity. The relationship node has the same name as the Activity node of the main activity.

For example, in the sample scenario, there could be a related but separate business process represented in an orchestration called RefinanceOrchestration. That orchestration could contain a LoanRefinance activity node, a Refinance ActivityID, and orchestration shapes such as Receive Appraisal Request. The relationship node and relationship ID, however, could be LoanID, indicating the link to the original LoanProcess activity.

Business Event Nodes

Business event or milestone node define events that are date or time specific. Pre-defined business event and milestone folders exist in the activity definition you import from the knowledge worker, and you cannot delete them without re-importing the activity definition file.

Working with business event nodes

You can drag shapes from the orchestration into the business events folder to track those events as execution of the shapes is completed. It is important to note that you can only map from the orchestration under which the parent activity resides. For example, if you have two orchestrations A and B, and you import the definition for B, then when you map events, you can only map from Orchestration B to the business event folder.

TPE uses the name of the shape for the name of the event folder, and this folder will have a unique icon. For example, in the sample loan-process scenario, TPE names them according to the event, such as Approved or Denied. You should only track business events once per business process when that business process spans multiple tracking profiles. If you have a conditional path in your orchestration, you can select the business event from both paths, as only one will ever execute. You should not select a shape inside a loop.

Data Item Nodes

Data item nodes exist in the activity definition file that the developer imports from the knowledge worker created observation model. The Tracking Profile Editor (TPE) names them for the data items they are tracking, such as Customer Name. You then drop one or more data items from the Message Schema View onto the node that corresponds to Customer Name.

Working with data item nodes

When mapping data item nodes you should only track data items once per business process when that business process spans multiple tracking profiles. If you have a conditional path in your orchestration, you can select the data item from both paths because only one will run. However, you should not choose a shape inside a loop unless the values will be different for the data item with each iteration.

Document Reference URL Nodes

Document Reference URL nodes exist in the activity definition file that the developer imports from the knowledge worker created observation model. Document Reference URL nodes contain references to a location that contains a document that is related to this activity. This can be any type of document, for example an activity that represents a purchase order approval work flow, the Document Reference URL might point to the underlying purchase order document.

Working with Document Reference URL nodes

The data source for the Document Reference URL is typically the **MessageRefURL** BizTalk Server system property. You can also use custom schemas that store URLs and map the property to the Document Reference URL from the custom schema.

Items to note about Document Reference URLs:

- You can add one or more Document Reference URLs, but each node must have unique name with the activity.

- The **MessageRefURL** property needed to populate a Document Reference URL node is only populated with a value in the case of WSS, FILE, and FTP transport adapters.

- There is no BAM user interface that displays this reference to business end-users. The intent is to allow custom user interface developers to gain access to associated document information so that they can present it to the end user.

Using the Tracking Profile Editor

You use the Tracking Profile Editor (TPE) mapping that developers use to map orchestrations and properties to BAM activity definitions.

Users of the TPE create a map, or tracking profile, between items in a BAM activity such as milestones and contextual data (sometimes called the visibility wish-list) and the BizTalk Server solution sources for those items.

For example, consider a BAM activity that includes a milestone called "PO Received". The developer knows, from having created it in other BizTalk Server development tools, that the actual process includes a Messaging port through which purchase orders flow to initiate processing. The developer determines that the activity milestone called "PO Received" is most correctly associated with a BizTalk Messaging property called "PortEndTime" for the port in their solution. The developer makes this and any other mappings to complete the tracking profile by loading the activity, selecting event sources, and dragging the appropriate items from the event source and dropping it on the corresponding nodes in the activity tree definition.

To create a tracking profile two prior steps that must have taken place:

1. A BAM activity has been defined by the business analyst, as part of an overall observation model, and deployed by your system administrator.

2. A BizTalk Server solution (Orchestrations, schemas, map, pipelines, etc.) has been successfully deployed in the target environment.

These steps are necessary since, after installation, the tool is not populated with any data to retrieve from the databases.

Considerations for Working with Tracking Profiles

You should consider the following items when you work with tracking profiles.

BAM and MOM integration and tracking

Integrating Business Activity Monitoring (BAM) and Microsoft Operations Manager (MOM) will not work for a specific activity instance.

If you create a tracking profile that does not have a mapped Send or Receive shape, a MessageID record is not generated for the activity relationship table.

In an orchestration-to-orchestration scenario where BAM data is tracked from both orchestrations, the second orchestration may not persist MOM integration data correctly.

Applying tracking profiles that monitor running processes

Updating a tracking profile can impact activity instances in progress if the activity includes a BAM continuation (see "Continuations"). Specifically, if the update to the tracking profile specifies a downstream interception of data for an activity item already recorded, it is possible that the original value will be overwritten. In essence, any single event stream will not be affected by the application of tracking profile updates because each stream object is tied to the specific version of the profile which was in place at the time the activity/stream started. However, because continuations are the means of correlating multiple event streams, streams not yet begun at the time of a profile update will pick up the changes in the update, thus introducing the possible data overwrite described.

Tracking profiles without a Send or Receive shape from which to draw message properties

Continuations track activities through context properties or payload data that are the same between activities. Properties such as Message Id, Service Id, and Instance Id change value between activities.

You can create tracking profiles to handle this scenario by the following methods:

- If an orchestration sends a message through dynamic binding to another orchestration, then a globally unique payload data value can be used for the continuation.

- If there is no unique payload data in the message, then the interchange ID of the message can be used. To use the interchange ID, you must track it on the same Receive shape. You cannot use the interchange ID if you create a new message, as the interchange ID changes when the new message is created. Tracking the interchange ID from any shape other than a Receive or Send shape is not reliable.

- You can also use message ID as long as the exact same message that is received from the pipeline is used in the orchestration, that is, the orchestration port is bound to a pipeline and a Receive shape and the message ID are tracked from that location. If you track the message ID from a different shape, then the message ID will be invalid for use in continuations.

- If an orchestration calls another orchestration and no message is passed, or an orchestration calls another orchestration but the callee does not have any Construct, Receive, or Send shape where payload data values can be retrieved, the user can use the instance ID. The instance ID for the called orchestrations does not change.

- If the orchestration loads another orchestration through an exec call rather than calling it directly, then there is no way to use any static message properties to track the activity. The user cannot enable a continuation. The only way to perform tracking in this instance is if a message is passed through the pipeline that contains unique payload data on which to perform the tracking.

You cannot use a session ID as continuation token for pass-through pipelines

In a tracking profile when you select items from a message payload the tracking profile is bound to the schema of that message. In a pass-through pipeline, the schema name value is a null value. When BAM attempts to load the configuration by port name and schema name it cannot make the match to the session ID schema and no data is tracked by the engine.

For pass-through pipelines, you can either remove all payload tracking from the tracking profile or use XML pipelines if you do need to track the payload data.

Using unique port names

When enumerating two-way ports, the TPE displays them as two logical ports, a send and a receive port. Each of these logical ports is displayed with the same name. BizTalk Server allows you to create one-way and two-way ports with the same name. For example, you can create a two-way port named "Port1" and then create a receive port with the same name. In these cases the TPE displays the receive port once and the two-way port twice, once as a receive port and once as a send port.

TPE will apply tracking profiles to both receive ports in this case. We recommend that ports be given unique names to help identify them properly.

Using TPE orchestrations with a parallel shape as the first shape

If you begin an orchestration with a Fork, Parallel, or Listen shape, you cannot map an activity ID on one of the branches. In cases of parallel processing you can map the ActivityID above the Fork shape. You can also let BAM generate the activity ID to avoid the issue of a parallel shape at the top of the orchestration.

Availability of message properties at design time

When creating tracking profiles, not all message properties are available. This is most likely to be encountered when the shape where the message properties are mapped from is at the top of an orchestration. In these instances, the value of the message properties is null.

An example of this is where the Listen shape is the first shape in an orchestration. When message properties are mapped from this shape, only the following properties have values: InstanceID, ServiceID and ServiceClassID. (MessageID is not in scope at this point and has a null value).

You cannot map shapes inside a while Loop shape to report a milestone

The TPE blocks mapping sources contained from within a Loop shape to activities that are mapped to items that are outside of the loop.

Looping activities refers to actions that loop within an orchestration. It is possible to capture the events from actions that loop within an orchestration. To do this, you create another activity and map all of the new activity milestones and data inside the loop. This is necessary because the data processing in the loop will occur more than once per schedule execution.

For example, If you have a purchase order with multiple line items that process in a loop, questions like "Which purchase orders have item prices of $100?" are ambiguous. Unambiguous questions are:

Which purchase orders have line items with a price of $100?

Which purchase orders have Total/Min/Max item prices of $100?

Creating unambiguous questions requires thinking of the line items as something separate from the purchase order. In the Tracking Profile Editor, the root activity (for example, a purchase order) maps to all actions outside the loop. The child activity (for example, a line item) maps to the actions inside the loop.

The typical approach to working with these types of constructs is to decompose the repeating loop and to have a related activity based on the inner activity that is related to the outer activity.

You need to use a payload item as the ActivityID for the root activityand have this payload item available in some of the messages inside the loop. Map the activity to the relationship node that displays under the child activity and name it as the root activity.

The tracking profile must be deployed last

If a tracking profile spans multiple applications, then the tracking profile in the application that will be deployed last application deployed. Otherwise, when the

deployment wizard calls BTTDeploy.exe you will receive the following message: "mapping source not found."

Mapping multiple BizTalk Server artifacts to a document reference URL or MessageID nodes

The TPE allows you to drag and drop from multiple BizTalk Server artifacts onto the same document reference URL or MessageID node.

In cases where multiple sources are mapped to the same item in a BAM activity, the last artifact that was encountered during BAM processing is the one that persists in the tracking data and is viewable in the BAM portal.

Updating BTT versions to match BizTalk solution versions

BAM developers and administrators can maintain version synchronization between tracking profiles and BizTalk solutions by automating the updating the version of information in the BTT file. To do this, modify the **Version** attribute in the **DataLevel** element of the BTT file. In the following sample element, you would modify the Version information in the TargetAssemblyName and SchemaName attributes.

```
<DataLevel Name="City" SourceTypeSelected="Messaging Payload"
TargetAssemblyName="TheImplementationOfOrderMgmt, Version=1.0.0.0,
Culture=neutral, PublicKeyToken=c5b33e44ffa4658b"
SchemaName="TheImplementationOfOrderMgmt.PurchaseOrder,TheImplementation
OfOrderMgmt, Version=1.0.0.0, Culture=neutral,
PublicKeyToken=c5b33e44ffa4658b" SomXPath="/*[local-name()='&lt;Schema&gt;'
and namespace-
uri()='http://TheImplementationOfOrderMgmt.PurchaseOrder']/*[local-
name()='PurchaseOrder' and namespace-
uri()='http://TheImplementationOfOrderMgmt.PurchaseOrder']/*[local-
name()='Header' and namespace-uri()='']/*[local-name()='ShipTo' and namespace-
uri()='']/@*[local-name()='City' and namespace-uri()='']" XPath="/*[local-
name()='PurchaseOrder' and namespace-
uri()='http://TheImplementationOfOrderMgmt.PurchaseOrder']/*[local-
name()='Header' and namespace-uri()='']/*[local-name()='ShipTo' and namespace-
uri()='']/@*[local-name()='City' and namespace-uri()='']" IsBeginActivity="true"
IsEndActivity="false">
```

Some orchestration shapes cannot be tracked in the TPE

The Orchestration Engine does not generate events for the following shapes and thus cannot be tracked or mapped in the TPE:

- Task

- All Branches

- Suspend

- Terminate

- Throw

- Catch

- MessageAssignment (because it is part of the Construct shape)

- Transform (because it is part of the Construct shape)

- Loop

Creating Tracking Profiles

You create a new tracking profile or modify an existing one to better manage and monitor a specific business process for your organization. The Tracking Profile Editor allows you to define the data to collect to meet the business analyst's requirement. The profile you create or modify can be as simple or as complex as you like depending on the nature of your business requirements.

As a developer, you create a new profile based on a BAM Activity definition. A deployed activity definition may already have a profile defined for it. If not, you create a tracking profile by performing four tasks:

- Selecting a deployment server and database

- Selecting a deployed BAM Activity definition from the BizTalk Management database

- Defining the data extraction from the orchestration

- Connecting activities, if the actual implementation of your business process spans more than one orchestration you can

In This Section

How to Create a New Profile

How to Map Event Sources

How to Map Item Data

How to Create a Continuation

How to Map Multiple Assemblies

How to Apply and Remove a Tracking Profile

How to Create a New Profile

You create tracking profiles to link BAM Activity Definitions to deployed assemblies and BizTalk Server Messaging properties. When you open the Tracking Profile Editor, you can create a new by either clicking on the import activity link

Prerequisites

An existing management database to which you have permissions.

The TPE configured to use the management database.

An existing BAM Activity Definition in the management database.

Procedure Title

1.   On the computer or computers that you have identified as the source system, open the **Tracking Profile Editor**. Click **Start**, click **Run**, type **BTSTrkEditor.exe**, and then click **OK**.

2.   In the middle of the left pane of the TPE, click the **Click here to import a BAM Activity Definition link**. This opens the **Import BAM Activity Definition** dialog.

3.   From the BAM Activity Definition Name list box selecting the activity that your tracking profile will be based on. If it is difficult to locate the activity in the list due to the number of deployed activities, you can type part of the name in the In String text box and click the **Search** button. This will limit the activities displayed to those whose names contain the partial name entered.

4.   Once you have selected the activity, click the **Ok** button. This will open a new tracking profile based on the activity selected and display the profile in the left pane of the TPE.

How to Map Event Sources

You map event sources to gain access to data items that BAM tracks to generate alerts.

Prerequisites

To map an orchestration schedule as an event source

1.   Open an existing tracking profile or create a new tracking profile. For more information about creating a tracking profile, see How to Create a New Profile.

2.   Click the **Select Event Source** button (located above the right pane in the Tracking Profile Editor).

3.   Select the **Select Orchestration Schedule** menu item from the cascading menu.

4.   Select the parent assembly from which to draw the orchestration by clicking the assembly containing the orchestration in the **Assembly Name** list box, and then click **Next**.
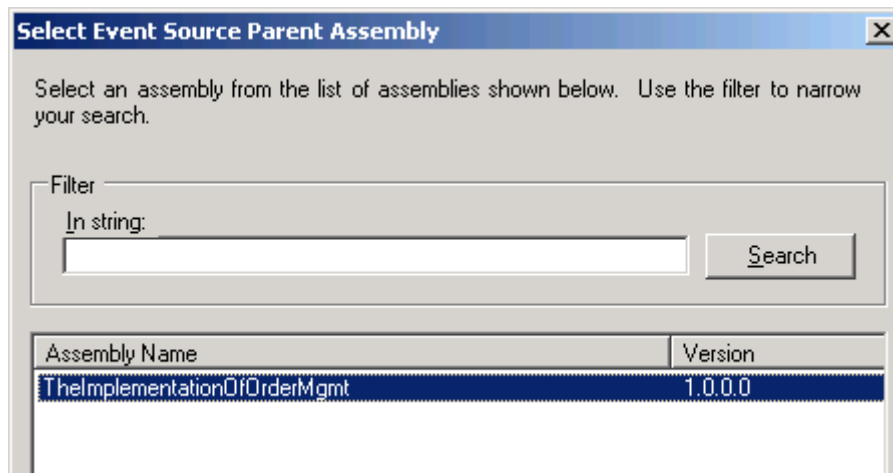
1.   Select the orchestration that is the source for the data items in the **Orchestration Names** list box, and then click **OK**.

2.   Select the data items in the right pane and drag them to the appropriate nodes in the activity in the left pane.

How to Map Item Data

You map Item data to define the data extraction from an orchestration.

Prerequisites

Deployed BAM activity definitions and orchestrations that you will connect.

Procedure Title

1.   Open an existing tracking profile or create a new tracking profile that references two activities. For more information on creating a tracking profile, see How to Create a New Profile.

2.   In the scenario, an activity definition called LoanProcessBamDef is imported. It contains the **LoanProcess** activity node, with a customer's **SSN** as an ActivityID. For more information, see Activity and ActivityID Nodes.

3.   Ensure that every activity has an ActivityID or a ContinuationID data item, such as customer SSN, to track.

4.   Map orchestration actions to the appropriate business events folder to indicate the event to track. For example, in a loan processing scenario, the

following items, among others, would be dragged under the **LoanProcess** activity folder:

- **LoanApplicationReceived**

- **CreditHistoryRequest**

- **CreditHistoryResponse**.

How to Create a Continuation

You create continuations to indicate which business events in one or more orchestrations are related by constructing connected activities.

Prerequisites

To perform this procedure, you must have deployed BAM activity definitions and orchestrations that you will connect to.

To create a continuation

1.   Open an existing tracking profile or create a tracking profile that references two activities. For more information about creating a tracking profile, see How to Create a New Profile.

2.   Identify a *continuation token,*which is a piece of unique information that is available to both activities. For example, if a **CreditHistory** activity is activated by a message sent from a **LoanProcess** activity within an **EquityLoan** orchestration, the SSN field of the message can be used as a continuation token because it is common to both activities.

3.   Right-click the first activity, and then select **New Continuation** to create a continuation node in the first activity (CreditHistory). Name the continuation node you just created.

4.   From the Orchestration Schedule View, select the continuation token you chose in step 2, such as SSN (in this case from the Send action) and drop it into the continuation node you created in step 3.

5.   Right-click the second activity and select **New ContinuationID** to create a Continuation ID node in the second activity. Name it using exactly the same name you chose in step 3, and drop in that node that contains the same data item (in this case, SSN from the Receive action).

6.   On the **File** menu, click **Save As** to save the tracking profile as a .btt file to the BizTalk Management database and to avoid overwriting any existing .btt file.

How to Map Multiple Assemblies

BizTalk applications can consist of multiple assemblies in which the data items that a BAM activity references reside. The following procedure shows you how to map multiple assemblies to a tracking profile.

Prerequisites

To map multiple assemblies

1.   Open an existing tracking profile or create a new tracking profile. For more information about creating a tracking profile, see How to Create a New Profile.

2.   Click the **Select Event Source** button (located above the right pane in the Tracking Profile Editor).

3.   Select the **Select Orchestration Schedule** menu item from the cascading menu.

4.   Select the parent assembly from which to draw the orchestration by clicking the assembly containing the orchestration in the **Assembly Name** list box, and then click **Next**.

5.   Select the orchestration that is the source for the data items in the **Orchestration Names** list box, and then click **OK**.

6.   Select the data items in the right pane and drag them to the appropriate nodes in the activity in the left pane.

7.   Repeat steps 2 through 6 to map additional assemblies.

Map the second assembly

1.   Click the **Select Event Source**.

2.   Select the **Select Orchestration Schedule** menu item from the cascading menu.

3.   Select next parent assembly from which to draw the orchestration by clicking on the assembly containing the orchestration in the **Assembly Name** listbox and click the **Next** button.

4.   Select the orchestration that is the source for the data items in the **Orchestration Names** listbox and click the **OK** button.

5.   Select the data items in the right hand pane and drag them to the appropriate nodes in the activity in the left hand pane.

How to Apply and Remove a Tracking Profile

Once you have created or modified the tracking profile, the next step is to apply it to a test database and verify the result through integration testing. You can apply the tracking profile from within Tracking Profile Editor (TPE) itself or by using the command line utility.

Prerequisites

A previously created tracking profile that you saved to your hard drive.

To apply the tracking profile from within the TPE

1.   Click **Start**, point to **Programs** and **Microsoft BizTalk Server 2006**, and then click **Tracking Profile Editor**.

2.   On the **File** menu, click **Open**. Navigate to the correct .btt file on your hard drive. Click **Open** to load it.

3.   On the **Tools** menu, click **Apply Tracking Profile** to apply the .btt file to a management database that you have set from the **Set Management Database** menu item on the **Tools** menu. Verify the result through integration testing.

4.   Notify the person in your organization responsible for deployment that the tracking profile tests correctly and is ready for deployment.

To apply the tracking profile from the command-line utility

1.   From the command prompt, run the bttdeploy.exe tool located in the \Tracking folder as follows:

2.   **Important**   You must have BizTalk Administrator privileges to use this tool.

3.   The Tracking Profile detects the exact Assembly Version, and bttdeploy.exe will fail if the Assembly fails to deploy.

4.   **Important**   When applying from the command line, BAM always applies the profile to the management database that you indicate when running the configuration wizard. It does not take into account the database setting that you have chosen in the Tracking Profile Editor option, "Set Management Database".

To remove a tracking profile

1.   Click **Start**, point to **Programs** and **Microsoft BizTalk Server 2006**, and then click **Tracking Profile Editor**.

2.   On the **File** menu, click **Open**. Navigate to the correct .btt file on your hard drive. Click **Open** to load it.

3.   On the **Tools** menu, click **Remove Tracking Profile** to remove the tracking profile based on the .btt file from the management.

Sample Tracking Profile

This topic describes a sample tracking profile for a loan process. In the example the business activity

LoanAppraisal

CreditHistory

EquityLoan

LoanCustomer (not shown)

The EquityLoan orchestration contains the Loan Process activity folder. That activity folder contains a unique identifier or ActivityID of customer SSN. The LoanProcess activity folder references loan process business events including:

LoanApplicationReceived

CHRequest

CHResponse

AppraisalRequest

AppraisalResponse

Approved

The following figure illustrates this business process in the Assembly view of Tracking Profile Editor (TPE). The data items map to the appropriate business event folders.

Tracking Profile Editor Keyboard Shortcuts

TPE Keyboard Shortcuts

The TPE provides keyboard shortcuts for navigating without a mouse. The following table lists the keyboard shortcuts that TPE supports.

Action	Keyboard shortcut
Create a new profile	ALT+F + N
Open an existing tracking profile	ALT+F + O
Save tracking profile	ALT+ F + S
Save As	ALT+ F + A

Import activity definition	ALT + F + I
Delete	Delete
Rename context menu item	F2
Open Help	F1
Apply tracking profile	ALT+ T + A
Remove tracking profile	ALT + T + R
Set Management Database	ALT + T + M
View Message Schema	Shift + F10 + M
Exit, close TPE	ALT+F + X
Select Orchestration Schedule	ALT + S + O
Select Messaging Payload	ALT + S + M
Select Context Property	ALT + S + C
Message Payload Schema	ALT + S + P
Context Property Schema	Shift + F10 + M
Message Property Schema	Shift + F10 + P

Best Practices for Tracking Profiles

This topic lists best practices guidelines and important notes for using Tracking Profile Editor.

Naming folders in Tracking Profile Editor

Note the following naming requirements when naming folders in Tracking Profile Editor:

- The length of the combination of folder name and data item instance values must not exceed 128 characters.

- For Continuation and ContinuationID folders, the naming of the folder is the key to correlating two orchestrations. For example, if Orchestration A is the parent of Orchestration B, Orchestration A contains a continuation folder whose name maps directly to the ContinuationID folder in Orchestration B.

Developing orchestrations for Tracking Profile Editor

'You cannot map an orchestration to a business activity if it starts or ends with an invalid shape. The Orchestration engine does not allow tracking for some shapes. They are:

- Message Assignment

- Transform

- Group (Task)

- Suspend

- Loop (While)

Use the following guidelines when mapping to business activities so that the Tracking Profile Editor and other Business Activity Monitoring (BAM) tools can use them:

- For the Group shape, use a non-transactional Scope shape.

- For the While shape, wrap it in a non-transactional Scope shape.

- For the Terminate shapes, there is no workaround, because the end event of this shape never occurs in a normal scenario.

- Do not start or end schedules with any of the shapes for which drag-and-drop operations are not permitted.

Security Considerations for Tracking Profile Editor

Solution developers, system administrators, or IT/Operations personnel must have administrative rights to retrieve or deploy the tracking profile into a database associated with an assembly.

With Tracking Profile Editor (TPE), administrators can:

- Retrieve an active tracking profile associated with one or more assemblies from the BizTalk Server management database

- Modify the tracking profile

- Apply a new or modified tracking profile into the BizTalk Server management database

- Modify saved tracking profile files (.btt)

Security   In the TPE workflow, assuming the development and deployment tasks are separated as is typically the case, the person responsible for deployment should have read-only access to the .btt file and the associated assembly file.

BAM End-to-End Walkthroughs

This section provides a series of walkthroughs designed to help you to learn how to enable real-time end-to-end visibility for a business activity, which represents a unit of work such as purchase order, using Business Activity Monitoring (BAM) for your applications. After completing these walkthroughs, you will be able to:

- Define a complete BAM observation model, which is a high-level definition of your business data (business analyst)

- Deploy a BAM observation model (administrator)

- Map a BAM observation model to an implementation (developer)

- View Key Performance Indicators (KPIs) and create alerts for them (business end user)

The scenario for this walkthrough is a simple order-management business process.

In order to achieve end-to-end visibility of a business activity, there are many roles that play an important part in configuring an application with BAM. Each walkthrough corresponds to a role and the tools used by that role, as described in the following table.

Walkthrough	Role	Tools
Walkthrough: Creating a BAM Activity with ODBA	Business analyst	Orchestration Designer for Business Analysts (ODBA) Task: Define the order management business activity.
Walkthrough: Creating a BAM Observation Model with the BAM Add-In for Excel	Business analyst or sales manager	BAM Add-In for Excel Task: Define the observation model by creating a specific Key Performance Indicator (KPI) for the business activity defined in the previous walkthrough.
Walkthrough: Deploying the BAM Observation Model	Administrator	BAM management command-line utility Task: Deploy the observation model. The management utility consumes the observation model and dynamically generates all of the BAM infrastructure (such as tables, Data Transformation Services (DTS) packages, stored procedures, and cubes) needed to correlate and manage the business events in the context of the activity.

| Walkthrough: Creating a Tracking Profile	Developer	Tracking Profile Editor Task: Map business events and data in the business activity (deployed in the previous walkthrough) to the actual implementation in the application.
Walkthrough: Consuming BAM Data	Business end user	BAM portal Task: Set custom business alerts on KPIs in order to monitor the real-time trends of the business.

In This Section

- Walkthrough: Creating a BAM Activity with ODBA

- Walkthrough: Creating a BAM Observation Model with the BAM Add-In for

Excel

- Walkthrough: Deploying the BAM Observation Model

- Walkthrough: Creating a Tracking Profile

- **Walkthrough: Consuming BAM Data**

Walkthrough: Creating a BAM Activity with ODBA

This walkthrough provides step-by-step procedures for creating a BAM activity, in this case an order management activity, by using Orchestration Designer for Business Analysts (ODBA). As a business analyst or the person responsible for creating the BAM activity, you begin by identifying key business milestones and data of interest for the activity. You conclude by exporting the activity, which includes the business milestones and data of interest to the business end user. The exported activity is the basis for the next walkthrough, "Walkthrough: Creating a BAM Observation Model with the BAM Add-In for Excel," where you identify the Key Performance Indicators (KPIs) for the activity.

Prerequisites

For the purposes of the walkthrough you must have previously created an ODBA file in Microsoft Office Visio 2003 containing the business activity defined in the following figure. The file name is assumed to be OrderMgmnt.vsd.
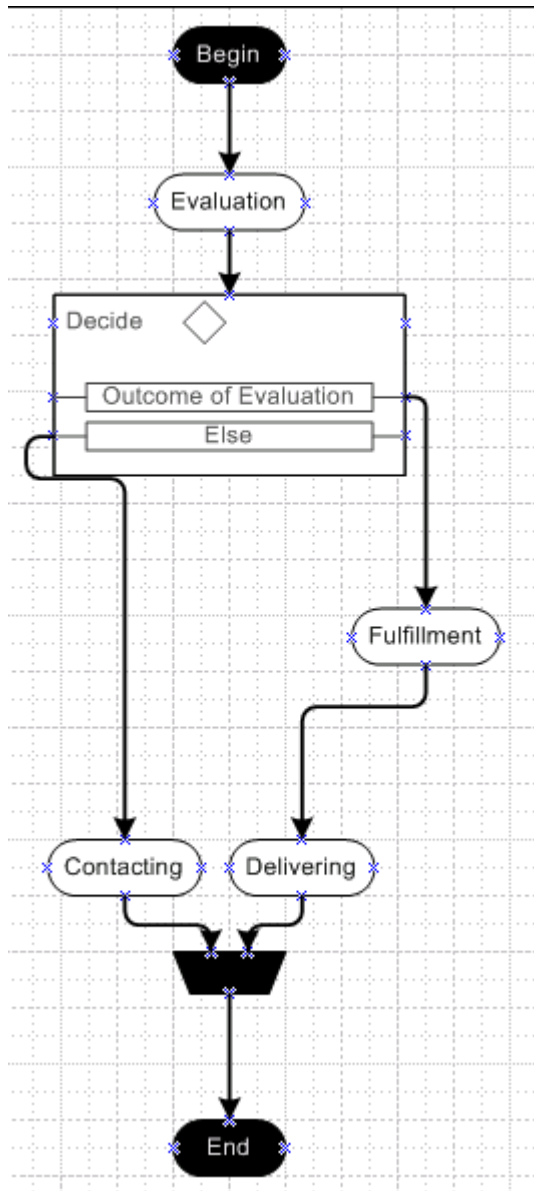
You should be familiar with the ODBA feature in BizTalk Server. For general information, see Defining Business Processes with ODBA . For specific information about information about creating the ODBA Visio file, see Creating a BAM Activity Definition with ODBA .

Open the order management activity

1.   Navigate to the location that contains the ODBA Visio file OrderMgmt.vsd.

2.   Double-click the file to open it in Visio.

Define the milestones of interest on the order management activity

1.   Select the arrow between the **Begin** and **Evaluation** shapes by right-clicking the arrow.

2.   Select the **Add Milestone** menu item from the context menu.

3.   In the milestone **Name** text box, type **received.**

4.   In the **Description** text box, type **Received is the first milestone of interest for the order management activity**, and then click **OK**.

5.   Select the arrow between the **Decide** and **Contacting** shapes on the left side of the diagram by right-clicking the arrow.

6.   Select the **Add Milestone** menu item from the context menu.

7.   In the milestone **Name** text box, type **denied** to name the milestone.

8.   In the **Description** text box, type **The Purchase Order was denied as a result of the evaluation stage**, and then click **OK**.

9.   Select the arrow between the **Decide** and **Fulfillment** shapes on the right side of the diagram by right-clicking the arrow.

10.   Select the **Add Milestone** menu item from the context menu.

11.   In the milestone **Name** text box, type **approved** to name the milestone.

12.   In the **Description** text box type **The Purchase Order was approved as a result of the evaluation stage**, and then click **Ok**.

13.   Select the arrow between the **Join** and the **End** shapes on the right side of the diagram by right-clicking the arrow.

14.   Select the **Add Milestone** menu item from the context menu.

15.   In the milestone **Name** text box, type **acknowledged** to name the milestone.

16.   In the **Description** text box, type **The current status of the order was acknowledged**, and then click **OK**.

Define the business data of interest for the OrderMgmt activity

1.   On the **ODBA** menu, click **Show/Hide Data of Interest Shape**.

2.   Double-click the data of interest shape which will open the data properties dialog box. Double-click the **Data of Interest** shape which opens the **Business Data of Interest** dialog box.

       Alternatively, you can right-click the **Data of Interest** shape, and then select **Edit Business Data Items**.

3.   In the **Name** text box, type **City** to create the name by which you can identify the data item.

4.   In **Data Type**, select **Business Data-Text** to specify that the type of data that this item contains is text.

5.   In the **Maximum Length** text box, leave the default of **50** in place.

6.   In the description dialog box, type **This is the city where goods ordered will be shipped**, and then click **OK**.

7.   Repeat steps 2 through 6 with the following data of interest names, types, and descriptions:

Name	Data Type	Maximum Length	1. Description
Amount	Business Data - Decimal	Not applicable	This is the amount of the purchase order
Product	Business Data - Text	50 (default)	This is the name of the product in the purchase order
State	1. Business Data - Text	50 (default)	This the name of the state where the purchase order originated

1.   Click **OK** on the **Business Data Item Properties** dialog box. This returns you to the **Business Data of Interest** dialog box.

2.   Click **OK** on the **Business Data of Interest** dialog box. This returns focus to the **Data of Interest** shape.

Export the OrderMgmt activity definition (BAM definition XML)

1.   Right-click the tab at the bottom of the Visio page titled **Page-1**, and then select the **Rename Page** context menu option.

2.   Type in **OrderMgmt**, and then press ENTER.

1.   On the **ODBA** menu, click the **Export BAM Definition File** menu item.

2.   Make sure that the **OrderMgmt** check box is selected in the **Export BAM Definition** dialog.

3.   Click the Browse button.

4.   Name the BAM Definition file **OrderMgmtODBA.xml** and save it to the desktop.

5.   Close the ODBA tool.

Walkthrough: Creating a BAM Observation Model with the BAM Add-In for Excel

In this walkthrough, the business analyst or sales manager creates a view through a set of dialog boxes in Microsoft Office Excel 2003. This view is based on the order management activity created in the previous walkthrough. Creating the view includes selecting specific data of interest and creating durations between business milestones, progress dimensions, and other activity artifacts. Once you complete the set of dialog boxes in Excel, you create the business process Key Performance Indicators (KPI) by dragging and dropping the specific dimensions and measures of interest.

Once you have created the observation model, you send the BAM definition to the administrator to deploy and create the BAM infrastructure, as described in the next walkthrough, Walkthrough: Deploying the BAM Observation Model.

Prerequisites

You must complete the following walkthrough: Walkthrough: Creating a BAM Activity with ODBA.

Import a BAM definition (observation model)

1.   Click **Start**, point to **Programs**, point to **Microsoft Office**, and then click **Microsoft Office Excel**.

1.   Click the **BAM** menu, and then click **Import XML**.

2.   Browse to the folder in which you stored your order management activity, and then select the OrderMgmtODBA.xml file.

3.   Click the **Open** button.

Create the Sales Manager view

1.   Create an Order Progress dimension by clicking the BAM menu and then clicking **BAM View**.

2.   Click **Next** on the **Welcome to the Business Activity Monitoring View Creation** dialog box.

3.   In the **BAM View** dialog box, click the **Edit an existing view** radio button, select the SalesManager view in the drop-down list, and then click **Next**.

4.   In **Edit BAM View: Name and Activities**, the name **SalesManager** is already filled in. Verify that the **OrderMgmt** activity check box is selected, and then click **Next**.

1.   In the **Edit BAM View: View Items** dialog box, verify that all the data items have already been selected, and then click **Next**.

2.   In the **Edit BAM View: View Items** dialog box, verify that the data items have meaningful alias names. The cycle duration has already been created between the received and the shipped business milestones. Click **Next**.

1.   On the **Edit BAM View: Aggregation Dimensions and Measures** dialog box, click the new Dimension button.

2.   Assign a name for the new dimension by typing **OrderProgress** in the **Dimension Name** text box.

3.   Select **Progress Dimension** in the **Dimension Type** drop-down list. Progress dimensions allow you to create aggregations of activities that are in process. For more information about progress dimensions,

4.   Click the **New Milestone** button.

5.   Type **Received** in the **Progress Milestone** text box. This specifies the name by which the milestone is known.

6.   Select **received (OrderMgmt)** in the **Business Milestone** drop-down list, and then click **OK**.

7.   Click the **New Stage** button.

8.   Type **Evaluation** in the **Progress Stage** text box. This specifies the name by which the stage is known. Click **OK**.

1.   Select the **Evaluation** stage in the **Progress milestones and stages** tree and click the **New Milestone** button. This will add a new milestone node for the received progress dimension.

2.   Type **Approved** in the **Progress Milestone** the text box.

3.   Select **Approved (OrderMgmt)** in the **Business milestone** drop-down list, and then click **OK**.

4.   Select the **Approved** milestone in the **Progress milestones and stages** tree, and then click the **New Milestone** button. This will add a new milestone node for the received progress dimension.

5.   Type **Denied** in the **Progress Milestone** the text box.

6.   Select **Denied (OrderMgmt)** in the **Business milestone** drop-down list, and then click **OK**.

7.   Select the **Approved** milestone in the **Progress milestones and stages** tree, and then click the **New Stage** button.

8.   Type **Fulfillment** in the **Progress Stage** text box, and then click **OK**.

9.   Select **Fulfillment** in the **Progress milestones and stages** tree, and then click the **New Milestone** button.

10.   Type **DeliveredAck** in the **Progress Milestone** the text box.

11.   Select **acknowledged (OrderMgmt)** in the **Business milestone** drop-down list, and then click **OK**.

12.   Select the **Denied** milestone in the **Progress milestones and stages** tree, and then click the **New Stage** button.

13.   Type **Contacting** in the **Progress Stage** text box, and then click **OK**.

14.   Select the **Contacting** in the **Progress milestones and stages** tree, and then click the **New Milestone** button.

15.   Type **DeniedAck** in the **Progress Milestone** the text box.

16.   Select **acknowledged (OrderMgmt)** in the **Business milestone** drop-down list, and then click **OK**.

17.   When you are finished, your Progress milestones and stages tree should look like the following screen.
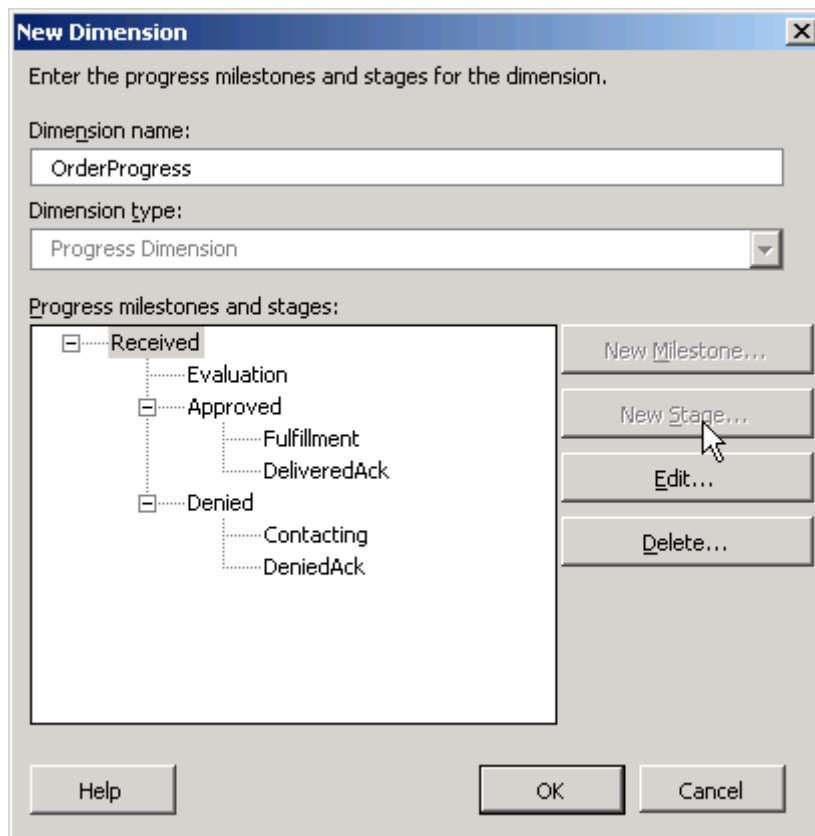
1.   Close the **New Dimensions** dialog box by clicking **OK**.

Create a count measure

1.   Click the **New Measure** button on the **Edit BAM View: Aggregation Dimensions and Measures** dialog box.

2.   Type **Count** in the **Measure name** text box.

3.   Click **Count** in the **Aggregation Type** frame, and then click **Ok**.

Create an average cycle duration measure

1.   Click the **New Measure** button on the **Edit BAM View: Aggregation Dimensions and Measures** dialog box.

2.   Type **AvgCycleDuration** in the **Measure name** text box.

3.   Select **CycleDuration (OrderMgmt)** in the **Base Data Item** drop-down list.

4.   Click **Average** in the **Aggregation Type** frame, and then click **OK**.

Create a time slice dimension

1.   Click the **New Dimension** button on the **Edit BAM View: Aggregation Dimensions and Measures** dialog box.

2.   Type **MyTimeSlice** in the **Progress milestone** text box.

3.   Select **Time Dimension** in the **Dimension Type** drop-down list.

4.   Select the **Received (OrderMgmt)** option in the **Base business milestone** drop-down list.

5.   Click **Year, quarter, month, day, hour, minute** in the **Display Settings** frame, and then click **OK**.

6.   Click **Next** on the **Edit BAM View: Aggregation Dimensions and Measures** dialog box.

7.   Click **Next** on the **Edit BAM View: Summary** dialog box.

8.   Click **Finish**. An empty PivotTable report and a prepopulated PivotTable field list are added to the worksheet.

Create the order progress business KPI

1.   In the **Pivot Table Field List**, select the **OrderProgress** field, and drag and drop it onto the **Drop Row Fields Here** box on the PivotTtable report.

1.   Select the **Count** field from the **Pivot Table Field List**, and drag it to the **Drop Data Items Here** box.

2.   Select the **TimeSlice** field from the **Pivot Table Field List** and drag it to the left edge of the PivotTable report.

3.   When the field is in the correct location the PivotTable report will look like the following figure.

1.   Double-click either cell on the PivotTable report containing **Received** to expand the table so that it shows all milestones and stages.

2.   Right-click a cell containing the Received milestone, and then select **Table Options** from the context menu.

3.   Type **OrderProgress** in the **Name** text box, and then click **OK**.

1.   Right-click a cell containing the Received milestone, and then select **Pivot Chart** from the context menu.

2.   Right-click the area around the chart, and then select **location** from the context menu.

3.   Click the **As object in:** radio button (make sure the SalesManager option in the drop-down list is selected), and then click **OK**.

4.   Place and size the chart to fit the worksheet and your preferences.

1.   Right-click a cell containing the Received milestone.

2.   On the PivotTable toolbar, click the real-time aggregation (RTA) icon. It is the last button on the right and looks like three squares collapsing into a pair of arrows.

3.   BAMExcelRTAButton

1.   Click the BAM menu, and then click **Export XML**.

2.   Click the Desktop button, type **OrderMgmt.xml**, and then click **Save**.

3.   Save the BAM Add-In for Excel with any name you want to the desktop, and then close Excel.

Walkthrough: Deploying the BAM Observation Model

In this walkthrough, the administrator deploys the BAM tracking configuration created in the previous walkthrough. You use the BAM management utility (bm.exe) to generate the BAM infrastructure. The BAM management utility consumes the BAM definition (observation model) that was created in the previous two walkthroughs.

Prerequisites

You must complete the following walkthroughs: Walkthrough: Creating a BAM Activity with ODBA and Walkthrough: Creating a BAM Observation Model with the BAM Add-In for Excel.

To deploy the BAM infrastructure

•   Click **Start**, and then click **Run**. In the **Open** text box, type **c:\Program Files\Microsoft BizTalk Server 2006\Tracking\BM deploy-all - DefinitionFile:c:\BAM\OrderMgmtExcel-out.xml**, and then click **OK**.

Walkthrough: Creating a Tracking Profile

In this walkthrough the developer maps the BAM activity (for these walkthroughs an order management activity) to the running business process, in this case a BizTalk solution consisting of an orchestration and messaging. To do this, you use the Tracking Profile Editor (TPE) in BizTalk Server 2006.

The TPE is a developer tool used to create and edit tracking profiles. Tracking profiles control the behavior of run-time components that intercept and store data. Changes in visibility requirements can be applied at any time by simply updating the tracking profile. By allowing visibility requirements to be modified at any time without impacting the running solution, tracking profiles provide an important point of flexibility for managing business processes.

The following steps take you through opening TPE, loading the activity, and then iteratively loading the various event sources and mappings between them and the target items in the BAM activity definition.

Prerequisites

Import a BAM activity

1.   Click **Start**, point to **Programs**, point to **Microsoft BizTalk Server 2006**, and then click **Tracking Profile Editor**.

2.   In the middle of the left pane of the TPE, click the **Click here to import a BAM Activity Definition** link. This opens the **Import BAM Activity Definition** dialog box.

3.   On the **Import BAM Activity Definition** dialog box, select **OrderMgmt** from the **BAM Activity Definition Name** list. Verify that the **Retrieve the current tracking settings for this activity definition** check box is not selected, and then click **OK**.

Establish the continuation folders for correlating activity data between parts of the process

1.   Right-click the **OrderMgmt** node at the top of the activity tree located in the left pane of the TPE. Select the **New Continuation** menu item from the context menu. Type **inPipelineThrows** as the name, and then press ENTER.

2.   Right-click the **OrderMgmt** node at the top of the activity tree. Select **New ContinuationID** from the context menu. Type **OrchCatches** for the name, and then press **ENTER**.

Load the BizTalk Server messaging property sources

1.   Click the **Select Event Source** button (located above the right pane of the TPE). Select **Select Messaging Property** from the cascading menu.

2.   Map the **Received** milestone. To do this, expand the top-level node labeled **<Schema>** and its first child node, **MessageProperties**. The tree now displays a list of available properties.

3.   Select the **PortEndTime** schema item and drag it onto the **received** node in the left (activity) pane.

4.   Right-click **PortEndTime** in the left pane and select **Set Port Mappings** from the context menu.

5.   Select the ReceivePO port in the left-side Port Name selection. Click the move item button (the top button with the greater than symbol '>' on it) to map the port, and then click OK. The port mappings dialog box provides a means to add and remove mappings. The list on the left is the set of ports from which to choose and the list on the right is always the list of active mappings. A mapping can contain one or more sources, but there must be at least one port mapped for each messaging source.

1.   Load a second BizTalk Server messaging source, this time for payload items. Click **Select an Event Source** button. Select the **Select Messaging Payload** from the cascading menu.

2.   In the **Select Event Source Parent Assembly** dialog box, select **TheImplementationOfOrderMgmt** from the list of assemblies, and then click the **Next** button.

3.   In the **Select Schema** dialog box, select **TheImplementationOfOrderMgmt.PurchaseOrder** from the list of schema names, and then click **OK**.

4.   Map the **City** and **poID** payload items. Select the **poID** schema item and drag it onto the **inPipelineThrows** item in the left pane. This establishes the token by which information recorded in the pipeline can be associated with other information recorded by the Orchestration.

5.   **Note**   The **poID** is located under the Header node of the schema.

6.   Right-click **poID** in the left pane and select **Set Port Mappings** from the context menu.

7.   Select the **ReceivePO** port in the left-side **Port Name** selection. Click the move item button (the top button with the greater than symbol '>' on it) to map the port, and then click **OK**.

8.   Select **City** in the schema, located under the **ShipTo** node, and drag it onto the **City** item in the left pane.

9.   Right-click **City** in the left pane and select **Set Port Mappings** from the context menu.

10.   Select the **ReceivePO** port in the left-side **Port Name** selection. Click the move item button (the top button with the greater than symbol '>' on it) to map the port, and then click **OK**.

Map additional BAM milestones

1.   Load the BizTalk Server orchestration source of the remaining BAM activity milestone items. Click the **Select an Event Source** button, and then **Select Orchestration Schedule** from the cascading menu.

2.   In the **Select Event Source Parent Assembly** dialog box, select **TheImplementationOfOrderMgmt** from the list of assemblies, and then click the **Next** button.

3.   In the **Select Orchestration** dialog box, select **TheImplementationOfOrderMgmt.OrderMgmtProcess** from the list of schedule names, and then click **OK**.

4.   Select the **FulfillmentDelay** shape and drag it onto the **Approved** milestone in the left pane.

5.   Select the **RejectionDelay** shape and drag it onto the **Denied** milestone in the left pane.

1.   Select the **SendPOAck** shape and drag it onto the **acknowledged** item in the left pane.

2.   Load the schema that acts as the source for the remaining activity items. Right-click the **ReceivePO** shape in the right pane. Select **Message Payload Schema** from the context menu.

3.   Map the remaining BAM activity items. To do this, select the **State** schema item, located under the **ShipTo** item, and drag it onto the **State** data item in the left pane.

4.   Select the **Product** schema item, located under Item, and drag it onto the **Product** data item in the left pane.

5.   Select the **Amount** schema item, located under the **Summary** item, and drag it onto the **Amount** data item in the left pane.

6.   Select the **poID** schema item, located under the **Header** node, and drag it onto the **OrchCatches** item in the left pane.

7.   (Optional) Save your work. To do this, save the BTT file using a name of your choosing by clicking the **File** menu item and clicking the **Save** menu item on the file menu.

Deploy the tracking profile

1.   Click the **Tools** menu and select **Apply Tracking Profile** from the menu. A dialog box will confirm the successful application of the profile. Click **OK** to confirm.

2.   Close the TPE.

Walkthrough: Consuming BAM Data

In this walkthrough you use the BAM portal to view BAM data, and you create and use BAM alerts. The BAM portal is a Web site that is part of BizTalk Server. The portal is designed primarily for the business end user, although it is useful to anyone who needs access to business performance data.

The following steps take you through opening the portal, reviewing the condition of a particular business process, defining an alert around a specific Key Performance Indicator (KPI) (in this scenario generating some inbound orders), and taking investigative actions when the alert is raised.

The walkthrough assumes 30 orders containing data conforming to the following format has been run through the order process.

<po:PurchaseOrder
xmlns:po="http://TheImplementationOfOrderMgmt.PurchaseOrder">

<Header poID="PO_8" poDateTime="2005-04-20T10:48:21Z">

<ShipTo Name="Contoso Ltd." Address="123 Some St." City="Redmond" State="WA" Zip="98052" />

<BillTo Name="Contoso Ltd." Address="123 Some St." City="Redmond" State="WA" Zip="98052" />

</Header>

<Item Product="Flash MC" Quantity="387" Price="1.73" />

<Summary Amount="669.51" />

<Timing EvalTime="4" FulfillmentTime="30" RejectionTime="29" />

</po:PurchaseOrder>

Access the BAM portal

1.   Click **Start**, point to **Programs**, point to **Microsoft BizTalk Server 2006**, and then click **BAM Portal Web Site**.

2.   In the **My Views** navigation pane at the left, click plus symbol (**+**) to expand the view called **SalesManager**.

3.   Click the Aggregations function to expand the list of available aggregations for the view. Click the **OrderProgress** aggregation.

4.   Click the Year item/header in the PivotTable and drag it to the navigation pane and drop it. This removes it from the PivotTable.

5.   On the PivotTable's toolbar, click the second icon from the right to open the **PivotTable Field List**.

6.   Select the **OrderProgress** item and drag it to the **Drop Row Fields Here** section of the PivotTable.

7.   Click the plus sign (**+**) to expand the cells in the PivotTable until **Evaluation** is shown with a corresponding total figure.

Create an alert

1.   Right-click the cell in the PivotTable containing the Evaluation total and select **Create Alert** from the context menu.

1.   Fill in the basic set of items needed to define and save an alert.

1.   In the **Name** text box, type **OrdersInEvaluation** to provide a meaningful name for the alert.

2.   In the **Message** text box, type **Alert has been sent**. This is the body of the alert mail that is received if the alert is raised.

3.   Enter threshold values that constitute a bottleneck in the processing of the orders. In the Threshold panel, select **Greater Than or Equal To** from the **Count** drop-down list (the default option). In the **Value** text box type **8** to set a value at which an accumulation of orders waiting for evaluation constitutes an unacceptable number in evaluation.

1. Set the security level for this alert. Select the **Allow others to see this alert and subscribe to it** check box. This allows other users of the portal to subscribe to this alert.

2. Click the **Save Alert** button in the upper-right corner of the **Alert Details** page.

3. Close the portal browser.Subscribe to the alert

1. In the **Alert Summary** frame, click the **Orders in evaluation exceed threshold** alert.

2. At the bottom of the **Alert Details** page, expand the **Subscriptions** frame by clicking the plus sign (**+**).

3. Click the **Add Subscriber** button.

4. Select **email** from the Transport drop-down list. This specifies that the alert will be delivered as an e-mail message.

5. In the **Address** text box type your e-mail address, and then click **Save**.

Receive an alert

1. Open your e-mail client and look for a message from BAM Alerting with a subject of **Orders in evaluation exceed threshold**.

2. Open the e-mail message and click the hyperlink in the alert e-mail message.

Investigate the items that caused the alert

1. Refine the data presentation and review the detailed status information for an order on the **refined query result** list. Expand the **Column Chooser** section of **Activity Search** page. Select the remaining items on the left side. Click the >> button to move the selected items to the list box on the right.

2. Collapse the **Column Chooser** section by clicking the minus sign [-] in the upper-left corner of the section. Click the **Execute Query** button located at the top-right of the page to run the query.

1. Review the detailed status information for an order on the refined query result list. Click any row in the **Results** section to access the underlying detailed status information for that item of work.

Open a trouble ticket to request investigative assistance

1. After reviewing the detailed status of the work item you may discover a problem with the order. Typical examples are errors such as missing Related Documents, an ordering of milestones that is incompatible with expected order, or empty data items that should not be empty. To get help from your system administrator, click the **Assistance** button at the bottom-right section of the page to open the **Request Technical Assistance** dialog box.

2.   In the **Subject** text box, type a meaningful subject for the assistance you are requesting.

3.   In the **Problem Description** text box, describe the problem for which you are requesting help.

4.   Click the **Send Report** button.

Monitoring Business Activities with BAM

Business Analysts use Business Activity Monitoring (BAM) to monitor data about business processes. Business analysts use the BAM workbook in Microsoft Office Excel to define what data to collect from business processes and to define the way in which business users will view the collected data.

Business analysts can use BAM to monitor processes that span multiple applications. For example, a purchase order process may have parts that are completed by different applications, including BizTalk Server, database activity, and messages sent from and to an internal Web site, a supplier Web site, and the shipper Web site.

Business users can use BAM to view both real-time and archived (historical) data about business processes. For example, a business user may want to know when a purchase order arrives for more than $100 (real-time data) or, how many purchase orders for more than $100 arrived this month (archived). BAM aggregates real-time and archived data to improve the speed at which it presents data.

For information about software and hardware requirements for installing BAM, download the latest version of the BizTalk Server Installation Guide, at http://go.microsoft.com/fwlink/?linkid=22120.

In This Section

- About Business Activity Monitoring

- Business Activity Monitoring Workflow

- Defining Data in BAM

- Using the BAM Workbook

- Defining Business Activities and Views in Excel

- Defining a Business Activity View

- Working with Preview Data

- Viewing Live BAM Data

About Business Activity Monitoring

There are two types of users who interact with the information worker features of BAM: business analysts and business users.

Business analysts use the BAM Excel Workbook to define the business processes they want to monitor. They also use the BAM Excel Workbook to define the way in which business users see the data collected by BAM. For example, a business analyst wants to present data about purchase orders and asks the following interesting questions about purchase orders:

- How many purchase orders did the company receive this (year, month, week, day, hour, and so on)?

- What is the current value of active purchase orders?

- What is the average time it takes for a purchase order to move though the stages of the purchase order process?

- How many purchase orders are in a particular stage of the process?

- How many customers ordered a particular product?

- What is the sum of the orders for a particular product?

- Are the quantities ordered different across geographic regions?

The business analyst uses these questions to define activities in the BAM Excel workbook that reflect the type of information that BAM should collect from the business process.

Business users use BAM to view the data defined by the business analyst and collected by BAM. A business user can use BAM in the following ways:

- View data about a single instance of a process. For example, if a business analyst created an activity to monitor a purchase order process, a business user could use BAM to see data about a particular purchase order.

- View data about related activity instances. For example, the invoice or shipments associated with a particular purchase order.

- View aggregated real-time or archived data about a process. Aggregated data is data over time. For example, a user could view the number of purchase orders with a value less than $500 that were received in the last (week, month, year, and so on), or the average value of purchase orders that arrived today.

View progress data, which allows a user to see data from the different stages of a process, such as the average time it takes for a purchase order to move from the received stage to the fulfilled stage.

Business Activity Monitoring Workflow

The following figure shows the four user roles who work with Business Activity Monitoring, and the tools that they use.

BAM Roles
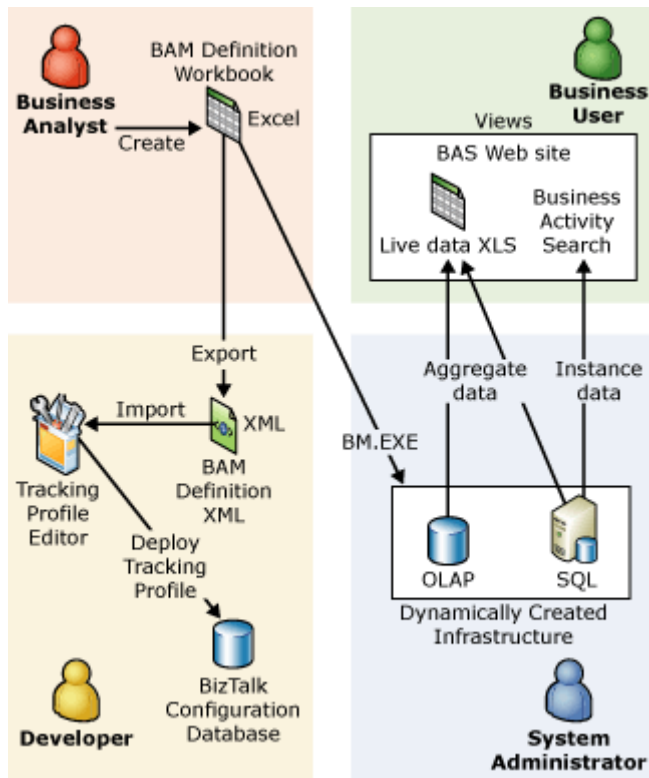
The following steps provide a high-level overview of the workflow for using Business Activity Monitoring.

Specifying the business data to collect

1.   The business analyst uses the BAM Activity Wizard to specify what data to collect for all business users.

2.   The business analyst uses the BAM View Wizard to define the view for each category of business user.

3.   When finished, he saves the activities and views in a Microsoft® Excel Workbook called the BAM Definition Workbook.

4.   The business analyst exports the BAM Definition Workbook to XML.

5.   The system administrator and developer use the XML to perform their roles.

Instructions for using the BAM Definition Workbook are located in Defining Business Activities and Views in Excel.

Managing the BAM Infrastructure

After the business analyst has defined the BAM view he wants, the system administrator uses the BAM management utility (BM.EXE), a command-line tool, to deploy the BAM infrastructure from the BAM Definition Workbook or the XML exported from the workbook.

The BAM management utility dynamically creates the tables, triggers, DTS packages, and OLAP cubes necessary to support the BAM view.

Each time the business analyst defines a different BAM view, or changes an existing BAM view, the system administrator must redeploy the BAM Definition Workbook.

Mapping the XML to an orchestration

1.   After the business analyst exports the BAM Definition Workbook to XML, the developer imports the XML file into the Tracking Profile Editor. The developer implements the business analyst's information requirements, mapping the XML to an orchestration.

    Using the Tracking Profile Editor, the developer performs the following steps to map the XML to an orchestration:

    a.   Loads the deployed assembly that is stored in the BizTalk Management database (also known as the Configuration database). The deployed assembly contains one or more orchestrations corresponding to the requirements that the business analyst specified in Step 1 above.

    b.   Defines the data to be extracted from an orchestration. You do this by dropping items from the message schemas and orchestration shapes into the appropriate business milestone (event) and data item folders.

    c.   When he is finished, he saves the profile as a BizTalk® Server tracking (.btt) file, to a storage database such as Visual SourceSafe.

2.   The developer deploys the .btt file to a testing database, and verifies the result through integration testing.

Deploying the Tracking Profile

Using the Tracking Profile Editor, the system administrator deploys the profile to one or more BizTalk Management databases.

Each time the developer changes the orchestration, or the requirements of the business users change and they want to track more data, the system administrator needs to redeploy the tracking profile using the bttdeploy.exe command line utility.

Viewing the business data

1.   The business user uses the _LiveData workbook, which is produced by the BM.exe utility. Each time the business user opens the _LiveData workbook, he receives a new live version of the data that is collected to monitor a specific aspect of the business process.

2.   To view data that is defined as real-time aggregation, the business user just needs to click Refresh in the workbook to view the data.

3.   If the aggregation data is not real-time, the business user views a snapshot of the business data that is taken at the time that the scheduled DTS package runs.

4.   If your organization has collaboration requirements, the business user can access the live data from the BAS Web site.

Defining Business Activities and Views in Excel

For the first step in any BAM solution you need to identify what data you're interested in and how that data should be interpreted. To do this, use the BAM Definition wizards from the Excel template, BAM.XLS. The wizards allow you to define a wish-list of the data of interest by Defining a Business Activity. You can also define the way the data should be interpreted and shown to different categories of business users. For more information about views, see Defining a Business Activity View.

The BAM wizards also enable you to define aggregations.

You can also rename activity items, for example, if the terminology some users are familiar with might not be exactly the same as the names of the corresponding data items in the activity. One reason could be if the view is in different language.

After you complete the wizards, Excel generates random preview data that you can use to define the desired charts, and so on. For more information about preview data, see Working with Preview Data.

You can use the BAM Definition workbook to process the data and to map the wish-list to the actual business implementation.

The BAM Microsoft® Excel template defines the data points and events that you need collected when a business process is run. You can get the BAM Excel template from a variety of sources. Ask your Business Administrator how to access to the template.

In This Section

•   Defining a Business Activity

•   Defining a Business Activity View

Defining a Business Activity

To indicate the data you need collected for reports, you must define a BAM activity. This contains a list of the important milestones and data items you want BAM to track. Use the BAM Excel template to create an activity as shown in the following procedure.

To define the way you want to monitor your business activities, you use the BAM.xls workbook as starting point. You can access this workbook in several places:

- If you have access to BAS collaboration site, you will find the workbook in **Business Activity Views**. For more information about Business Activity Views, see Defining a Business Activity View

- On the BizTalk CD search recursively for the workbook from the Msi\Program Files\Tracking folder. For example, in the US look for the Msi\Program Files\Tracking\1033 Directory.

- If you have full product installation of BizTalk you will find the workbook in the directory [Installation Drive]\Program Files\Microsoft BizTalk Server 2006\Tracking

To create a business activity

1.   Open the BAM template named **BAM.xls**.

2.   In the Microsoft Excel dialog box, click **Enable Macros**.

3.   On the **BAM** menu, click **BAM Activity**.

   The **Business Activity Monitoring Activity Wizard** appears.

4.   Click **New Activity**.

   The **New Activity** dialog box appears.

5.   Type a descriptive name for the activity in the **Activity Name** text box.

To create a new item

1.   Click **New Item**.

2.   In the **BAM Activity Item** dialog box, in the **BAM Activity Item** box, type a descriptive name for the activity item.

3.   From the **Item type** drop-down menu, select a type for this item. Possible values include:

Item type	Description
Business Milestone	A date/time value. For example, an approval date for a purchase order.
Business Data – Text	A string containing any alphanumeric characters. For example, Ship to: City, State/Province and Zip/Postal code.
Business Data – Integer	A whole number value. For example, the total number of purchases.
Business Data – Float	A decimal value. For example the total dollar amount of the PO.

4.   If you select the item type "Business Data – Text", you must enter the maximum number of characters for the string in the **Maximum length** box.

5.   Repeat steps 1 through 3 to add as many items as needed to this activity.

6.   After you complete the Business Activity Monitoring Activity Wizard, the Business Activity Monitoring View Wizard starts automatically.

For more information about using this wizard, see Defining a Business Activity View.

Defining a Business Activity View

Sometimes you do not want to expose all of your business data to all business users. You can create different BAM activity views that expose different types of information.

In This Section

- Using the BAM Excel Template

- Renaming Activity View Items

- Defining Durations

- Defining Milestone Groups

- Defining Aggregations

Using the BAM Excel Template

To use the BAM Excel template to define a BAM view, follow the steps in this procedure.

This procedure assumes you have already opened the BAM Excel template. You can access the template in several places:

- If you have access to BAS collaboration site, you will find the workbook in Business Activity Views.

- On the BizTalk CD, search for the workbook from the Msi\Program Files\Tracking folder. For example, in the US look for the Msi\Program Files\Tracking\1033 Directory.

- If you have full product installation of BizTalk you will find the workbook in the directory [Installation Drive]\Program Files\Microsoft BizTalk Server 2006\Tracking.

To open the BAM Excel template

1.   Open the file named **BAM.xls**.

2.   In the **Microsoft Excel** dialog box, click **Enable Macros**.

To define a BAM view

1.   On **BAM** menu, click **BAM Activity** and define your activity.

       For more information about creating an activity, see <u>Defining a Business Activity</u>.

2.   On the **BAM** menu, click **BAM View**.

3.   In the **Business Activity Monitoring View Wizard**, click **Next** to continue.

4.   On the **BAM View** page, verify that **Create a new view** is selected, and click **Next**.

5.   On the **New BAM View: Name and Activities** page, in the **View Name** box, type a descriptive name.

6.   In the list of activities, select the check boxes for each activity you want included in the BAM view you are currently defining, and then click **Next**.

7.   On the **New BAM View: View Items** page, select the check boxes for each activity item you want included in the BAM view you are currently defining, and click **Next**.

Renaming Activity View Items

Often the data you collect for a business activity contains sensitive information, for example credit card numbers or social security numbers. This means that some types of business users should see only a filtered subset of the data.

You rename a view item in the BAM View wizard. Use this wizard to create business activities, items, and views. For more information about using the wizard, see <u>Define Business Activities and Views in Excel</u>.

To rename activity view items

1.   On the Excel menu, click **BAM**, click **BAM View**, and then click **Next** until you see the **Select the items for this view** page.

2.   On the **Select the items for this view** page, select the items in the list and then click **Next**.

3.   In the second **New BAM View: View Items** page, click **Edit**.

4.   In the **Rename data item to** box, type a new name, and then click **OK**.

Defining Durations

Duration specifies the amount of time, defined by a beginning and ending milestone for a particular activity item. For example, the number of days a PO is valid.

Use the BAM View wizard to set or modify durations. Before you can set durations, use this wizard to create business activities, items, and business activity views. For more information about using the wizard, see <u>Define Business Activities and Views in Excel</u>.

To define durations

1.   In the BAM View wizard, click **Next** until you see the second **New BAM View: View Items** page, and then click **New Duration**.

2.   Type a name for the duration.

3.   From the drop-down list, select **Start business milestone** and an **End business milestone**, and then click **OK**.

Defining Milestone Groups

Use a milestone group to put related milestones together, for example the beginning and end milestones that define how long a Purchase Order is valid. You can then use the milestone group as a single milestone.

Use the BAM View wizard to create new milestone groups. Before you can create milestones or milestone groups, use this wizard to create business activities, items,

and business activity views. For more information about using the wizard, see <u>Define Business Activities and Views in Excel</u>.
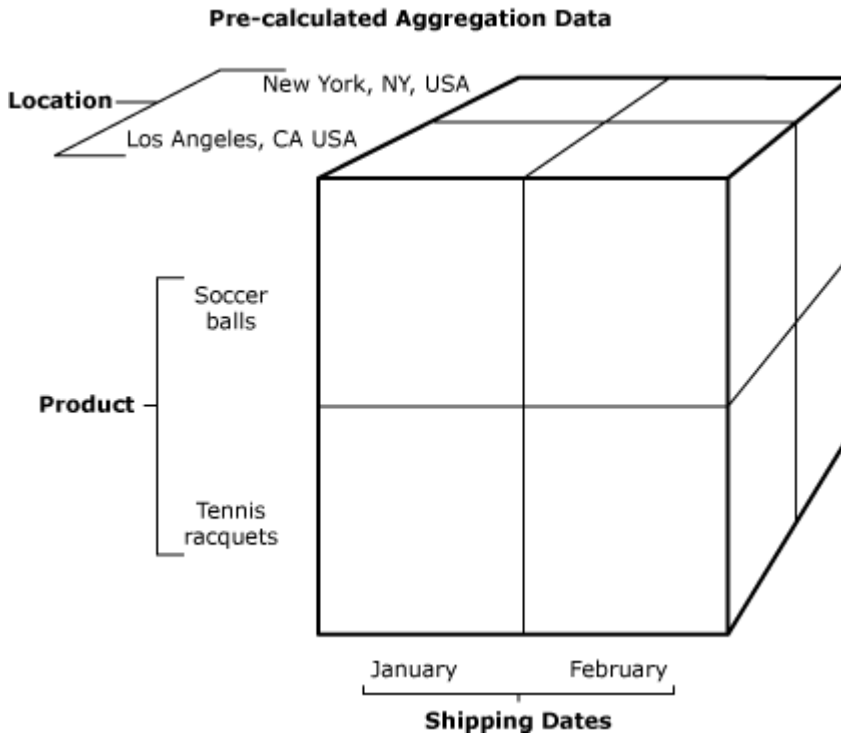
To define milestone groups

1.   In the BAM View wizard, click **Next** until you see the second **New BAM View: View Items** page. Click **New Group**.

2.   Type a new milestone alias.

3.   In the window below the alias box, click the milestones you want to include in the group, and then click **OK**.

4.   When you finish adding aliases, durations, and groups, click **Next**.

5.   To complete the BAM View wizard, click **Finish**.

Defining Aggregations

Excel defines **aggregations** as pre-calculated summaries of data that improve query response time by having the answers ready before the questions are asked. For example, when a data warehouse fact table contains hundreds of thousands of rows, a query requesting the shipping schedules for two particular products can take a long time to answer if the fact table has to be scanned to compute the answer. However, the response can be almost immediate if the summarization data to answer this query has been pre-calculated.

The following figure displays an example of pre-calculated aggregation data.

Pre-calculated Aggregation Data

Pre-calculated Aggregation Data

Location —— New York, NY, USA
Los Angeles, CA USA

Product — Soccer balls

Tennis racquets

January   February

Shipping Dates

The above figure summarizes the numbers of each product, shipped to specific locations over a two-month time period. Excel typically defines this data as **measure**. The data used for filtering and categorization, Excel defines as **dimension**.

For the user experience of browsing multidimensional data, see the Pivot table topic in Excel Help.

In This Section

- Defining Measures

- Defining Dimensions

Defining Measures

A measure defines how an activity is calculated. When you create a BAM view you can define a measure for the activity in the view. For example, for a purchase order activity, you can calculate the total PO amount, the number of POs, the average PO amount, and so on.

BAM supported measures include:

- Sum

- Count

- Average

- Minimum

- Maximum

To add new measures

1. In the BAM View wizard, click **Next** until you see the **New BAM View: Aggregation Dimensions and Measures** page. Click **New Measures**.

2. Type a name for your measure.

3. From the drop-down list, select the **Base data item**.

4. Click the **Aggregation type** you want to use. The choices include:

   - Sum

   - Count

   - Average

   - Minimum (not supported)

   - Maximum (not supported)

5. Click **OK**. When you finish adding dimensions and measures, click **Next**.

6. On the **New BAM View: Summary** page, review the information regarding your new view, and then click **Next**.

7. Click **Finish** to complete the **BAM View Wizard**.

If you added measures and dimensions to your BAM view, you can add those items to the PivotTable in the Excel workbook. For more information about creating a PivotTable.

Defining Dimensions

Microsoft Excel defines dimensions as categories used to organize data in a table into levels that will be used for analysis. For example, a location data dimension might contain levels such as city, state/province, and country/region. When creating BAM Views in the BAM View wizard, you can add one or more of the following dimension types:

- Progress Dimension

- Data Dimension

- Time Dimension

- Numeric Range Dimension

Pre-calculated Aggregation Data

Pre-calculated Aggregation Data
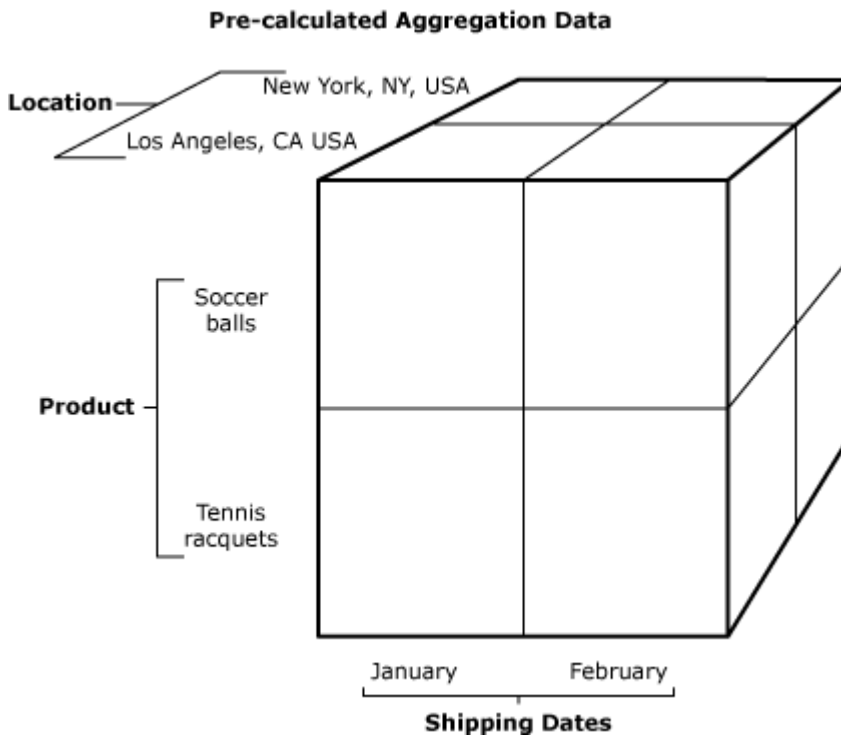
You add new dimensions in the BAM View wizard. Before you can add dimensions, you need to use the wizard to create business activities, items, and business activity views. For more information about using the wizard, see Define Business Activities and Views in Excel.

To add new dimensions

1.   In the BAM View wizard, click **Next** until you see the **New BAM View: Aggregation Dimensions and Measures** page. Click **New Dimensions**.

2.   Type a name for the dimension.

3.   From the drop-down list, select a dimension type.

4.   Based on the dimension type you selected, fill in the appropriate data, and then click **OK**.

Progress Dimension

BAM introduces a new type of dimension: the Progress Dimension. This dimension allows the creation of aggregations relating to the progress of activities still in process.

For example, consider a purchasing business process where you receive 1,000 Purchase Orders. You can use the progress dimension on rows to create the following table.

OrderProgress_Level1	Count
Received	1000

You can then open the Received process to view further details about the progress of the activities, such as:

		Count
Received	Evaluating	300
	Approved	500
	Denied	200

This means that from the 1000 purchase orders you received, 500 were approved, 200 were denied, and 300 are currently being evaluated.

Received, Approved, and Denied represent **milestones**. The corresponding numbers in the Count column show how many orders have passed through these milestones. Evaluating is a **stage** that the orders pass through between the Received and Approved or Denied milestones.

You can use progress dimensions in combination with any other dimensions. For example, by using the progress dimension Order Progress on rows and the data dimension Product on columns, the following results occur:

		Tennis Racquets	Soccer Balls
Received	Evaluating	250	50
	Approved	200	300
	Denied	150	50

Progress Dimensions provide especially useful information for charts based on real-time aggregations (RTA). RTAs allow you to see the current state of the business process and to easily identify process bottlenecks.

For more information about RTAs see, Real-time Aggregations.

Creating Progress Dimensions

To create progress dimensions, you use the BAM View wizard. Before you can create dimensions, you need to create business activities, items, and business activity

views. For more information about using the wizard, see <u>Defining Business Activities and Views in Excel</u>.

To create progress dimensions

1.   On the Excel menu, click **BAM**, and then click **BAM View**.

2.   In the BAM View wizard, click **Next** until you see the **New BAM View: Aggregation Dimensions and Measures** page. Click **New Dimension**.

3.   On the **New Dimension** page, type the following information:

   • Dimension name: **OrderProgress**

   • Dimension type: **Progress Dimension**

4.   Click **New Child** to open the **New Progress Stage** dialog box and type (or select) the following information:

   • Progress stage: **Received**

   • Business milestone: **PurchaseOrder.Received**

5.   Repeat the previous step to create a second child process stage using the following information:

   • Progress stage: **Evaluating**

   • Business milestone: **PurchaseOrder.Received**

6.   Click **Evaluating**, and then click **New Sibling**. In the **New Progress Stage** dialog box type (or select) the following information:

   • Progress stage: **Approved**

   • Business Milestone: **PurchaseOrder.Approved**

7.   Click **Approved**, and then click **New Child**. In the **New Progress Stage** dialog box type (or select) the following information:

   • Progress stage: **Fulfilling**

   • Business Milestone: **PurchaseOrder.Approved**

8.   Click **Fulfilling**, and then click **New Sibling**. In the **New Progress Stage** dialog box type (or select) the following information:

   • Progress stage: **Delivered**

   • Business Milestone: **PurchaseOrder. Delivered**

9.   Click **Approved**, and then click **New Sibling**.In the **New Progress Stage** dialog box type (or select) the following information:

- Progress stage: **Denied**

- Business Milestone: **PurchaseOrder.Denied**

10.   Click on **OK** to save the new Progress Dimension.

Data Dimension

Defining a data dimension allows the value of some text items in the BAM activity to be used on rows or columns. For example a data dimension named Product can be used to create the following table:

Product	Count
Tennis racquets	100
Soccer balls	200

Also, more than one data dimension can be defined in the BAM view wizard. For example, defining a data dimension named **Location** with levels for **State** and **City** can be used to create the following table:

	California		Washington
	Los Angeles	San Francisco	Seattle
Tennis racquets	50	20	30
Soccer balls	130	50	20

In this table, the Product dimension was used as the rows, and the Location dimension was used as the columns.

Time Dimension

The time dimension allows aggregations to be created with respect to time. For example a time dimension can be used to create the following table:

Year	Month	Count
2003	January	120
	February	230
	March	350

	April	280

The time dimension can be combined with any other dimension. For example, you can use the time dimension on rows and the data dimension on columns to create the following table:

	Tennis racquets	**Soccer balls**
January	50	70
February	120	110
March	300	50
April	220	60

Numeric Range Dimension

The numeric range dimension allows aggregations to be categorized based on friendly names of given ranges. For example, a business analyst can define a numeric range dimension named PO Size with the ranges Small for purchase orders between 0-$100, Medium for purchase orders between $100 to $1,000, and Large for purchase orders exceeding $1,000.

PO Size	**Count**
Small	500
Medium	350
Large	225

Working with Preview Data

When you have defined a BAM view that includes dimensions and measures, you need to update one or more PivotTables associated with that view.

A PivotTable report in Excel is an interactive table that enables you to easily combine and compare large amounts of data. The values in its rows and columns can be rotated to look at different summaries of the displayed data. A PivotTable also enables you perform calculations on the data, such as aggregate counts or averages.

To use the PivotTable after you have created it, follow the steps in this procedure. For more information about using PivotTables, see the Microsoft Excel documentation.

To use the PivotTable

1.   Create a BAM view to be used with a PivotTable. For more information about creating a BAM view, see <u>Defining a Business Activity View</u>

2.   Using the **Excel PivotTable Field List**, drag-and-drop one or more available dimensions into the column and row areas of the PivotTable template.

3.   Using the **Excel PivotTable Field List**, drag-and-drop one or more available measures into the data items area of the PivotTable template.

        As you are updating the PivotTable, you will notice that it is populated with sample data. This helps you determine how the PivotTable should be configured.

4.   Using the **PivotTable** toolbar, associate a chart with the PivotTable.

5.   Save your Excel workbook.

Defining Real-Time Aggregations

In some cases, specific slices of the multi-dimensional aggregations are so time-sensitive that you want them to be available in real time. For example, your business is selling perishable products and you want the aggregation of product quantity in different stages of delivery to be available in real time. At the same time, you want other aggregations such as the age of your typical customers, but only at the end of the month for business intelligence analysis.

•   Using the **PivotTable** toolbar, mark the selected PivotTable as a real-time aggregation (RTA), if applicable. The type of data contained in the PivotTable should determine whether it needs to be viewed in real-time or not. For example, a report that tracks hourly Web orders might need to viewed in real-time, whereas a report that tracks daily sales totals would not be viewed in real-time.

RTA button
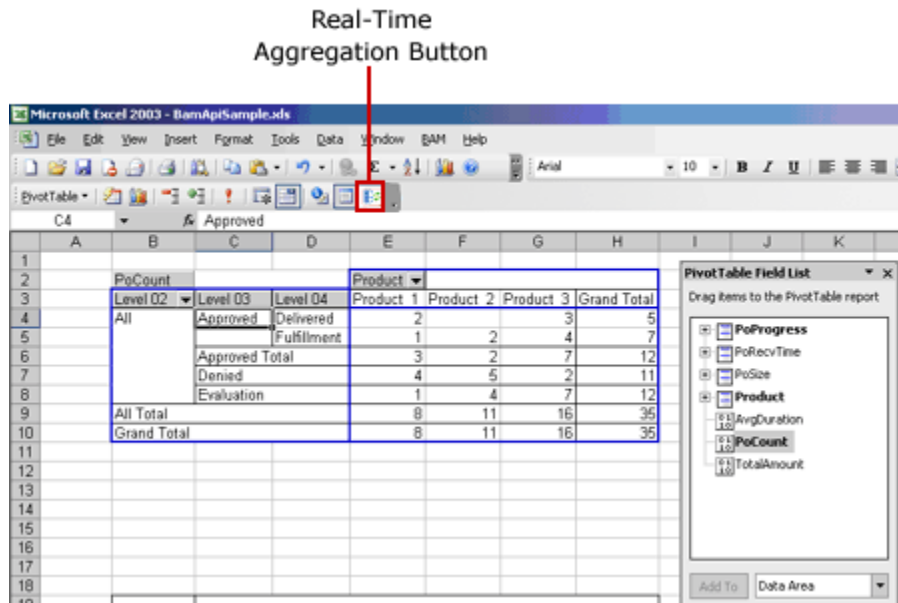
For more information about Pivot tables, see Excel online help. After you create your Pivot table, you can view your live BAM data.

Viewing Live BAM Data

Following are the different ways you can view live BAM data.

- View a single activity instance (for example, Purchase Order or Loan Process) as it evolves in real-time or from the historical (archived) records. This view shows only the data relevant at the business level and hides complexity of the diverse implementation. For more information about viewing a single activity instance, see Viewing an Individual Activity.

- Navigate to the related activity instances, for example, Shipments associated with given Purchase Order, or the Invoice in which it is included. For more information about navigating to related activity instances, see Viewing Related Activities and Related Documents.

- Browse Aggregations (Key Performance Indicators) around all the Business Activities that are currently being processed or have already happened. The Aggregations can be done in Real-Time or can be based on a snapshot of the Activities taken at specific time. For more information about browsing aggregations, see Viewing BAM Aggregations.

- Use Progress Dimensions, one of the new features in BAM. These dimensions allow filtering or grouping of the Activities at a given stage of completion. For more information about Progress Dimensions, see Progress Dimension.

- Search for activity instances based on their progress or business data. For example, you can search for loans that are waiting for customer signature and the dollar amount is greater than a given value. For more information about searching activities in BAM, see Searching and Navigating Activities.

In This Section

- Using the BAS Collaboration Site (deleted 1/17/06)

- Viewing BAM Aggregations

- Searching and Navigating Activities

Using the BAS Collaboration Site (deleted 1/17/06)

The BAS environment consists of a Web site hosted in Microsoft Windows SharePoint Services. Windows SharePoint Services provides a common user interface for all of the services included in BAS.

BAS works with Microsoft Office to provide collaborative functionality to Business Users. You can interact with and work together around business processes in this collaborative environment. BAS enables you to be proactively involved with business processes by providing a shared workspace that correlates business events and information and that provides the tools for working with them. Use this site to view your BAM aggregations, or search for individual activity instances.

For more information about the BAS collaboration site, see the BAS Information Worker User Guide.

Viewing BAM Aggregations

The BAM aggregations appear in Excel exactly as defined by the business analyst.

After the system administrator deploys the workbooks, he must upload the corresponding live data workbooks to the BAS Web site.

If you do not use the BAS Web site, obtain the _LiveData Workbook from the system administrator through e-mail or a file share.

To view your BAM aggregations

1. In Internet Explorer, in the **Address** box, type the URL of the BAS Windows SharePoint Services collaboration site.

2. From the **Quick Launch** navigation pane, in the Business Processes Management area, click **Business Activity Views**.

   Your BAM aggregation workbooks appear in the Business Activity Views window.

For more information about what you can do with the live data workbooks, see Viewing Live BAM Data.

The following figure shows pre-calculated BAM aggregations.

Pre-calculated BAM aggregations
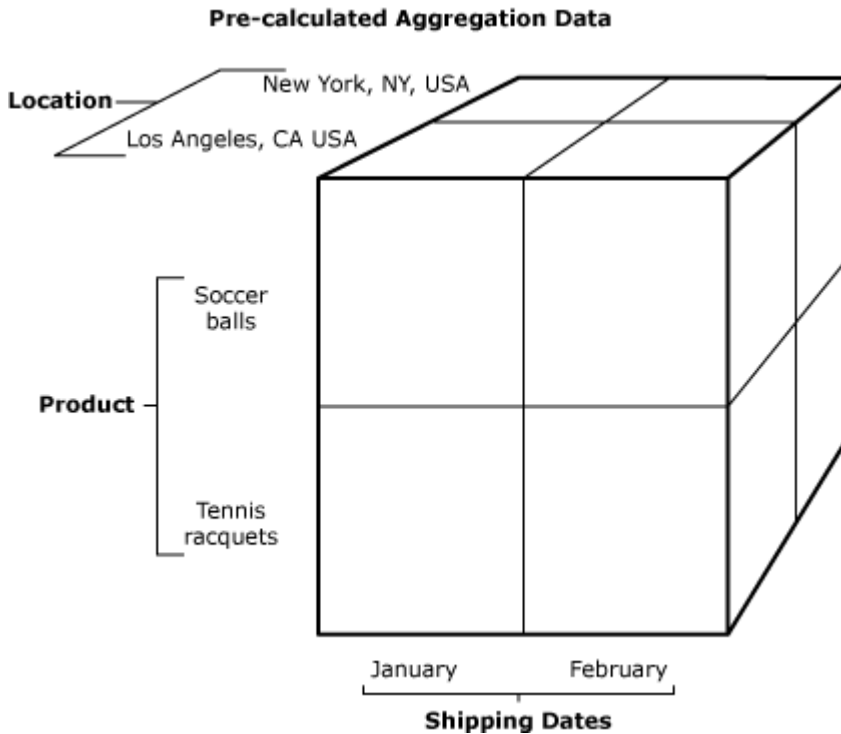
What are Real-time Aggregations

The business analyst can define your pivot table as a real-time aggregation (RTA) or a scheduled aggregation. An RTA gives you an up-to-the-minute view of your data, for example, where a particular PO is in the shipping process. You can refresh your screen to update the view of the data throughout the day.

What are Scheduled Aggregations

A scheduled aggregation represents a snapshot of the business at a specific time, for example, a summary of this morning's shipments. Ask your database administrator when your aggregations are processed, and then you can look at the historical data.

Searching and Navigating Activities

Use the Business Activity Search on the BAS Web portal to search the tracking data, also known as instances, stored in the BAM database.

The search queries against both live data for currently executing processes, as well as stored data.

To search the BAM database

1. On the BAS Windows SharePoint Services collaboration site, click **Business Activity Search** in the Quick Launch pane on the left side of the page.

    The **Business Activities Search** page opens.

2. In the **View Name** drop-down list, select the desired BAM View.

3. In the **Available Activities** list box, select the activity you want to search against and click **Add**, and then click **Next**.

4. In the Available Columns list box, select one or more columns to be displayed in the search results and click **Add**.

5. To change the relative position of the columns, select a column in the **Selected Columns** list box and use the **Move Up** or **Move Down** buttons as appropriate. This is an optional step.

6. Click **Next**.

7. Use the drop-down lists on this page to create filters for the search criteria. Use the **Add Clause** button to add a new blank clause to the end of the query. When all of the clauses are added, click **Finish**.

    The results are displayed on the Business Activity Search Results page.

Viewing an Individual Activity

You view activities in the Business Activity Search Results page of the BAS Windows SharePoint Services collaboration site. For the steps to search for activities, see Searching and Navigating Activities.

To view detailed tracking information

1. After you search for available activities on the BAS Windows SharePoint Services collaboration site, in the Business Activity Search Results page, click the **Activity ID** of the activity you want to view.

    The Business Activity Details window appears.

After you view a single business activity, you can look at any related activities or related documents.

Viewing Related Activities and Related Documents

In the BAS Windows SharePoint Services collaboration site, you can view related activities and documents through the Business Activity Search Results page. You can navigate from one related activity to another, for example, if you search for a specific Purchase Order, you can look at the multiple shipments on the PO. You can also check for related documents, for example, for that same PO, you can look at the related sales order.

For the steps to search for activities, see Searching and Navigating Activities

To view related activities

1.   In the BAS Windows SharePoint Services collaboration site, search for a specific activity.

2.   On the Business Activity Details page, click the plus (+) icons under **Related Activities**.

      The list of related activities appears.

3.   Click a related activity in the list.

To view related documents

1.   In the BAS Windows SharePoint Services collaboration site, search for a specific activity.

2.   In the Business Activity Details page click a related document in the list at the bottom of the window.

Managing Business Activity Monitoring

Business Activity Monitoring (BAM) empowers business users by providing real-time access to critical business performance indicators. Business users can use this information to improve the speed and effectiveness of business operations. BAM collects the business activity information in a real-time, transaction-consistent way regardless of the complexity of a heterogeneous environment.

For more information about BAM, see the topics listed in the following table.

Description	Topic
Role of the business analyst	Business Activity Monitoring Workflow
BAM architecture and concepts	**Business Activity Monitoring (BAM)**
Deploying the BAM Definition Workbook	Deploying BAM Definitions

BAM Definition schema	BAM Configuration Schema
How business users use the BAM Definition Workbook	"Business Activity Monitoring" in the *Information Workers Users Guide*
BAM security	Business Activity Monitoring Security Recommendations
Backing up and restoring BAM	How to Back Up the BAM Analysis and Tracking Analysis Server Databases

This section contains:

- Prerequisites for Using BAM Workbooks

- Managing the BAM Dynamic Infrastructure

- Deploying Tracking Profiles with the BizTalk Tracking Profiles Management Utility

- Managing the Tracking Data Decode Services (TDDS)

- Changing BAM Runtime Settings

BAM Scenarios for Administrators

This topic describes typical scenarios for BAM Administrators.

Configuring BAM

Normal

Configuring BAM Alerts Using the Configuration Manager

Configuring BAM Tools Using the Configuration Manager

Configuring BAM Portal Using the Configuration Manager

Distributed NS

Moving the PIT – Disaster Recovery

Backing Up and Restoring BAM

Configuring 2 BTS Groups to 1 PIT

Working with BAM definitions

BM.Exe in detail

Changing Activities -

Applying indexes to activity tables

And when.. i.e. query analysis via alerts. -

Setting Security on Views -

Distributed Navigation

Exporting Definitions, Importing to Excel,

- Pulling out alerts

- we use Excel + bam.xla to define a BAM workbook, but we can not use Excel + bam.xla to define alerts.

- we can export an Excel BAM workbook to an xml.

- we can define alerts in the xml exported from an Excel BAM workbook

- we can import an xml to Excel. However, if that xml contains alerts, alerts will not be imported to Excel

- continue with scenario 4, if we make some changes to the workbook and then decide to save the change and export the workbook to xml, the alerts will be lost (if you save it with the same name)

- You can use Excel PivotTable feature to define RTA and OLAP table and export to XML for deployment and so on. It will be difficult to manually create RTA and OLAP table directly in XML file.

- Note that when you import XML using Excel, the RTA and OLAP table layout is lost. You need to re-arrange the tables again.

Auditing changes

Backup and restore

Managing Alerts

How to use the Portal as an admin to manage alerts & Subscriptions

You can create, update, and delete alerts using the BAM management utility. For more information using bm.exe to manager alerts see the xxxx sections in

You can also create etc alerts through the bam portal

Scenario, users send ask to create

Prerequisites for Using BAM Workbooks

Business analysts use the BAM Microsoft® Excel template to define the events and data to be tracked, along with the logical groupings and aggregations of the tracked data views that users are able to access.

To provide the Excel template to many users, you can upload this spreadsheet to the Business Activity Services Microsoft Windows® SharePoint® Services collaboration site, and set the appropriate permissions for user access. For more information on using the Windows SharePoint Services collaboration site, see the "BAS Collaboration Site" topic in the *Information Workers User's Guide*. To obtain the user guide, refer to the important note located in <u>Managing Business Activity Monitoring</u>.

BAM Excel Macro Requirements

You can run the BAM macros in the BAM Excel template on any computer, whether BizTalk Server 2006 is installed or not. However, the computer must have the binaries listed in the following table installed for the Excel template macros to work.

DLL name	Version
VBE6.dll	6.3.91.8
Excel.exe	10.0.4524.0
stdole2.tlb	3.50.5014.0
mso.dll	10.0.4219.0
fm20.dll	2.1.7017.1
scrrun.dll	5.6.0.6626
MSCOMCTL.OCX	6.1.95.45
msado15.dll	2.71.9030.0
msxml4.dll	4.0.9406.0 (SP1)

Scheduling the DTS Packages

Users create BAM views based on data stored in an Online Analytical Processing (OLAP) cube. The Cube Update Data Transformation Services (DTS) package refreshes the data in the cube so that OLAP views reflect the correct data.

You must run the BAM Cube Update DTS package at least once for the OLAP views to work. For ongoing maintenance, you should schedule the package to run on a regular basis.

You can schedule a saved Data Transformation Services (DTS) package to execute at specific times, either once or at recurring intervals. For example:

- Daily at 12:00 midnight.

- Weekly on Sunday at 6:00 A.M.

- The first or last day of the month.

A scheduled DTS package is executed by SQL Server Agent as a job.

For more information about running DTS packages, see SQL Server Books Online.

To run the BAM cube update DTS package once

1. Open the Microsoft SQL Server **Enterprise Manager**.

2. Expand the Data Transformation Services node.

3. Right-click the **BAM_AN_<ViewName> Package**, and select **Execute Package**.

To run the Maintaining BAM data DTS package once

1. Open the Microsoft SQL Server **Enterprise Manager**.

2. Expand the Data Transformation Services node.

3. Right-click the **BAM_DM_<activity name> Package**, and select **Execute Package**.

To schedule the BAM DTS packages to run

1. From the **Start** menu, point to **All Programs**, point to **SQL Server**, and then click **SQL Server Enterprise Manager**.

2. In the SQL Server Enterprise Manager console tree, expand **Data Transformation Services**, and then click **Local Package**.

3. In the details pane, right-click either **BAM_AN_<ViewName> Package** or **BAM_DM_<activity name> Package**, and then click **Schedule Package**.

4. In the **Edit Recurring Job Schedule** dialog box, complete the required information.

For more information about creating BAM views, see "Defining a Business Activity View" in the *Information Workers Users Guide*.

Archiving Primary Import Database Data

An administrator can specify the time window for archiving activity instance data in the primary import database. You use the OnlineWindowTimeUnit and OnlineWindowTimeLength properties in the BAM_Metadata_Activities table in the BAMPrimaryImport database.

If business users have deployed multiple activities, you can specify a different time window for each activity. For information about deploying activities, see "Defining a Business Activity" in *Information Workers Users Guide*.

The following table describes the values you can use for OnlineWindowTimeUnit and OnlineWindowTimeLength.

Property	Value
OnlineWindowTimeUnit	This property can be: month, day, hour, or minute. The default value of this property is month.
OnlineWindowTimeLength	This property must be an integer. The default value of this property is 6.

BAM moves data out of the BAM primary import database by partition, when the partition is older than the online window (current time - OnlineWindowTimeLength of OnlineWindowTimeUnit). For example, for OnlineWindowTimeLength = 5 and OnlineWindowTimeUnit = day, partitions older than 5 days are removed.

BAM moves archived activity instance data into the BAM archiving database. You specify the BAM archiving database in the BAMConfiguration.xml file. For information about the BAMConfiguration.xml file, see <u>BAM Configuration Schema</u>.

BAM will not archive activity instance data if you have not run the BAM data maintenance Data Transformation Services (DTS) package, which processes the instance data into the activity cube.

Creating a Partitioned View in the Archiving Database

When you run the BAM data maintenance package (BAM_DM_<activity name>) BAM copies each partition in the Primary Import database to a separate table in the Archiving database. If you detach the archiving database and reattach it for Querying, it will be very difficult to locate the data for your query.

You can create partitioned views in the Archiving database to facilitate locating the data. BAM supports up to 253 partitions. BAM generates one BAM data maintenance DTS package for each activity, which copies the data to the Archiving database and then drops it from the Primary Import database. If the Archiving Database crashes after the data is copied but before the next backup, data is lost.

The solution is to have a single Archiving package, which will first copy the old data from all Activities, then back up the Archiving database, and finally drop the partitions that were copied from the Primary Import database.

Prerequisites

To create a partitioned view in archiving database

1.   Open the SQL Query Analyzer, and connect to the BAM archiving database.

2.   On the **Tools** menu, select **Options**.

3.   In the **Options** dialog box, on the **Results** tab, do the following:

Use this	To do this
Default results target	From the drop-down list, select Results to Text.
Results output format	From the drop-down list, select Tab Delimited.

4.   In a blank query window, type the following SQL script. Replace <activity name> with your activity name, replace <view type> with either Instances for instance view or Relationships for relationship view.

How to Deploy Tracking Profiles with the BizTalk Tracking Profiles Management Utility

A business manager asks a solutions developer to create a new tracking profile or modify an existing one to better manage and monitor a specific business process for your organization.

The solutions developer uses the Tracking Profile Editor (TPE) to define the data that the business analyst requires.

After a solutions developer creates or modifies the tracking profile, an administrator uses the bttdeploy.exe command line utility to deploy it so that the changes take affect and the data is collected. The solutions developer can deploy tracking profiles with the TPE.

To deploy the tracking profile from the command-line utility

1.   From a command prompt, move to the directory <installation path>\Program Files\Microsoft BizTalk Server 2006\Tracking\.

2.   Type **bttdeploy.exe <profile name>.btt**.

3.   Press ENTER.

Managing the Tracking Data Decode Services (TDDS)

The BAM Event Bus Service, also known as the Tracking Data Decode Service (TDDS), processes tracking data (streams) stored in a source database and persists that data in a query table format in the destination database.

The BAM Event Bus service moves event data from the MessageBox database to the BAM Primary Import database. This service processes and persists both Business intelligence and BizTalk Health Monitoring data.

The BAM Event Bus service runs as a Microsoft® Windows NT® Service. You use the BAM Event Bus snap-in to monitor the activities of the BAM Event Bus service.

You monitor the activities of a transactional application, such as Microsoft BizTalk® Server, by collecting event data during execution, and then temporarily storing the data in the same database as the application state—for example, the MessageBox database.

The BAM Event Bus service reads the event data, decodes it, and then stores it in a Microsoft SQL Server™ database, where you can easily query the data. You use the BAM Event Bus service to manage the event data, which provides the following advantages:

- Event data always matches the state of the application, and it never exposes uncommitted progress.

- Performance impact on the running application is minimal because the event data saves as few records in the same local transaction as the application state change.

- SQL Server storage for the application state is further optimized for execution performance. The event data is used to generate the reports, so you do not need to generate the reports from SQL Server.

- The work to store the event data in a form that you can query is not done in the application servers and databases. It is offloaded to the machines that run the BAM Event Bus service and the Destination SQL Server database.

- Event data is in a form you can query with low latency. The BAM Event Bus services coordinate their resources to achieve the minimum possible latency.

The BAM Event Bus server coordinates its resources by using a connection to a central database, which contains the configuration information. Every minute, each active BAM Event Bus service sends a message to the central database, which contains the state of the BAM Event Bus service at that point in time.

This message is referred to as a heartbeat message. Each BAM Event Bus service also checks for new work that needs to be done. For example, the BAM Event Bus service checks for sessions that are not owned, such as a MessageBox database that has been added.

The BAM Event Bus session is the movement of the event data from the source database, such as the MessageBox, to the destination database that contains the event data in a format that you can query. The same BAM Event Bus service can process one or more sessions.

The following figure shows a group of BAM Event Bus servers, which make up a BAM Event Bus server pool.

Diagram of a BAM Event Bus server pool
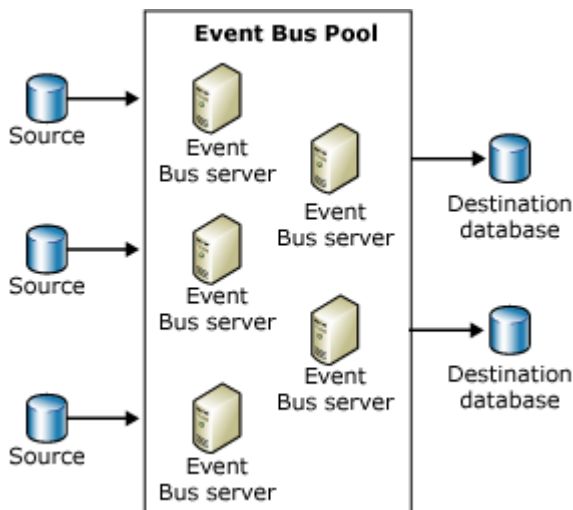
In This Section

- BAM Performance Counters

BAM Performance Counters

Performance counters allow you to monitor specific aspects of work performed by the BAM Event Bus Service. Performance counters can help you identify and troubleshoot server performance issues.

The following table lists the performance counters available in Business Activity Monitoring.

Counter	Description
Events being processed	The number of events the BAM Event Bus Service is currently processing
Batches being processed	The number of batches the BAM Event Bus Service is currently processing
Events Committed	The number of events the BAM Event Bus Service has committed to SQL Server in the last second.
Records	The number of records the BAM Event Bus Service has committed

Committed	to SQL Server in the last second.
Batches Committed	The number of batches the BAM Event Bus Service has committed to SQL Server in the last second.
Total Events	The number of events the BAM Event Bus Service has processed since you started it.
Total Records	Then number of records the BAM Event Bus Service has processed since you started it.
Total Batches	Then number of batches the BAM Event Bus Service has processed since you started it.
Total Failed Batches	Then number of batches the BAM Event Bus Service has failed to process since you started it.
Total Failed Events	Then number of events the BAM Event Bus Service has failed to process since you started it.

The performance counters are located in the Performance Monitor (perfmon) under the BizTalk:TDDS performance object. The instance name of the performance counters are a combination of the source server name and the name of the source database. If two of the BAM Event Bus Services are running on the same computer against two different source databases, you will see two instances of the BAM Event Bus Service counters.

Changing BAM Runtime Settings

In general, you configure the BAM Runtime components when you install Microsoft® BizTalk® Server 2006. You may want to change the BAM Runtime settings, for example to scale up your system or if you have password expiration policy.

This section contains:

- BAM Configuration Schema

- **Managing the BAM Query Web Service**

- Updating OLAP and Real Time Aggregation Connection String Properties

BAM Configuration Schema

The BAM configuration schema defines an XML document that contains information about your infrastructure that the BAM Manager Utility uses for deployment. You can deploy your databases to multiple servers for scalability. To support this scalability, ensure that the BAM configuration XML document contains the different server names and configuration settings for the following databases:

- BAMPrimaryImport

- BAMStarSchema

- BAMAnalysis

- BAMArchive

In This Section

- BAM DTS Packages

- Defining the Time Window and Time Slice Properties

- Modifying the BAM Configuration Schema

BAM DTS Packages

An administrator can update parameters for the following BAM DTS packages:

- The **CubeUpdate** Data Transformation Services (DTS) package is always located on the same server as the Star Schema database.

- The **DataMaintenance** DTS package is always located on the same server as the Primary Import database.

The DTS packages use the following parameters in the BAMConfiguration.xml file.

Parameter	Description
ConnectionTimeOut	The DTS connection time out value (in seconds) is an integer. If you omit the ConnectionTimeOut parameter, the configuration file uses 60 seconds, the default value.
Encryption	By default, the DTS packages do not encrypt data while they transform the data (Encryption value is 0). Set Encryption to 1 to encrypt the data during transformation.
OwnerPassword	The password for the DTS package owner. DTS package owners can open and modify DTS packages. For information about DTS package owners, see SQL Server Books Online.
UserPassword	The password for the DTS user. DTS package users can run DTS packages. For information about DTS package users, see SQL Server Books Online.

The DTS packages use the following naming conventions in the BAMConfiguration.xml file:

- **CubeUpdate** DTS package

  bam_AN_<*CubeName*>, where CubeName is the name of the cube. The BAM workbook generates the cube name from the view name. If you modify the cube name in the BAM configuration XML document, the new cube name is used in the DTS package name.

- **DataMaintenance** DTS package

  bam_DM_<*ActivityName*>, where ActivityName is the name of the activity.

You run the CubeUpdate DTS package to aggregate the scheduled aggregation. In the next section, you can specify the time window for real-time data aggregation.

Defining the Time Window and Time Slice Properties

Administrators use the TimeWindow and the TimeSlice properties in the BAMConfiguration.xml file to define the life of the data in the real-time aggregation tables in the BAM primary import database.

For information about undeploying a BAM definition, see Undeploying BAM Definitions. For information about deploying a BAM definition, see Deploying BAM Definitions.

If you want to change the TimeWindow and TimeSlice values without undeploying the BAM infrastructure, you can modifying the columns in the BAM_Metadata_Activities table in the BAM primary import database.

TimeSlice

You use the TimeSlice property to group completed BAM instance data. The TimeSlice property uses time that the data is written to the BAM primary import database to group BAM instance data.

For example, instance A is completed and persisted in the BAM primary import database at 1/1/2000 1:02 a.m. Instance B is completed and persisted in the BAM primary import database at 1/1/2000 1:04 a.m. If you set the value of the TimeSlice property to five minutes, instance A and instance B are grouped together.

To change the TimeSlice value in the BAM Configuration file

1.   Change the value in this line of the BAM Configuration file:

To change the TimeSlice value in the BAM_Metadata_Activities table

1.   Modify the RTATimeSlice values, located in the bam_Metadata_Activities table in the BAMPrimaryImport database. If you deploy multiple real-time aggregations (RTA) for one or more activities, you have the option to specify a different time window for each RTA.

TimeWindow

When you run the CubeUpdate DTS package, the package moves data from the BAM primary import database into the BAM cubes. The package moves real-time aggregation data as grouped BAM data instances after all of the data in the group is older than the age specified in the RTA window property.

To change the TimeWindow value in the BAM Configuration file

1.   Change the value in this line of the BAM Configuration file:

To change the TimeWindow value in the BAM_Metadata_Activities table

1.   Modify the RTAWindow values, located in the bam_Metadata_Activities table in the BAMPrimaryImport database. If you deploy multiple real-time aggregations (RTA) for one or more activities, you have the option to specify a different time window for each RTA.

Modifying the BAM Configuration Schema

The Configuration Wizard creates this configuration file automatically. You must modify this file manually if you change your server names or other configuration information after you complete the deployment.

For information about undeploying a BAM definition, see Undeploying BAM Definitions. For information about deploying a BAM definition, see Deploying BAM Definitions.

Updating OLAP and Real Time Aggregation Connection String Properties

For information about updating the OLAP and Real Time Aggregation connection strings, see the following topics that describe restoring BAM databases to alternate computers:

•   How to Update References to the BAM Primary Import Database Name and Connection String

•   How to Update References to the BAM Analysis Server Database Name

Business Activity Monitoring Security Recommendations

With Business Activity Monitoring (BAM), information workers can have visibility into the business Process. For more information about Business Activity Monitoring, see Managing Business Activity Monitoring. It is recommended you follow these guidelines for securing and deploying BAM in your environment.

- To deploy BAM infrastructure, the administration computer must have SQL Server client tools, Decision Support Objects (DSO) tools, SQL Server 2000 SP3 or higher, and Analysis Server 2000 SP3 or higher installed.

- Multiple users need access to the live and archived data: salespeople, business managers, and others. These users must have domain accounts in the domain where the BAM Primary Import and BAM Analysis databases are (corporate domain), but their accounts do not have access to BizTalk resources.

- In order for users to access views of the BAM data, you must use the BAM Manager Utility (BM.exe) to grant the users permissions to the views, and the users must have SQL logins. For more information about the BAM Manager utility, see BAM Management Utility.

- In order to add users to roles that have access to the cubes in the BAM Analysis database, you must have an administration computer on the same domain as the BAM databases.

- The person administering BAM must be db_owner for the BAM Primary Import, Star Schema, and BAM Archive databases. They must also be an OLAP administrator for the BAM Analysis database.

- If you are deploying Excel workbooks (.xls), you need to have Excel on the administration computer. Before you deploy a workbook, open it and ensure the macro security is set to high, and that there are no warnings.

- If there is no business need to distribute to the users BAM Excel workbooks that connect to the real data, it is recommended that you deploy the workbooks by using XML files. This would eliminate the need to install Excel on the administration computer, and the need to verify the macro security.

Managing the BAM Dynamic Infrastructure

Business Activity Monitoring (BAM) features use a dynamically generated SQL and online analytical processing (OLAP) infrastructure. Administrators use the BAM Management utility to deploy the BAM definition workbook or XML file, which the business analyst develops.

The BAM dynamic infrastructure consists of the BAM definition workbook, BAM deployments, the BAM Data Transformation Services (DTS) packages, and the BAM databases. For more information about the BAM dynamic infrastructure, see BAM Dynamic Infrastructure .

Administrators perform the following management tasks for the BAM infrastructure, which are described in this section:

- Deploy and undeploy BAM definitions and views

- Manage user access to BAM views

- Run the BAM DTS packages

- Back up the BAM databases

In This Section

- Managing BAM Definitions

- Managing BAM Security

- Managing Aggregations

Managing BAM Definitions

A BAM definition is part of the BAM infrastructure. It defines the data to track and aggregate, as well as the business end user's view on the tracking data. The topics in this section give procedures for managing the elements of BAM definitions, which include activities, views, artifacts, and alerts.

In This Section

- How to Deploy BAM Definitions

- How to Undeploy BAM Definitions

- How to List Changes to the BAM Infrastructure

- How to List BAM Activities

- How to Remove BAM Activities

- How to List BAM Views

- How to Remove BAM Views

- How to Enable Alerts

- How to Disable Alerts

- How to Remove BAM Alerts

- How to Update BAM Artifacts

- How to Remove Deployed Artifacts

How to Deploy BAM Definitions

Administrators use the **deploy-all** BAM Management utility command to deploy a BAM definition from the Excel workbook or the XML definitions file exported from the workbook. When you perform a complete installation of BizTalk Server 2006, the Configuration Wizard automatically configures the BAM Configuration XML file.

Before you can deploy a BAM definition XML file, you must ensure that the locale used to create this file matches the locale of the computer on which you are deploying it. For example, if you create the file on a computer running a Japanese version of Microsoft Windows, the computer you deploy the file on must be set to the Japanese locale. If the file and the computer settings do not match, you must switch the computer you use to run the BAM Management utility to the correct locale and you must restart it before running the utility.

For information about changing locale settings on your computer, see the Help for your operating system.

To deploy a BAM definition

1.   Open a command prompt as follows: Click **Start**, click **Run**, type **cmd**, and then click **OK**.

2.   Navigate to the tracking folder by typing **C:\Program Files\Microsoft BizTalk Server 2006\Tracking** at the command prompt. Press **ENTER**.

3.   Type **bm deploy-all -DefinitionFile:<BAM definition file>**.

4.   Press **ENTER**.

How to Undeploy BAM Definitions

Administrators use the **remove-all** command of the BAM Management utility to remove all views and underlying activity tables for a particular BAM definition file. You select the BAM definition file or specify the listchange number for which you want to remove all views and activity tables with its number.

To undeploy using the BAM definition

1.   Open a command prompt as follows: Click **Start**, click **Run**, type **cmd**, and then click **OK**.

2.   Navigate to the tracking folder by typing **C:\Program Files\Microsoft BizTalk Server 2006\Tracking** at the command prompt. Press **ENTER**.

3.   Type **bm remove-all DefinitionFile:<def file>**.

4.   Press **ENTER**.

How to List Changes to the BAM Infrastructure

Administrators use the **get-changes** command of the BAM Management utility to list current information about a deployed BAM definition. You can also use the get-changes command to list current deployment artifacts in the BAM Primary Import database.

The information includes successful deployments and undeployments, the name of the BAM definition file, the name of the BAM configuration file, the names of all views associated with the definition file, and the names of all Windows groups and user accounts that have permission to the views associated with the BAM definition. The get-changes list shows deployments and undeployments with their associated identification numbers.

To list changes to the BAM definition

1.   Open a command prompt as follows: Click **Start**, click **Run**, type **cmd**, and then click **OK**.

2.   Navigate to the tracking folder by typing **C:\Program Files\Microsoft BizTalk Server 2006\Tracking** at the command prompt. Press **ENTER**.

3.   Type **bm get-changes.**

4.   Press **ENTER**.

How to List BAM Activities

Administrators use the **get-activities** command of the BAM Management utility to list the activities in the local BAM Primary Import database.

Prerequisites

To list BAM Activities

1.   Open a command prompt as follows: Click **Start**, click **Run**, type **cmd**, and then click **OK**.

2.   Navigate to the tracking folder by typing **C:\Program Files\Microsoft BizTalk Server 2006\Tracking** at the command prompt. Press **ENTER**.

3.   Type **bm get-activities.**

4.   Press **ENTER**.

How to Remove BAM Activities

Administrators use the **remove-activity** command to remove the specified activity from the BAM Primary Import database.

To remove a BAM activity

1.   Open a command prompt as follows: Click **Start**, click **Run**, type **cmd**, and then click **OK**.

2.   Navigate to the tracking folder by typing **C:\Program Files\Microsoft BizTalk Server 2006\Tracking** at the command prompt. Press **ENTER**.

3.   Type **bm remove-activity -Name:<activity name>**.

4.   Press **ENTER**.

How to List BAM Views

Administrators use the **get-views** command to list the BAM views deployed in the BAM Primary Import database on the computer on which the command is run.

To list BAM views

1.   Open a command prompt as follows: Click **Start**, click **Run**, type **cmd**, and then click **OK**.

2.   Navigate to the tracking folder by typing **C:\Program Files\Microsoft BizTalk Server 2006\Tracking** at the command prompt. Press **ENTER**.

3.   Type **bm get-views**.

4.   Press **ENTER**.

How to Remove BAM Views

Administrators use the **remove-view** command to remove a view from the BAM Primary Import database.

To remove BAM views

1.   Open a command prompt as follows: Click **Start**, click **Run**, type **cmd**, and then click **OK**.

2.   Navigate to the tracking folder by typing **C:\Program Files\Microsoft BizTalk Server 2006\Tracking** at the command prompt. Press **ENTER**.

3.   Type **bm remove-view -Name:<view name>**.

4.   Press **ENTER**.

How to Enable Alerts

Administrators use the **enable-alerts** command to enable all of the alerts on the specified view.

To enable an alert

1.   Open a command prompt as follows: Click **Start**, click **Run**, type **cmd**, and then click **OK**.

2.   Navigate to the tracking folder by typing **C:\Program Files\Microsoft BizTalk Server 2006\Tracking at the command prompt**. Press **ENTER**.

3.   Type **bm enable-alerts -View:<view name>**.

4.   Press **ENTER**.

How to Disable Alerts

Administrators use the **disable-alerts** command to disable all of the alerts on the specified view.

To disable an alert

1.   Open a command prompt as follows: Click **Start**, click **Run**, type **cmd**, and then click **OK**.

2.   Navigate to the tracking folder by typing **C:\Program Files\Microsoft BizTalk Server 2006\Tracking at the command prompt**. Press **ENTER**.

3.   Type **bm disable-alerts -View:<view name>**.

4.   Press **ENTER**.

How to Remove BAM Alerts

Administrators use the **remove-alerts** command to remove all the alerts from a specified view.

To remove BAM alerts

1.   Open a command prompt as follows: Click **Start**, click **Run**, type **cmd**, and then click **OK**.

2.   Navigate to the tracking folder by typing **C:\Program Files\Microsoft BizTalk Server 2006\Trackin**g at the command prompt. Press **ENTER**.

3.   Type **bm remove-alerts -View:<view name>**.

4.   Press **ENTER**.

How to Update BAM Artifacts

Administrators use the **update-all** command to update artifacts deployed in the BAM Primary Import database. The supplied definition is either an XML file or an Excel Workbook containing information about the artifacts to be updated.

To update BAM Artifacts

1.   Open a command prompt as follows: Click **Start**, click **Run**, type **cmd**, and then click **OK**.

2.   Navigate to the tracking folder by typing **C:\Program Files\Microsoft BizTalk Server 2006\Tracking** at the command prompt. Press **ENTER**.

3.   Type **bm update-all -DefinitionFile:<def file>**.

4.   Press **ENTER**.

How to Remove Deployed Artifacts

Administrators use the **remove-all** command to remove artifacts deployed in the BAM Primary Import database. The supplied definition is either an XML file or an Excel Workbook containing information about the artifacts to be updated.

To remove deployed artifacts

1.   Open a command prompt as follows: Click **Start**, click **Run**, type **cmd**, and then click **OK**.

2.   Navigate to the tracking folder by typing **C:\Program Files\Microsoft BizTalk Server 2006\Tracking** at the command prompt. Press **ENTER**.

3.   Type **bm remove-all DefinitionFile:<def file>**.

4.   Press **ENTER**.

Managing BAM Security

This section provides information about BAM security. It includes procedures for managing user accounts for BAM views and managing subscriptions to alerts.

For information about security for the BAM portal, see Security Considerations for the BAM Portal.

For information about security for the Tracking Profile Editor (TPE), see Security Considerations for Tracking Profile Editor.

In This Section

- How to List Accounts in a View Role

- How to Add Accounts to a View Role

- How to Remove Accounts from a View Role

- How to List Subscribers to an Alert

- How to Add Subscribers to an Alert

- How to Remove Subscribers from an Alert

How to List Accounts in a View Role

Administrators use the **get-accounts** BAM Management utility command to get a list all user accounts for a view role, meaning all user accounts with access to the specified view.

For information about viewing BAM views, see How to List BAM Views.

To list all members with access to a view

1.   Open a command prompt as follows: Click **Start**, click **Run**, type **cmd**, and then click **OK**.

2.   Navigate to the tracking folder by typing **C:\Program Files\Microsoft BizTalk Server 2006\Tracking** at the command prompt. Press **ENTER**.

3.   Type **bm get-accounts -View:<view name>**.

4.   Press **ENTER**.

How to Add Accounts to a View Role

Administrators use the **add-account** command to associate users with BAM views and protect BAM Excel Spreadsheet views from unauthorized access. When users save BAM views, the views reference a SQL connection string that is hidden within the workbook. The workbook is protected, but you must ensure that the document is protected.

When you associate users with BAM views, you restrict access to the views to only the users or groups to whom you grant access.

For information about viewing BAM views, see **How to List BAM Views**.

To add an account to a view role

1. Open a command prompt as follows: Click **Start**, click **Run**, type **cmd**, and then click **OK**.

2. Navigate to the tracking folder by typing **C:\Program Files\Microsoft BizTalk Server 2006\Tracking at the command prompt**. Press **ENTER**.

3. Type **bm add-account -AccountName:<account name> -View:<view name>**.

4. Press **ENTER**

How to Remove Accounts from a View Role

Administrators use the **remove-account** command to disassociate users from BAM views and protect BAM Excel Spreadsheet views from unauthorized access.

For information about viewing BAM views, see **How to List BAM Views**.

To remove an account from a role

1. Open a command prompt as follows: Click **Start**, click **Run**, type **cmd**, and then click **OK**.

2. Navigate to the tracking folder by typing **C:\Program Files\Microsoft BizTalk Server 2006\Tracking at the command prompt**. Press **ENTER**.

3. Type **bm remove-account -AccountName:<account name> -View:<view name>.**

4. Press **ENTER**.

How to List Subscribers to an Alert

Administrators use the **get-subscriptions** command to list all of the subscribers to a specified alert.

To list subscribers to an alert

1. Open a command prompt as follows: Click **Start**, click **Run**, type **cmd**, and then click **OK**.

2. Navigate to the tracking folder by typing C:\Program Files\Microsoft BizTalk Server 2006\Tracking at the command prompt. Press **ENTER**.

3. Type **bm get-subscriptions -View:<view name> -Alert:<alert name>**.

4. Press **ENTER**.

How to Add Subscribers to an Alert

Administrators use the **add-subscriptions** command to add a subscriber to a specified alert.

To add subscribers to an alert

1.   Open a command prompt as follows: Click **Start**, click **Run**, type **cmd**, and then click **OK**.

2.   Navigate to the tracking folder by typing C:\Program Files\Microsoft BizTalk Server 2006\Tracking at the command prompt. Press **ENTER**.

3.   Type **bm add-subscription -View:<view name> -Alert:<alert name> - AccountName:<account name> -Type: [File | Email][-Email:<e-mail address>]**.

How to Remove Subscribers from an Alert

Administrators use the **remove-subscriptions** command to remove the specified user as a subscriber from an alert.

To remove subscribers from an alert

1.   Open a command prompt as follows: Click **Start**, click **Run**, type **cmd**, and then click **OK**.

2.   Navigate to the tracking folder by typing **C:\Program Files\Microsoft BizTalk Server 2006\Tracking at the command prompt**. Press **ENTER**.

3.   Type **bm remove-subscription -View:<view name> -Alert:<alert name> -AccountName:<account name>**.

4.   Press **ENTER**.

Managing Aggregations

Aggregations are cumulative operations against a data set to derive a generally representative characteristic or description for that data set. This section describes how to manage different elements of an aggregation, such as indexes, activities, and real-time aggregations (RTAs).

For general information about aggregations, see What Is an Aggregation? and Aggregations on the BAM Portal Page.

In This Section

•   How to Create an Index

•   How to Get the Indexes on an Aggregation

- How to Delete an Index

- How to Set the Duration on an Activity Window

- How to Get the Duration on an Activity Window

- How to Set the Duration on an RTA Window

- How to Get the Duration on an RTA Window

- Defining BAM Aggregations

How to Create an Index

Administrators use the **create-index** command to create an index on the specified activity at the specified checkpoints.

To create an index on an activity

1. From a command prompt, browse to the following directory: <installation path>\Program Files\Microsoft BizTalk Server 2006\Tracking\.

2. Type **bm create-index -IndexName:<index name> -Activity:<activity name> -Checkpoint:<checkpoint1>**.

3. Press **ENTER**.

How to Get the Indexes on an Aggregation

Administrators use the **get-index** command to get a list of all the indexes on the specified activity.

To get the a list of indexes on an aggregation

1. Open a command prompt as follows: Click **Start**, click **Run**, type **cmd**, and then click **OK**.

2. Navigate to the tracking folder by typing **C:\Program Files\Microsoft BizTalk Server 2006\Tracking at the command prompt**. Press **ENTER**.

3. Type **bm get-index -Activity:<activity name>**.

4. Press **ENTER**.

How to Delete an Index

Administrators use the **create-index** command to create an index on the specified activity at the specified checkpoints.

To delete an index on an activity

1.   Open a command prompt as follows: Click **Start**, click **Run**, type **cmd**, and then click **OK**.

2.   Navigate to the tracking folder by typing **C:\Program Files\Microsoft BizTalk Server 2006\Tracking at the command prompt**. Press **ENTER**.

3.   Type **bm create-index -IndexName:<index name> -Activity:<activity name> -Checkpoint:<checkpoint1>**.

4.   Press **ENTER**.

How to Set the Duration on an Activity Window

Administrators use the **set-activitywindow** command to set the duration for the specified activity.

To set the duration on an activity

1.   Open a command prompt as follows: Click **Start**, click **Run**, type **cmd**, and then click **OK**.

2.   Navigate to the tracking folder by typing **C:\Program Files\Microsoft BizTalk Server 2006\Tracking at the command prompt**. Press **ENTER**.

3.   Type **bm set-activitywindow -Activity:<activity name> -TimeLength:<integer number> -TimeUnit:Month|Day|Hour|Minute**.

4.   Press **ENTER**.

How to Get the Duration on an Activity Window

Administrators use the **get-rtawindow** command to get the duration for the specified real-time aggregation (RTA). The command returns the length of the duration and the units by which the duration is measured.

To get the duration on an activity

1.   Open a command prompt as follows: Click **Start**, click **Run**, type **cmd**, and then click **OK**.

2.   Navigate to the tracking folder by typing **C:\Program Files\Microsoft BizTalk Server 2006\Tracking at the command prompt**. Press **ENTER**.

3.   Type **bm get-rtawindow -View:<view name> -Activity:<activity name> -Rta:<RTA name>**.

4.   Press **ENTER**.

How to Set the Duration on an RTA Window

Administrators use the **set-rtawindow** command to set the duration for the specified real-time aggregation (RTA).

To set the duration on an aggregation

1.   Open a command prompt as follows: Click **Start**, click **Run**, type **cmd**, and then click **OK**.

2.   Navigate to the tracking folder by typing **C:\Program Files\Microsoft BizTalk Server 2006\Tracking at the command prompt**. Press **ENTER**.

3.   Type **bm set-rtawindow -View:<view name> -Activity:<activity name> -Name:<RTA name> -TimeLength:<integer number>-TimeUnit:Day|Hour|Minute**.

4.   Press **ENTER**.

How to Get the Duration on an RTA Window

Administrators use the **get-rtawindow** command to get the duration for the specified real-time aggregation (RTA). The command returns the length of the duration and the units by which the duration is measured.

To get the duration on an aggregation

1.   Open a command prompt as follows: Click **Start**, click **Run**, type **cmd**, and then click **OK**.

2.   Navigate to the tracking folder by typing **C:\Program Files\Microsoft BizTalk Server 2006\Tracking at the command prompt**. Press **ENTER**.

3.   Type **bm get-rtawindow -View:<view name> -Activity:<activity name> -Rta:<RTA name>**.

4.   Press **ENTER**.

Defining BAM Aggregations

BAM supports two types of data aggregation:

•   Online analytical processing (OLAP) aggregations

•   Real-time aggregations (RTA)

BAM uses Microsoft Analysis Service to implement OLAP aggregations.

You must configure the triggers on the BAM Primary Import database that define RTA.

To define OLAP aggregations

1. In the BAM Excel workbook, create a view, add at least one dimension and one measure to the PivotTable report, clear the RTA toolbar button, and then save the workbook.

    -   For information about opening the BAM workbook, creating a view, and adding dimensions and measures, see "Defining a Business Activity View" and "Defining Aggregations" in the *Information Workers User Guide*.

2. Deploy the workbook.

    -   To deploy the workbook, follow the instructions in <u>Deploying BAM Definitions</u>.

3. A solutions developer uses the DirectEventStream class to import events into the BAM Primary Import database.

    -   For information about the DirectEventStream class, see **DirectEventStream Class**.

4. Run the update cube Data Transformation Services (DTS) package.

    -   For information about running the update cube DTS package, see <u>Scheduling the DTS Packages</u>.

5. Open the most recent live data copy of the workbook to see the OLAP aggregations.

To define the RTA

1. In the BAM Excel workbook, create a view, add at least one dimension and one measure to the PivotTable report, select the RTA toolbar button, and then save the workbook.

    -   For information about opening the BAM workbook, creating a view, and adding dimensions and measures, see "Defining a Business Activity View" and "Defining Aggregations" in the *Information Workers User Guide*.

2. Deploy the workbook.

    -   To deploy the workbook, follow the instructions in <u>Deploying BAM Definitions</u>.

3. A solutions developer uses the DirectEventStream Class imports events into the BAM Primary Import database.

- For information about the DirectEventStream Class, see **DirectEventStream Class**.

4.   Open the most recent live data copy of the workbook to see the RTAs.

Managing BAM Databases

Administrators use the BAM Management utility (bm.exe) to set up, manage, and update the BAM databases. This section shows you how to use the BAM Management utility to perform these common administrator tasks for the BAM databases, which are described in the following table.

For information about backing up and restoring the BAM databases, see Backing Up and Restoring BAM.

Database	Default database name	Description
BAM Primary Import database	BAMPrimaryImport	Where BAM collects raw tracking data.
BAM Notification Services Application database	BAMAlertsApplication	Contains alert information for BAM notifications. For example, when you create an alert using the BAM portal, entries are inserted in the database specifying the conditions and events to which the alert pertains, as well as other supporting data items for the alert.
BAM Notification Services Instance database	BAMAlertsNSMain	Contains instance information specifying how the notification services connect to the system that BAM is monitoring.
BAM Star Schema database	BAMStarSchema	Contains the staging table, and the measure and dimension tables.
BAM Analysis database	BAMAnalysis	Contains BAM OLAP cubes for both online and offline analysis.
BAM Archive database	BAMArchive	Archives old business activity data. You can create a BAM Archive database to minimize the accumulation of business activity data in the BAM Primary Import database.

In This Section

- How to Set Up the BAM Databases Using the BAM Management Utility

- How to Retrieve the BAM Configuration File Using the BAM Management Utility

- How to Update the BAM Configuration Using the BAM Management Utility

- How to Reference Additional BAM Primary Import Databases

- How to Disable a BAM Primary Import Database Reference

- How to List All Referenced Databases

How to Set Up the BAM Databases Using the BAM Management Utility

Administrators typically use the BizTalk Server configuration utility to set up the BAM databases. You can use the BAM Management utility (bm.exe) as an alternate method to set up the databases.

Prerequisites

The following are prerequisites for performing the procedure in this topic:

- You must have Administrator permissions on the SQL server on which you are setting up the databases.

- You must have a BAM configuration file containing XML data from which to set up the file.

To set up the BAM databases using the BAM Management utility

1. Open a command prompt as follows: Click **Start**, click **Run**, type **cmd**, and then click **OK**.

2. Navigate to the tracking folder by typing **C:\Program Files\Microsoft BizTalk Server 2006\Tracking** at the command prompt. Press **ENTER**.

3. Type the following at the command line prompt: **bm setup-databases-ConfigFile:<configuration file>**, where *<configuration file>* is replaced by the name of your BAM configuration file. Press **ENTER**.

How to Retrieve the BAM Configuration File Using the BAM Management Utility

Administrators and developers can use the BAM Management utility to retrieve the current configuration of the BAM infrastructure. The retrieved configuration can be used to migrate a BAM installation to a new server or it can be modified and used to update an existing BAM installation.

Prerequisites

The following are prerequisites for performing the procedure in this topic:

- Configured BAM databases

- Permissions to read the BAM Primary Import database

To retrieve the BAM configuration file using the BAM Management utility

1. Open a command prompt as follows: Click **Start**, click **Run**, type **cmd**, and then click **OK**.

2. Navigate to the tracking folder by typing **C:\Program Files\Microsoft BizTalk Server 2006\Tracking** at the command prompt. Press **ENTER**.

3. Type the following at the command line prompt: **bm get-config -FileName:<output file>**, where *<output file>* is replaced by the name of your BAM configuration file. Press **ENTER**.

How to Update the BAM Configuration Using the BAM Management Utility

Administrators can use the BAM Management utility to update an existing BAM installation.

Prerequisites

You must have Administrator permissions on the BAM database being updated.

To update the BAM configuration using the BAM Management utility

1. Open a command prompt as follows: Click **Start**, click **Run**, type **cmd**, and then click **OK**.

2. Navigate to the tracking folder by typing **C:\Program Files\Microsoft BizTalk Server 2006\Tracking** at the command prompt. Press **ENTER**.

3. Type the following at the command line prompt: **bm update-config -FileName:<config file>**, where *<configuration file>* is replaced by the name of your BAM configuration file. Press **ENTER**.

How to Reference Additional BAM Primary Import Databases

Administrators use the **enable-reference** command to reference additional BAM Primary Import databases. You reference multiple BAM Primary Import databases to facilitate viewing distributed BAM activities.

Prerequisites

To enable a reference to an additional BAM Primary Import database

1.   Open a command prompt as follows: Click **Start**, click **Run**, type **cmd**, and then click **OK**.

2.   Navigate to the tracking folder by typing **C:\Program Files\Microsoft BizTalk Server 2006\Tracking** at the command prompt. Press **ENTER**.

3.   Type the following at the command line prompt: **bm enable-reference -TargetServer:<target server> -TargetDatabase:<target database>**, where *<target server>* is replaced by the name of the SQL server on which the target BAM Primary Import database specified by *<***target database***>* resides. Press **ENTER**.

How to Disable a BAM Primary Import Database Reference

Administrators use the **disable-reference** command to disable a reference to a specified BAM Primary Import database.

To disable a reference to a BAM Primary Import database

1.   Open a command prompt as follows: Click **Start**, click **Run**, type **cmd**, and then click **OK**.

2.   Navigate to the tracking folder by typing **C:\Program Files\Microsoft BizTalk Server 2006\Tracking** at the command prompt. Press **ENTER**.

3.   Type the following at the command line prompt: **disable-reference -TargetServer:<target server> -TargetDatabase:<target database>[-Server:<server>][-Database:<database>]**, where *<target server>* is replaced by the name of the SQL server on which the target BAM Primary Import database specified by *<target database>* resides. Press **ENTER**.

How to List All Referenced Databases

Administrators use the **get-references** command to list all of the BAM Primary Import databases on other BizTalk servers that have been given references to the local BAM Primary Import database.

Prerequisites

To list all referenced databases

1.   Open a command prompt as follows: Click **Start**, click **Run**, type **cmd**, and then click **OK**.

2.   Navigate to the tracking folder by typing **C:\Program Files\Microsoft BizTalk Server 2006\Tracking** at the command prompt. Press **ENTER**.

3.   Type the following at the command line prompt: **bm.exe get-references**. Press **ENTER**.

Human Workflow Services [BPM]

This section provides information about creating and managing Human Workflow Services (HWS) enabled applications.

In This Section

- What Is Human Workflow Services (HWS)?

- Managing Human Workflow Services

- Using the HWS Administration Console

- Developing Human Workflow Services Solutions

What Is Human Workflow Services (HWS)?

The Microsoft® BizTalk® Server 2006 engine enables you to connect applications to carry out a business process. Many processes—for example, handling a purchase order or responding to a request for proposal—require human intervention. Human-oriented business processes like these are commonly called "workflow." BizTalk Server 2006 can help to automate workflow, but more is required to enable people to interact with and control a workflow process.

Human Workflow Services (HWS) is a standard part of BizTalk Server 2006. HWS provides a workflow infrastructure built on the BizTalk Server 2006 engine. You access the HWS infrastructure by using Web services, so it can be used by any client application. The applications in Microsoft Office are among the most important clients that use HWS. Many information workers use Microsoft Word, Microsoft Outlook®, Microsoft Excel, and Microsoft Office InfoPath™, and so it makes sense to allow these common tools to be the environment from which people participate in workflows.

The following figure shows the basic model.

HWS workflow model
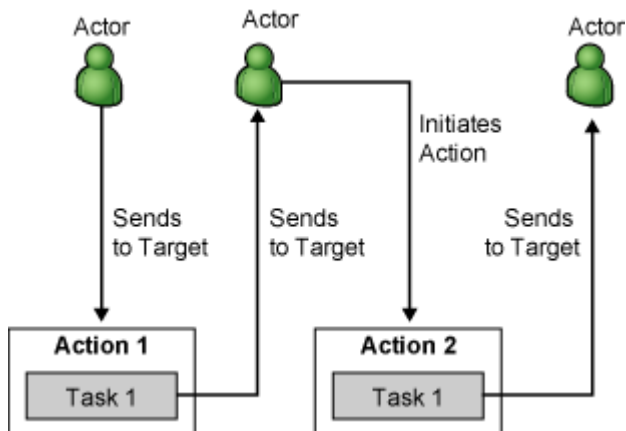
Activity flows are composed of one or more actions. An action represents a fundamental business process, or unit of workflow, that you cannot reduce to further sub-actions. Actions may contain zero or more tasks defining work.

Actions do not always send tasks. In dependent composition for example, actions send synchronization message and no tasks. A synchronization message is a message that an action sends or receives to or from another action when the action runs. For more information about dependent action composition, see Creating Dependent Actions. You assign tasks to the participants (called actors) in the workflow, which are external entities that either start an activity flow or participate in an ongoing activity flow. The actor who initiates the first action within an activity flow becomes the owner of the activity flow.

In This Section

- HWS Scenario: Health Risk Assessment

- HWS Scenario: Proposal Review

- HWS Workflow Model

- HWS Architecture

- HWS Components

- HWS User Roles

HWS Scenario: Health Risk Assessment

The scenario described in this section is an example of a health risk assessment workflow. It illustrates the use of the Human Workflow Services (HWS) Web service in BizTalk Server 2006 to implement this business process as an HWS activity flow or workflow.

This workflow is composed of both automated tasks and tasks that require human action. Users create the activity flow as they add tasks at run time.

Each step in the activity flow is a task that a user performs. Users in activity flows are called the target actors of each task.

An HWS implementation of a health risk assessment might contain the following steps:

1.   The applicant, the first workflow participant, initiates the workflow by filling in a   new   assessment   form.

       The only action the applicant can perform is to assign the completed assessment form to the HR person for review.

2.   The applicant assigns the completed form to the HR person for review.

3.   The assign action triggers the review task assigned to the HR person.

4.   HWS   sends   the   HR   person   a   notification   about   the   task.

       The notification is an e-mail message that contains instructions for completing the task and a link to a Web site that displays the assessment form.

5.   The HR person navigates to the Web site and reviews the assessment.

       HWS provides three actions that the HR person can perform.

       •   **Accept** the assessment, which will complete the activity flow.

              This will send a special message, called the task message, to HWS indicating that the task is complete, for example, setting the task status to complete."

       •   **Reject** the assessment, which will complete the activity flow.

              This will send a task message to HWS indicating that the task is complete.

       •   **Delegate** the task. For example, the HR person delegates the review to an external consultant.

              Delegating the task sets the task status to deferred.

              The HR person uses the delegate action to pass the task to the external consultant.

              The delegate action does not create a new task. The delegate action accepts the task triggered by the assign action as a parameter.

              The external consultant can accept the assessment or reject the assessment.

The external consultant uses an HWS task message to tell the HWS system about the new status of the activity flow.

For another example of how you can apply this implementation model to a real-life business situation,

HWS Scenario: Proposal Review

The scenario described in this section is an example of a workflow to automate a proposal review process. It illustrates the use of the Human Workflow Services (HWS) Web service in BizTalk Server 2006 to implement this business process as an HWS activity flow or workflow.

This workflow is composed of both automated tasks and tasks that require human action. Each step in the activity flow is a task that a user performs. Users in activity flows are called the target actors of each task. This sample activity flow contains a defined set of tasks based on an activity model.

An HWS implementation of a proposal review process might contain the following steps:

1.   The first workflow participant, known in this scenario as the initiating actor, starts the review process.

2.   The initiating actor supplies the action parameters and an optional resource. For information about action parameters, see **ActionParameters Class**.

3.   HWS uses the action parameters and the information contained in the fact store to constrain the actions that the initiating actor can perform.

4.   HWS returns information about the constraints to the initiating actor.

5.   The initiating actor assigns the review task to three reviewers, known as target actors.

6.   HWS notifies the first target actor about the assigned task.

      The notification is an e-mail message that contains instructions for completing the task and a link to a Web site that displays the proposal.

7.   The first target actor receives the proposal review request and responds to it.

8.   The target actor completes the task.

      This task, for example, may be making a decision regarding the proposal such as accept, reject, or delegate.

9.   The target actor initiates the next action or task.

10.   HWS notifies the second target actor about the assigned task.

11.   The second target actor responds to the notification and completes the task.

12.   HWS notifies the third target actor about the assigned task.

13.   The third target actor responds to the notification and completes the task.

14.   The workflow is complete.

HWS creates an activity flow that contains the ordered sequence of the workflow tasks. The activity flow includes all of the actions, the action initiators, and targets.

For another example of how you can apply this implementation model to a real-life business situation,

HWS Workflow Model

Human Workflow Services (HWS), a **workflow** is a set of activities that takes place between people or processes in a specific context.

The following figure shows the basic workflow model.

HWS workflow model
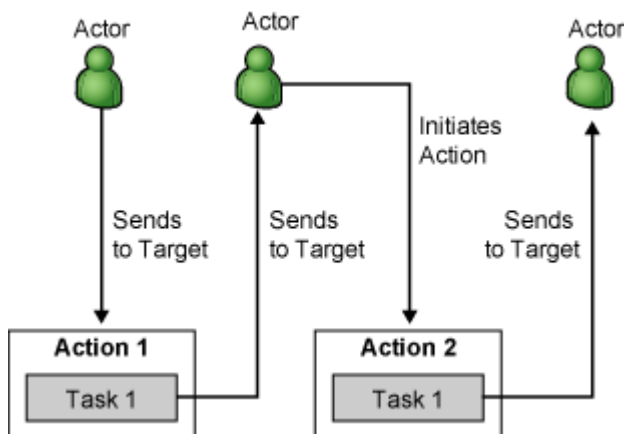
An action represents a fundamental business process, or unit of workflow, that you cannot reduce to further sub-actions. Actions may contain zero or more tasks defining work. Actions do not always send tasks. In dependent composition for example, actions send synchronization message and no tasks.

You assign tasks to the participants (called actors) in the workflow, which are external entities that either start an activity flow or participate in an ongoing activity flow. The actor who initiates the first action within an activity flow becomes the owner of the activity flow.

HWS Architecture

Human Workflow Services (HWS) supports communication, collaboration, and decision making for business people. It provides simple task tracking, run-time composition of workflow, and design-time composition of templates into a loosely coupled architecture.

Participants can use HWS to create workflows at their discretion, constrained only by rules that ensure that the workflows are meaningful for their organization and conform to well-defined organizational practices.

HWS Architecture
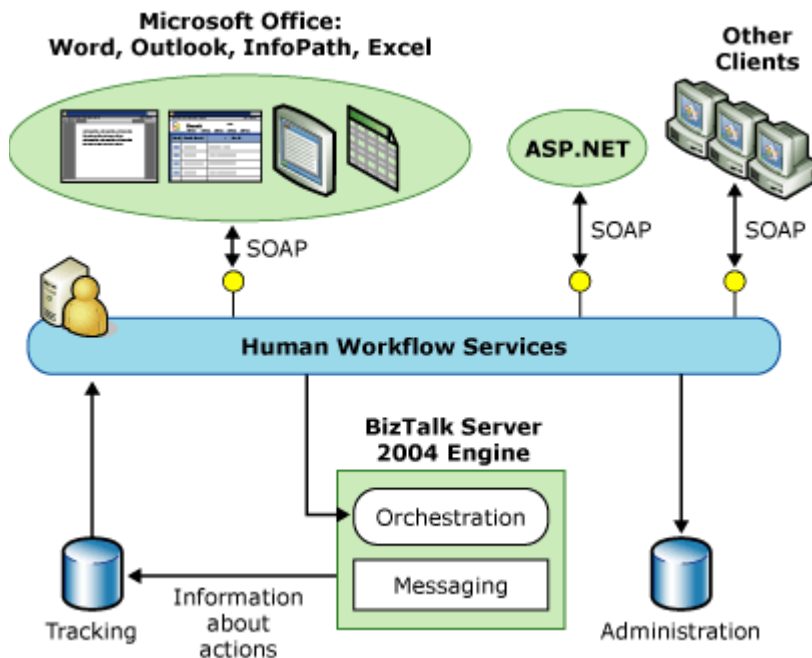
Microsoft Outlook® and Microsoft Word are likely to be the most common clients, along with custom forms built by using InfoPath. ASP.NET applications and anything else that can access standard Web services can also use this part of BizTalk Server 2006.

For clients built on the .NET Framework, HWS provides a library that exposes all of its Web services as .NET-based objects. Although it is not shown in the figure, applications are administered by using the HWS Administration console, an MMC snap-in.

HWS consists of the following processes:

- **HWS Web service interface.** The HWS Web service interface encapsulates the functionality that client applications, such as Microsoft Word, Microsoft Outlook, Windows® SharePoint™ Services, and Microsoft Exchange, need in order to provide workflow capabilities to information workers. Client applications

register with HWS Web services to participate in an activity flow. HWS provides three major services to client applications:

- **Constraint service.** The Constraint service uses information from the fact store to determine the list of actions a user or client can perform. After a user or client selects an action from the constrained set, the Composition service composes the selected actions with those already in use.

  In addition to constraining the actions users may perform, the Constraint service monitors the ability of client applications to add activities to workflows. When a client attempts to attach an action to the activity flow, the Constraint service checks the fact store to see which actions the client can attach.

- **Tracking service.** The Tracking service keeps track of the state of the activity flow, and reconstructs the activity flow as requested by the client. Actions emit tracking events that the Tracking service consumes. Client applications access the tracking events to provide an up-to-date workflow to the information worker.

- **Composition service**. The Composition service associates a unique ID with each client request and uses this ID to keep track of actions that the user performs as part of an activity flow. It provides the capability to dynamically compose actions into HWS workflows. It allows actions to be added to existing activity flows, subject to the constraints imposed by the Constraint service. The composition of actions is governed by constraints that are enforced by the HWS services.

- **HWS Administration console and WMI providers.** The HWS Administration console and the HWS Administration WMI providers provide system administrators with some degree of flexibility in performing administrative tasks. For example, you can define constraints either through the HWS Administration console or through WMI in a programmatic manner.

- **BizTalk Server 2006 platform.** HWS uses the HWS message protocol for workflow execution and tracking.

HWS Components

The following figure shows the basic components used in Human Workflow Services.

Human Workflow Services overview
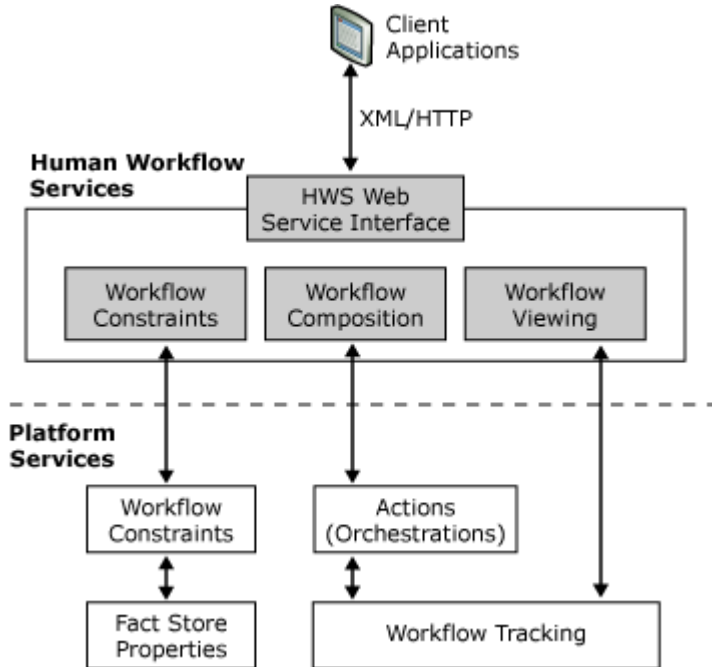
Actors

An actor is an entity external to the workflow that performs tasks as part of a workflow. For example, an actor could be a human being or an application.

Actions

To support this general approach to workflow applications, Human Workflow Services defines a few core abstractions, all of which are built on the BizTalk Server 2006 engine. Every workflow is built from one or more building blocks called actions, each of which is implemented as an orchestration.

Actions are BizTalk orchestrations that receive and send messages that contain extensions to the task schema. An action assigns a task to an actor by sending the actor an HWS message.

For example, a workflow for responding to a request for proposal might have actions such as Review, Approve, Delegate, and Escalate. Like all orchestrations, those used with HWS are created by using Orchestration Designer. Each HWS orchestration has a number of standard behaviors along with any customized behavior created by the developer who builds the application.

HWS uses this process to keep track of assigned tasks and to provide a representation of actor interactions in specific activity flows. For information about task schema extensions, see About Action Template Schemas .

HWS creates tracked events for every action. These tracked events provide up-to-date views of workflow progress. The Constraint service uses tracked events to determine what workflows each actor can create. Actions track the following events:

- Activation point of action.

- Ending point of an action.

- All HWS messages sent or received

Users can perform actions as needed. Additionally, actions can be part of a stored sequence called an activity model.

Developers use an HWS action project template to construct actions. For information about the HWS action project template, see Creating Actions .

Activity models

You use activity models to maximize efficiency. Activity models provide the sequence of actions in an HWS workflow. You use the transitions between each step in an activity model to define the action sequence.

A trusted user can create, perform, and initiate actions in an activity model on behalf of a target participant (HWS enforces system-level constraints by default).

For information about defining activity models, see About Activity Models .

Activation blocks

An activity model may consist of one or more activation blocks. The initiator of the activation block must supply parameters for all actions within the activation block in order to activate them (defaults will be used if parameters are not provided). For information about action parameters, see ActionParameters.

Keep the following guidelines in mind when working with activation blocks:

- Each step within an activation block has a unique ID. Two activation blocks cannot share the same step (although they may have actions of the same type).

- The actor providing the activation parameters for the action instance within an activation block must also provide the activation parameters for all dependent actions within the activation block.

- Loops may originate from any action but must reference the first action within an activation block, and therefore by definition are independently composed.

You can define constraints to restrict the initiator of the root step and targets for any other step within the activity model. In addition, the initiator of all independently composed steps other than the root step must receive a task from another step within the activity model.

For information about creating activation blocks, see <u>ActivityModel</u>.

Activity flows

The actions in a workflow occur in a defined order, called an activity flow. Each activity flow has an activity model that captures actions in a particular order and enables the person designing the activity flow to add structure to the workflow. The users of an HWS application are called actors, and communication between actors and actions happens through tasks, which are XML-defined messages.

Each action has one or more tasks associated with it, and so when an actor clicks a button in an Office application that does something in a workflow, that actor is sending a particular task message to some action (that is, to an orchestration).

Constraints

An HWS application can also impose constraints on the people who use it based on their roles. For example, an application might allow only managers to approve purchase orders of more than a million euros, or might allow only vice presidents to delegate tasks. To support this, the creator of a workflow application can define constraints that rely on roles defined in Active Directory®, in a Microsoft SQL Server™ database, or in other ways.

For information about creating constraints, see <u>About Action Constraints</u> .

Fact retriever

The fact retriever implements a standard HWS interface to enable the Constraint service to retrieve facts from the fact store about users (actors) who initiate or are the target of actions within an activity flow.

HWS User Roles

Workflow design is a collaborative process involving people with diverse skills and responsibilities.

Different user roles have different responsibilities in the creation, administration, and use of workflows. User roles represent behaviors and access rights. Each role has a defined set of behaviors or actions and a defined set of permissions, as follows:

- Action developers use the Human Workflow Services (HWS) project template in Orchestration Designer to create business processes for the HWS framework.

- Activity model designers build activity models with the business processes the action developers created. Activity model designers belong to the Workflow designer SQL Server™ group.

- HWS administrators manage workflows in the Human Workflow Services Microsoft Management Console (MMC). HWS administrators belong to the HWS Administrators SQL Server group.

- Workflow users use the workflows to complete tasks.

- Trusted users are a special class of users with more privileges than normal users. Trusted users can activate new actions and activity models, interrupt workflows, respond to tasks, and retrieve tracking information for specific users.

The following table describes user roles and responsibilities as well as the tools each user typically requires in the HWS workflow process.

User	Activities	Tools
Action developer	• Develops custom actions • Integrates external services into actions • Deploys and tests actions	• Microsoft Visual Studio® • HWS Administration console

Activity model designer	• Composes actions in an a-priori order (governed by activity model)
	
	• Defines constraints for the activity models, which include:
	
	  • Owners constraints
	
	  • Observers constraints
	
	  • Initiator-level constraints (root action)
	
	  • Target-level constraints
	
	• Deploys and uninstalls activity models to which they have ownership rights
		• Orchestration Designer with the HWS action template
		
		• Activity Model Designer API
		
		• Microsoft Visual Studio

HWS administrator	• Installs and configures HWS Web services • Deploys and uninstalls actions • Sets up policies that map actions to their organization • Adds/removes/updates activity model system-level constraints • Deletes and archives activity flows	• HWS Administration console • Administration API

Workflow user	• Initiates and composes actions and activity models governed by constraints • Responds to assigned task messages • Views activity flow information based on visibility rules • Views list of tasks assigned to them • Views list of activity flows that they are participating in • Views list of actions and activity models that they can initiate	Client provides access via the HWS Web service
Trusted user	• Enables visibility into all activity flows running on the system • Activates new actions and activity models, interrupts workflows, responds to tasks, and retrieves tracking information on behalf of a specific user. • Stops, aborts, or rolls back action instances	HWS Web service

Managing Human Workflow Services

You use Human Workflow Services (HWS) to manage workflows in Microsoft® BizTalk® Server 2006. HWS is part of a suite of tools and features in BizTalk Server 2006 specifically designed for information workers. HWS enables information workers to construct, modify, participate in, and track the progress of workflows from various HWS-enabled applications. It gives information workers the ability not only to customize a workflow to their needs but also to capture their day-to-day activities as a workflow. With HWS, information workers can:

• Perform workflow activities either one by one as needed or together as part of a predefined model.

• Monitor what workflow participants do as part of a day-to-day workflow.

• Track the status of workflow activities with visibility into the state of the business process.

This section provides instructions on how to use the HWS Administration console to perform administrative tasks. For information about using Windows Management Instrumentation (WMI) to script administrative tasks, see **Human Workflow Services Classes**.

In This Section

- Using the HWS Administration Console

- Enabling an Action for Use in Human Workflow Services

- Adding Trusted Users

- Editing Human Workflow Services Properties

- Starting and Stopping the Human Workflow Services Web Service

- Changing the Web Service User Context and Password

- Registering and Unregistering Actions

- Adding, Editing, and Deleting Constraints

- Installing, Editing, and Removing Fact Retrievers

- Working with Activity Flows

- Configuring Human Workflow Services

- Optimizing Human Workflow Services

Using the HWS Administration Console

You can use the Human Workflow Services (HWS) Administration snap-in to manage the HWS administration components. The left side of the administration console, called the console tree, displays the HWS objects, such as Web Services, Fact Retrievers, and Actions. The right side of the administration console, called the results pane, displays information about the selected item in the console tree.

The HWS Administration snap-in contains the following nodes that you can use to manage the HWS objects:

- **Web Services**

    Use the **Web Services** node to view a list of all currently installed HWS Web services. You can start or stop the IIS Web site the Web service is running on from within this node. For more information, see <u>Starting and Stopping the Human Workflow Services Web Service</u> .

    Constraints

    Use the **Constraints** node results pane to view all the system-level and activity model system-level constraints defined in the Human Workflow Services (HWS) server.

    By default when an action is registered in the Action node, no user can run it

until constraints are added to it. These constraints explicitly define the set of users who can initiate an action, or be the target of an action. Constraints enforce a security model for either granting or denying permissions to users or groups of users running HWS. For more information, see <u>Adding, Editing, and Deleting Constraints</u> .

- **Actions**

  Use the **Actions** node to view information about Human Workflow Services (HWS) orchestrations deployed on the HWS server. All orchestrations that have been created using the HWS action project template will appear with a status of Enlisted, Invalid, Registered, Running, or Not Enlisted. For more information, see <u>Registering and Unregistering Actions</u> .

- **Fact Retrievers**

  Use the **Fact Retrievers** node to manage the fact retrievers associated with Human Workflow Services (HWS). A list of the fact retrievers currently installed is displayed. A default fact retriever named Intrinsic will always exist, and supplies user name, time, and date properties. For more information, see <u>Installing, Editing, and Removing Fact Retrievers</u> .

- **Activity Models**

  Use the **Activity Models** node to view all the enabled activity models installed on the Human Workflow Services (HWS) server and currently available. The results pane displays a list of the activity models available in the HWS server.

- **Activity Flows**

  Use the **Activity Flows** node to view a summary of all workflows that are running or have been run in this HWS system. This node also enables you to delete or archive activity specific workflow data. For more information, <u>Working with Activity Flows</u> .

How to Open the HWS Administrative Console

use the HWS Administration console to interact with actions, activity flows, fact retrievers, constraints, and web services.

Some operations performed in the HWS Administration console change the state of the HWS system. Operations that change the state of the HWS system include activities such as registering an action, changing a property, or adding a constraint. When a state changing operation is performed, the HWS Administration console must let all the installed Web services for the system know that a change has occurred.

If the HWS Administration console has not yet notified all installed HWS Web services of a state change at the time the associated operation completes, the HWS Administration console will display an error message.

The error message informs you that you must refresh or restart the Web services that use the setting that the operation changed so that they use the new setting.

To verify that the state change has occurred, refresh the HWS Administration console.

Prerequisites

To open the Human Workflow Services Administration snap-in

1.   Click **Start**, point to **All Programs**, point to **Microsoft BizTalk Server 2006**, and then click **HWS Server Administration**.

2.   In the HWS Administration snap-in, expand the **Human Workflow Services (Local)** node.

Enabling an Action for Use in Human Workflow Services

Administering Human Workflow Services (HWS) involves working with the HWS Administration console to view, update, and create trusted users, workflow properties, Web services, actions, constraints, fact retrievers, activity models, and workflows, as well as managing the data associated with workflows.

You must enable actions for use in HWS.

Prerequisites

To enable an action

1.   Register your action in the **Action** node of the HWS Administration console. For more information, see Registering and Unregistering Actions.

       This registers the action with the HWS Web service. After the action is registered with HWS, it is still unavailable to all actors in the system.

2.   Add a constraint that allows one or more actors to execute the action.

       If your constraint needs to use information from a new fact store, you must add a new fact retriever in the **Fact Retrievers** node. For more information, see Installing, Editing, and Removing Fact Retrievers.

Add a constraint to your registered action in the **Constraints** node. For more information, see Adding, Editing, and Deleting Constraints.

A valid constraint must have at least one initiator and one target to fully enable an action for use within HWS.

Adding Trusted Users

You use the Human Workflow Services (HWS) Administration console to specify the accounts that the HWS system trusts. These trusted accounts can bypass the tracking authentication check in the Web service.

Microsoft® BizTalk® Server installs the user who installs HWS as a trusted user by default. All users with the authority to use the HWS Administration console can add or remove users to or from the Trusted Users group.

Note the following information about trusted users:

- Local users can be trusted users. However, only local users on the HWS administration computer are valid.

- If local users of computers other than the HWS administration computer are trusted users, an error will occur when you use the HWS Administration console or WMI to update the HWS core properties.

- By default, the HWS WS user is added as a trusted user. If the HWS WS user is a local user, update of HWS core properties via WMI/MMC may fail.

- If your configuration contains multiple HWS administration computers, you can delete local users from the other administration computers and add the necessary local users to the administration computer that is being worked on.

Prerequisites

To add trusted users

1. Click **Start**, point to **All Programs**, point to **Microsoft BizTalk Server 2006**, and then click **HWS Server Administration**.

2. In the Administration console, expand the **Human Workflow Services (Local)** node.

3. Right-click the **Human Workflow Services (Local)** node, and then click **Properties**.

4. In the **Human Workflow Services (Local) Properties** dialog box, click the **Trusted Accounts** tab, and then click **Add.**

5. In the **Add Trusted Account** dialog box, type the domain and user name information for the trusted user you want to add, and then click **OK**.

6. Click **OK** to close the **Human Workflow Services (Local) Properties** dialog box.

   You must restart all installed Web services before the trusted user change is processed.

Editing Human Workflow Services Properties

You use the Human Workflow Services (HWS) Administration console to change the values of the HWS properties. The following table describes the HWS properties.

HWS property	Description
BizTalk Server database computer	Use this property to identify the computer on which Microsoft® BizTalk® Server 2006 is installed.
URL for activation	Use this property to change the Hws_Activate messages receive location.
URL for task response and interrupt	Use this property to change the Hws_Interrupt messages receive location.
HWS admin database server	Use this property to view the identity of the computer on which the HWS administration database is installed. This property is read-only
HWS admin database name	Use this property to view the identity of the HWS administration database. This property is read-only.
Use constraints as guidance	Select this check box to bypass the Constraints system at run time. Use the **Trusted Accounts** tab to add trusted HWS users.

For information about adding a trusted user, see Adding Trusted Users.

Prerequisites

To edit Human Workflow Services properties

1.   Click **Start**, point to **All Programs**, point to **Microsoft BizTalk Server 2006**, and then click **HWS Server Administration**.

2.   In the Administration console, click the **Human Workflow Services (Local)** node.

3.   On the **Action** menu, click **Properties**.

4.   In the **Human Workflow Services (Local) Properties** dialog box, update the values in the text boxes as necessary.

Starting and Stopping the Human Workflow Services Web Service

You may need to stop and restart the Human Workflow Services (HWS) Web service—for example, if you edited the HWS properties and want to refresh the Web services that use them. To stop and start the HWS Web service, you must stop and start the Internet Information Services (IIS) Web site.

It is strongly recommend that no other applications run on the same site as the HWS service.

You must be a local administrator to perform this procedure. BizTalk® Administrators cannot start or stop the Web service.

1.   Click **Start**, point to **All Programs**, point to **Microsoft BizTalk Server 2006**, and then click **HWS Server Administration**.

2.   In the Administration console, expand the **Human Workflow Services (Local)** node.

3.   Click the **Web Services** node.

4.   In the details pane, click the Web service you want to start.

5.   On the **Action** menu, click **Start Web Service**.

Prerequisites

To stop the Web service

1.   Click the **Web Services** node.

2.   In the details pane, click the Web service you want to stop.

3.   On the **Action** menu, click **Stop Web Service**.

Changing the Web Service User Context and Password

Microsoft® BizTalk® Server 2006 installation sets the password and user context for the HWS Web service. To change the password and user context for the Web service, follow the steps outlined in the Knowledge Base article "Use the ASP.NET Utility to Encrypt Credentials and Session State Connection Strings," located at http://go.microsoft.com/fwlink/?LinkID=16728.

Prerequisites

To update the user context and password for IIS version 6.0 on Windows Server 2003

1.   Update the identity for HwsAppPool:

a.   Run the following from the product installation folder:

aspnet_setreg.exe -k:"SOFTWARE\Microsoft\BizTalk
Server\3.0\Hws\WebService\Identity" -u:<user> –p:<password>

b.   Press ENTER.

2.   Give the user read permission for the registry key
**\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\BizTalk
Server\3.0\Hws\WebService**.

3.   Give the user read permission for the registry key
**\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\BizTalk
Server\3.0\Hws\WebService\Identity\ASPNET_SETREG**.

4.   Add the user to the IIS_WPG group.

5.   Add the user to the STS_WPG group if the group is available.

6.   Add the user to the HWS_WS_USER role of BizTalkHwsDb, BizTalkMgmtDb, and BizTalkDTADb.

7.   Add the user to the BizTalk Isolated Host Users.

8.   At the command prompt, type **iisreset**, and then press ENTER.

To update the user context and password for IIS version 5.0 on Windows 2000 Server

1.   Update the identity for the HwsMessages COM+ application:

•   Run the following from the product installation folder:

aspnet_setreg.exe -k:"SOFTWARE\Microsoft\BizTalk
Server\3.0\Hws\WebService\Identity" -u:<user> –p:<password>

2.   Give the user read permission for the registry key
**\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\BizTalk
Server\3.0\Hws\WebService**.

3.   Give the user read permission for the registry key
**\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\BizTalk
Server\3.0\Hws\WebService\Identity\ASPNET_SETREG**.

4.   Add the user to the IIS_WPG group.

5.   Add the user to teh HWS_WS_USER role of BizTalkHwsDb, BizTalkMgmtDb, and BizTalkDTADb.

6.   Add the user to the BizTalk Isolated Host Users.

7.   At the command prompt, type **iisreset**, and then press ENTER.

Registering and Unregistering Actions

You must register actions for use in Human Workflow Services (HWS). You use the HWS Administration console to register an action after a solutions developer creates, compiles, deploys, and enlists the action.

Unregistering an action removes from the system all constraints referencing that action.

Prerequisites

To register an action in running state

1.   Click **Start**, point to **All Programs**, point to **Microsoft BizTalk Server 2006**, and then click **HWS Server Administration**.

2.   In the Administration console, expand the **Human Workflow Services (Local)** node.

3.   Click the **actions** node.

4.   In the details pane, click the action to be registered.

5.   On the **Action** menu, click **Register Action: <*action name*>**.

To unregister a registered action

1.   Click the **actions** node.

2.   In the details pane, click the action to be unregistered.

3.   On the **Action** menu, click **Unregister Action: <*action name*>**.

Adding, Editing, and Deleting Constraints

A constraint is a property that restricts the initiator of the root action and targets for any other action within the activity model. You can use the Human Workflow Services (HWS) Administration console to add, edit, and delete constraints. For information about constraints, see About Action Constraints .

A solutions developer uses the **ActivityModel.AddRoleConstraint** method to set the Initiator, Owner, and Observer roles. For information about the **ActivityModel.AddRoleConstraint** method, see **ActivityModel.AddRoleConstraint Method**. For information about sample code demonstrating the **ActivityModel.AddRoleConstraint** method, see

HWS (BizTalk Server Samples Folder) .

You can add constraints for users in the following roles:

- **Initiator**. An activity model initiator is a user who starts an action. Often a particular user or set of users is designated as the initiator.

- **Owner**. An activity model owner has permission to extend an activity flow by adding actions as needed to the activity model. The owner can perform the steps in the activity model.

- **Observer**. An activity model observer has permission to view an activity flow, but does not need to participate in the activity flow.

- **Enacted on**. An enacted-on user is a user on which a business action acts. For example, if a user does not complete an action, the initiator or owner can use an escalate action to inform the manager that the user did not complete the action. In this case, the user whose action resulted in the need to escalate is the enacted-on user. The manager of the enacted-on user is the target of an escalate action.

- **Target**. A target user is a user to which an owner or initiator assigns an action.

1. Click **Start**, point to **All Programs**, point to **Microsoft BizTalk Server 2006**, and then click **HWS Server Administration**.

2. In the Administration console, expand the **Human Workflow Services (Local)** node.

3. Click the **Constraints** node.

4. On the **Action** menu, click **Add Constraint**.

5. In the **Constraint Properties** area, select an **action** if you want to apply the constraint to the action for all activity models in which it is included, and then do the following:

Use this	To do this
Activity model	Select an activity model from the drop-down list if you want to apply the constraint to an action in that specific activity model only.
Negative Constraint	Select the check box to define a negative constraint. Negative constraints prevent users who match the criteria of the constraint from performing the action.
Target group name	Select the check box to create an alias for the set of target users to whom this constraint applies. Type the alias name in the text box. When executing the action, the initiating user can refer to the alias instead of specifying all users returned from the constraint. This is useful when the initiating user targets many actors.

6.   In the **Constraint Clauses** area, click **Add**.

7.   In the **Add Clause** dialog box, do the following:

Use this	To do this
Constraint Fact	Select from the drop-down list the fact store property to be evaluated.
Operator	Select from the drop-down list the operation you want performed on the property value.
Values	Type a new value or select an existing value on which to perform the operation.
Multiple Values	Use this text box if multiple values are needed (for example, an OR statement). Type a value, click **Add to Value List**, type the next value, and so on.

8.   The values available in the **Operator** drop-down list are dependent on the data type of the property. The following table describes the data types and their corresponding operators.

Data type	Description	Permitted operations
String	A property of data type string can contain a maximum of 256 Unicode characters.	• Equals • Not Equal • Includes one of • Does not include any of
User ID	A property of data type User ID can contain a maximum of 256 Unicode characters.	• Equals • Not Equal • Includes one of • Does not include any of
Boolean	A property of data type Boolean can contain the value **true** or **false**.	• Equals • Not Equal

Integer	A property of data type integer is a 32-bit integer.	• Equals • Not Equal • Greater Than • Less Than • Includes one of • Does not include any of
Real	A property of data type real is a 32-bit floating-point number.	• Equals • Not Equal • Greater Than • Less Than • Includes one of • Does not include any of
Date	A property of data type date contains a date. The user locale defines the date format.	• Equals • Not Equal • Greater Than • Less Than • Includes one of • Does not include any of
Time	A property of data type time contains a time. The user locale defines the time format.	• Equals • Not Equal • Greater Than

		•	Less Than
		•	Includes one of
		•	Does not include any of
		•	Equals
		•	Not Equal
		•	Greater Than
Datetime	A property of data type datetime contains a combined date and time. The user locale determines the datetime format.	•	Less Than
		•	Includes one of
		•	Does not include any of

9.   Click **OK** to create the constraint.

Prerequisites

To edit a constraint

1.   Click **Start**, point to **All Programs**, point to **Microsoft BizTalk Server 2006**, and then click **HWS Server Administration**.

2.   In the Administration console, expand the **Human Workflow Services (Local)** node.

3.   Click the **Constraints** node.

4.   In the details pane, click the constraint you want to edit.

5.   On the **Action** menu, click **Properties**.

6.   In the **<constraint name> Properties** dialog box, update the fields as necessary.

7.   Click **OK** to finish editing the constraint.

To delete a constraint

1.   Click **Start**, point to **All Programs**, point to **Microsoft BizTalk Server 2006**, and then click **HWS Server Administration**.

2.   In the Administration console, expand the **Human Workflow Services (Local)** node.

3.   Click the **Constraints** node.

4.   In the details pane, click the constraint you want to delete.

5.   On the **Action** menu, click **Remove Constraint**.

6.   In the **Constraint Removal Warning** dialog box, click **Yes**.

Installing, Editing, and Removing Fact Retrievers

You use the Human Workflow Services (HWS) Administration console to install fact retrievers. The fact retriever implements a standard HWS interface to enable the Constraint service to retrieve facts from the fact store about users who initiate or are the target of actions within an activity flow (actors).

For information about creating HWS fact retrievers, see About Fact Retrievers .

Prerequisites

To install a fact retriever

1.   Click **Start**, point to **All Programs**, point to **Microsoft BizTalk Server 2006**, and then click **HWS Server Administration**.

2.   In the Administration console, expand the **Human Workflow Services (Local)** node.

3.   Click the **Fact Retrievers** node.

4.   On the **Action** menu, click **Add Fact Retriever**.

5.   In the **Add Fact Retriever** dialog box, fill in the information about the fact retriever assembly as follows:

Use this	To do this
Assembly culture	Enter the culture of the fact retriever assembly.
Assembly name	Enter the name of the fact retriever assembly.
Assembly version	Enter the version of the fact retriever assembly.
Assembly public key	Enter the public key used for the fact retriever assembly.
Fact retriever Typename	Enter the name of the class that implements the **IFactRetriever Interface**.
Adapter connection	Enter the full connection string required to access the fact

string	store.

6.   Click **OK**.

To edit a fact retriever

1.   Click **Start**, point to **All Programs**, point to **Microsoft BizTalk Server 2006**, and then click **HWS Server Administration**.

2.   In the Administration console, expand the **Human Workflow Services (Local)** node.

3.   Expand the **Fact Retrievers** node.

4.   Click the **<Fact Retriever name>**.

5.   On the **Action** menu, click **Properties**.

       The **<Fact Retriever Name> Properties** dialog box appears.

6.   You can edit the fact retriever display name, description, and connection string. All other properties are read-only

7.   Click **OK** to finish editing.

To remove a fact retriever

1.   Click **Start**, point to **All Programs**, point to **Microsoft BizTalk Server 2006**, and then click **HWS Server Administration**.

2.   In the Administration console, expand the **Human Workflow Services (Local)** node.

3.   Expand the **Fact Retrievers** node.

4.   Click the **<Fact Retriever Name>**.

5.   On the **Action** menu, click **Remove Fact Retriever : <Fact Retriever Name>**.

6.   Click **Yes** on the warning dialog to remove the fact retriever

Working with Activity Flows

An activity flow contains the ordered sequence of tasks of a Human Workflow Services (HWS) workflow including actions, action initiators, and action targets.

An administrator uses Windows Management Instrumentation (WMI) or a solutions developer uses the **ActivityFlow** class to create and modify activity flows. For information about using WMI to create and modify activity flows, see **HWS_Activity**

Flow (WMI). For information about the **ActivityFlow** class, see **ActivityFlow Class**.

You use the **Activity Flows** node in the HWS Administration console to view a summary of all workflows that are running or that have run in this HWS system. This node also enables you to delete or archive activity-specific workflow data.

Prerequisites

To view inactive activity flows

1. Click **Start**, point to **All Programs**, point to **Microsoft BizTalk Server 2006**, and then click **HWS Server Administration**.

2. In the HWS Administration console, expand the **Human Workflow Services (Local)** node.

3. Right-click the **Activity Flows** node and then click **Properties.**

4. In the **Activity Flows Property** dialog box, select the **Inactive** check box, and then click **OK.**

    A list of inactive activity flows appears in the results pane.

To back up the activity flow tracking data

1. Click **Start**, point to **All Programs**, point to **Microsoft BizTalk Server 2006**, and then click **HWS Server Administration**.

2. In the HWS Administration console, expand the **Human Workflow Services (Local)** node.

3. Click the **Activity Flows** node.

4. In the details pane, click the activity flow you want to back up.

5. On the **Action** menu, click **Backup Activity Flow Data**.

6. In the **Save As** dialog box, type the name of the new backup file.

To delete an activity flow

1. Click **Start**, point to **All Programs**, point to **Microsoft BizTalk Server 2006**, and then click **HWS Server Administration**.

2. In the HWS Administration console, expand the **Human Workflow Services (Local)** node.

3. Click the **Activity Flows** node.

4. In the details pane, click the activity flow you want to delete.

5.   On the **Action** menu, click **Delete Activity Flow**.

To delete a set of activity flows

1.   Click **Start**, point to **All Programs**, point to **Microsoft BizTalk Server 2006**, and then click **HWS Server Administration**.

2.   In the Administration console, expand the **Human Workflow Services (Local)** node.

3.   Click the **Activity Flows** node.

4.   On the **Action** menu, click **Batch Delete**.

5.   On the **Batch Activity Flow Delete** dialog box, do the following:

Use this	To do this
Time based	In the left box, type the value of time as an integer (for example, 1). (For days, 0=Sunday.) In the right box, from the drop-down list, select the unit of time (years, months, weeks, days). Selecting the value and unit of time indicates that you want to delete all activity flows older than the time you indicated. For example, selecting **1 week** results in deleting all activity flows older than 1 week. After you specify the value and unit of time, the number of activity flows that will be deleted appears above the **Delete** button.
Activity based	In the left box, type the value of time as an integer (for example, 1). In the right box, from the drop-down list, select the unit of time (years, months, weeks, days). Selecting the value and unit of time indicates that you want to delete all activity flows that have not run since the time you indicated. For example, selecting **1 week** results in deleting all activity flows that have not run in the previous seven days. After you specify the value and unit of time, the number of activity flows that will be deleted appears above the **Delete** button.

6.   Click **Delete** in the appropriate section, and then click **Close**.

Configuring Human Workflow Services

In a distributed configuration of Microsoft® BizTalk® Server 2006 where the BizTalk application host computer is different from the host computer running the BizTalk Server receive handlers, you must perform the following additional configuration tasks after completing the BizTalk Server 2006 installation and configuration steps:

- For custom Web services installations, if there are any fact retrievers (in addition to the one built into HWS), you must deploy fact retriever assemblies and store them in the global assembly cache on all servers that have Web services installed.

- For the Human Workflow Services (HWS) Web server-only installation, deploy fact retrievers and store them in the global assembly cache on each Web server.

- For an Administrative Tools installation, deploy fact retrievers and store them in the global assembly cache for each administrative server.

- You must enable HWS to work with HTTPS.

Prerequisites

To enable HWS to work with HTTPS

1. Update the URLs to use HTTPS in the HWS_Service table of the BizTalkHwsDb database.

2. Use the HWS Administration console or Windows® Management Instrumentation (WMI) to change the Activate/Interrupt Port to use HTTPS.

For information about using WMI to administer HWS, see **Human Workflow Services Classes**.

Optimizing Human Workflow Services

The system administrator can use a number of methods and tools to plan, analyze, and test the implementation of a workflow in order to optimize its performance.

Performance counters allow you to monitor specific aspects of work performed on the site or system by service. Performance counters can help you identify and troubleshoot server performance issues.

The following table lists the performance counters available in Human Workflow Services (HWS).

Counter	Description
New Activity Flows	Number of requests for creating new activity flows
New Action Added	Number of requests for instantiating actions or activity

	blocks
Activity Flows Retrieved	Number of requests for retrieving activity flow info
Action Instances Retrieved	Number of requests for retrieving action info
Tasks Retrieved	Number of requests for retrieving task info

Prerequisites

To access performance counters

1.   On the Desktop, click **Start**, point to **Programs**, point to **Administrative Tools**, and then click **Performance.**

2.   In the **Performance** dialog box, click **Add**

3.   In the **Add Counters** dialog box, from the **Performance** object drop-down list, select **HWS Performance Counters**, and then click **Add.**

4.   In the **Add Counters** dialog box, do one of the following:

Use this	To do this
All counters	Select this option to select all counters from the provided list.
Select counters from list	Select this option to select specific counters from the provided list.

1.   After selecting the counters, click **Add** and then click **Close**.

The selected performance counters appear on the Performance screen.

Human Workflow Services Security Recommendations

With Human Workflow Services (HWS), information workers can construct, modify, participate in, and track the progress of workflows easily from various HWS-enabled applications. For more information about Human Workflow Services, see Managing Human Workflow Services. It is recommended you follow these guidelines for securing and deploying HWS in your environment.

- Any authenticated domain users can attempt to run HWS Web service methods. Anonymous users cannot.

- You need to ensure there are mitigations for Denial of Service (DoS) attacks in place to avoid a domain user causing a DoS on the HWS Web service. For more information, see **Mitigating Denial of Service Attacks**.

- When you use Secure HTTP (S/HTTP), the computer where HWS is running has a signed certificate telling all the clients who it is. If a Certificate Authority

(CA) that is not trusted by the client signs the certificate, the HWS Administration console reports an error every time it accesses the HWS Web service (which it does in almost every administrative task.) You must add the CA to the list of trusted CAs for the user, or the computer.

- HWS relies on the fact retriever to find out information about the people in your company. If you use a custom fact retriever that relies on an external resource, you may need to open additional network paths to and from the services domain.

- HWS transmits the connection string for the fact retriever in clear text. If you use SQL Server authentication, you need to include the username and password in the connection string, leading to potential information disclosure. It is recommended you use Windows authentication for the fact retriever to avoid the need of having this information in the connection string.

- Users in the intranet domain can design activity models. If you want these users to deploy their activity models, a BizTalk administrator can:

  - Deploy the activity model for them.

  - Create a specific account these users can use to deploy activity models. Follow the recommendations for the connection string for the fact retriever described in the preceding point.

  - Give the users permissions to deploy their own activity models either by replicating their accounts in the data domain, or by giving their account the appropriate permissions in the data domain. This requires opening the SQL port on the firewall protecting the data domain, and giving these users access to connect to the HWS administration database, which is the least secure option.

Developing Human Workflow Services Solutions

Before you begin, you should be familiar with the concepts in **What Is Human Workflow Services (HWS)?** For Human Workflow Services code samples, see HWS (BizTalk Server Samples Folder).

Human Workflow Services (HWS) addresses the requirements for supporting a human-oriented automated workflow. HWS not only enables you to adapt a workflow to the needs of workers in an organization, but it also enables you to capture business activities as a workflow. You can use HWS to create actions in Microsoft® BizTalk® Server 2006 Orchestration Designer, and then those actions serve as the building blocks for you to compose rich workflows that model your business processes.

Using HWS involves the following major steps:

- Creating an action using the HWS Action template

- Configuring the action

- Configuring constraints

- Creating the workflow model

- Driving the action using the HWS Web service

HWS provides a development framework for creating client applications that support human-oriented workflows. When implemented in a client application, HWS enables information workers to create processes easily where none previously existed. In addition, you can use HWS to:

- Create and participate in workflows by using client applications that expose HWS workflow constructs through the HWS Web services.

- Author workflows by composing workflow building blocks on a case-by-case basis or by using modeled workflows.

- Customize workflows to suit departmental policies by leveraging the workflow constraints system.

- View the status of ongoing workflows in real time, and easily identify process bottlenecks.

Client applications leverage HWS to enable information workers to create and participate in workflows.

This section contains:

- Creating an HWS Project

- About Actions

- Creating Actions

- About Action Templates

- About Action Constraints

- About Activity Models

- About Fact Retrievers

- About Action Template Schemas

- Creating Action Template Schemas

Creating an HWS Project

This section discusses how to create a Human Workflow Services (HWS) project in Microsoft® Visual Studio® .NET and discusses the files created as a result.

To create an HWS project

1.   Open Visual Studio .NET.

2.   In the Visual Studio .NET window, on the **File** menu, click **New**, and then click **Project**.

3.   In the New Project window, from the Project Types list click **BizTalk Projects**, and then from the Templates list, click **BizTalk Server Human Workflow Project**.

4.   In the **BizTalk Server Human Workflow Project** dialog, in the **Name** box, type the name of your project, and then click **OK**.

HWS creates the following files by default when you create a new HWS project:

- **Action.odx**: This is the starter orchestration file used to build HWS actions.

- **Hws_Activate.xsd**: This is the XML schema for the activation message of the action. You can extend the ActionSection of this schema to hold action specific parameters and values. An action can have only one task message schema associated with it.

- **Hws_Synchronize.xsd**: This is the XML schema for a synchronize message that an action may send or receive to or from another action during execution. You can extend the ActionSection of this schema to hold action or message specific parameters and values. An action can send or receive multiple synchronize messages of the same type or different types. The Target Namespace of the schema defines a message type. If an action needs to send, receive, or synchronize messages of multiple types, then you should use multiple copies of Hws_Synchronize.xsd to define them. If multiple actions share the same set of synchronize message types by way of sending or receiving them, then consider building these into a common assembly that the action projects can reference.

- **Hws_Task.xsd**: Actions use this XML message schema to assign tasks to a user and receive corresponding responses. You can extend the ActionSection of this schema to hold action or message specific parameters and values. An action can send or receive multiple task messages of the same type or different types. If multiple DLLs must share the same schema with the same namespace, you should define them in a separate DLL or defined them in one DLL that all other DLLs reference. If this is not done, and the schema is defined in multiple deployed assemblies, then messages of that schema type will fail since the pipeline will be unable to find the correct schema

The action template references a number of external references and the following are of particular interest:

- **Microsoft.BizTalk.Hws.HwsPromotedProperties**. This assembly contains the promoted property schema definition for properties promoted out of HWS schemas.

- **Microsoft.BizTalk.Hws.HwsSchemas**. This assembly contains schema definitions for Hws_ActivateResponse, Hws_Interrupt, and Hws_Finish messages. Actions send or receive these messages.

- **Microsoft.BizTalk.Hws.HwsExceptions**. This assembly contains definitions for **AbortException** and **RollbackException** classes used by the action. It also contains a definition of the **TemplateExceptionResourceManager** class, which you use to retrieve language-specific messages for exceptions.

Note that adding a new item to the project does not enable you to add a new Synchronize or Task message schema. You must make copies from an existing file and include them into the project. It is important to change the target namespace in the new copies of the schemas.

To change the target namespace

1.   Open the schema in BizTalk Schema Editor.

2.   In the BizTalk Schema Editor dialog, right-click the **<Schema>** node in the tree view and click **Properties**. The Properties dialog appears.

3.   In the Properties dialog, the Target Namespace displays under the General category for the <Schema> node.

4.   In the Target Namespace field, type the name of your new target namespace and click **OK**.

About Actions

Actions are individual Microsoft® BizTalk® Server orchestrations that represent the smallest unit of work that you can perform. You can chain several actions together to form a dynamic activity flow. For action code samples, see <u>Actions (BizTalk Server Sample)</u>.

Creating Actions

This section provides all the details for creating actions. Creating actions is summarized as follows:

1.   Create a new Microsoft® BizTalk® Server HWS project in Microsoft® Visual Studio® .NET for the action.

2.   Modify the activation message.

3.   Modify the task message.

4.   Add business logic to your action.

5.   Create and bind the action's ports.

6.   Deploy and register your action.

To create an action, you must first create a new HWS project in Microsoft Visual Studio .NET (see Creating an HWS Project. In your new project, you will see an action template, which is an orchestration that implements your action logic, and three schema files. The action template is not as complex as it appears, since you only change part of the orchestration. The rest of the orchestration exists only to support other parts of HWS functionality. For a detailed explanation of the action template, see About Action Templates.

Note   You cannot publish actions as a Web service. If you try to publish actions as a Web service you will receive errors indicating that the receive port types are not public. Instead, we recommend that all your activations go through the Web service.

This section contains:

•   Modifying the Activation Message

•   Modifying the Task Message

•   Adding Action Logic and Tasks

•   Creating and Binding HWS Ports

•   Deploying and Registering Actions

•   Creating Dependent Actions

•   Correlating Actions with Tasks

Modifying the Activation Message

When you create an action, you must first modify the activation message to supply your action with data it needs to operate. The Hws_Activate.xsd file contains the schema for the activation message. This schema defines the message sent from the user through the **AddActionToActivityFlow** or **AddActivationBlockToActivityFlow** methods on the Human Workflow Services (HWS) Web service; this message causes the action to begin processing. When you open the Hws_Activate.xsd file and fully expand the activation message schema, you will see that the activation message contains the following three major sections:

•   **HwsSection**. This section provides a location for data on which the HWS engine operates. You should not modify this section of the activation message. Any modifications you make to this section may introduce errors into the system when you run your action. For more information about the various elements in this section of the activation schema, see About Action Template Schemas.

- **ActionSection**. This section is initially empty in any new HWS Action project. This section the "user defined" section of the activation message. You add any elements to send data into your action that you want in the ActionSection. You can add any data type and as many schema elements as you want inside the ActionSection tags.

- **Payloads**. This section is largely irrelevant at design time. You use this section at run time to attach auxiliary resource information to the action, which is any information that is pertinent to the process but that the action does not consume.

After you modify the activation message to suit your action, change the target namespace.

To change the target namespace

1.   In the BizTalk Schema Editor dialog, right-click the **<Schema>** node in the tree view and click **Properties**. The Properties dialog appears.

2.   In the Properties dialog, the Target Namespace displays under the General category for the <Schema> node.

3.   In the Target Namespace field, type the name of your new target namespace and click **OK**.

With each new HWS action project, the Target Namespace value always defaults to http://tempuri.org/Hws_Activate_Sample. You must change the namespace to something unique for your action, for example: http://tempuri.org/MyAction_Activate.

Modifying the Task Message

A task message is a message sent through a BizTalk port that conforms to the Hws_Task schema template. Most actions will communicate in some way with the users of the system and a task message is the recommended way of doing this.

Since it is common for an action to send at least one task, any new HWS action project is created with an Hws_Task.xsd file in it, which is the schema for that task. If you open the Hws_Task.xsd file in Visual Studio .NET and expand all the nodes in the schema, you will see that the Hws_Task schema looks very similar to the Hws_Activate schema. The schema has three major sections, which are the same as the activate message:

- **HwsSection**. Just like the activate schema, the HwsSection contains various elements that carry information required by HWS in order for the message to work properly. You should not modify the elements in this section when creating a new action.

- **ActionSection**. You add any action-specific elements in this section. Data placed in this section of a task message is data for the user that receives the task, such as a string field that explains what the users needs to do. As in the

activate message, you can add as many elements of any type inside the ActionSection tags.

- **Payloads**. This section is largely irrelevant at design time. You use this section at run time to attach auxiliary resource information to the action, which is any information that is pertinent to the process but that the action does not consume.

After you modify the task message to suit your action, you must change the target namespace.

To change the target namespace

1.   In the BizTalk Schema Editor dialog, right-click the **<Schema>** node in the tree view and click **Properties**. The Properties dialog appears.

2.   In the Properties dialog, the Target Namespace displays under the General category for the <Schema> node.

3.   In the Target Namespace field, type the name of your new target namespace and click **OK**.

With each new HWS action project, this value always defaults to http://tempuri.org/Hws_Task_Sample. You must change the namespace to something unique for your action, for example: http://tempuri.org/MyAction_Task.

You may want your action to send multiple task types, and when developing an activity model, multiple task types can help add definition to an activity flow since it is easier to understand that the tasks are different.

To add more task schema types to an action

1.   Copy of the Hws_Task.xsd file and provide an alternate name for the copy.

2.   In Visual Studio.NET, add the Hws_Task.xsd file to your HWS Action project using the Visual Studio Add Existing Item command.

3.   This procedure will save you the trouble of replicating all the required HwsSection elements if you had instead just created a new schema. Ensure that you alter the Target Namespace property of the new schema since you cannot use the same Target Namespace, even within a single action project.

Adding Action Logic and Tasks

After you modify your message schemas, you need to make your action do something. At this point, manipulating the way the action performs is the same as designing any other orchestration in BizTalk Orchestration Designer. You should follow a few important design guidelines when creating actions:

- **Placing action logic**. Place all action logic inside the DoAllActionSpecificLogicHere group. All other sections of the HWS action template

support the run time features of the HWS system, and modifying any other sections may lead to errors when you run the action.

- **Constructing and sending tasks**. First, create a message that contains a message type of one of the task schemas you want. Then, populate all required fields of the HwsSection of the task message. Most of these fields can be populated by creating a map between the activate message and the task message, and copying the data from the activate message. Additionally, the following fields do not appear in the activation message; however, you must set them to a value before the orchestration sends them out:

  - ActorElementXPath

  - TargetActor

  - TaskStatus

  - PercentageComplete

  - NumberofResponses

  - TaskDescription

  - TaskID

- For more information about each of these elements, see <u>About Action Template Schemas</u>.

- **Receiving responses to tasks**. If your action needs to wait for one or more responses from the users that receive tasks, the action adds receive shapes that use the same schema as the user tasks. Task messages and task response messages must share the same schema.

Creating and Binding HWS Ports

You must bind the logical ports of HWS-enabled actions to physical HTTP ports that you create using BizTalk Explorer in Microsoft Visual Studio .NET.

A typical HWS action has two logical receive ports that you need to bind:

- ActionActivationPort

- ActionInterruptPort

Use the following procedure to bind the ports.

To create a receive port for ActionActivationPort binding

1.   In BizTalk Explorer, create a new one way receive port do the following:

Use this	To do this
Name field	Type **HwsActivatePort**
Authentication	Click **Not Required**
Tracking Type	Click **<None>**
Inbound Maps	Click **No maps**

2.   Add a new receive location for HwsActivatePort and do the following:

Use this	To do this
Name field	Type **HwsHttpActivateReceiveLocation**
Transport Type	Click **HTTP**

3.   In the Address (URI) field, click the ellipsis button and in the **HTTP Transport Properties** dialog box, do the following:

Use this	To do this
Virtual directory plus ISAPI extension	Type **/HwsMessages/BtsHttpReceive.dll?Activate**
Public address	This is optional and you may leave this blank. Optionally, you may type this value: **http://<computer name>/HwsMessages/BtsHttpReceive.dll?Activate**, where *<computer name>* is the computer that runs HWS.
Return content type	Click **text/XML**
Return correlation handle on success	Click **True**

4.   Click **OK**.

5.   In the Receive Handler field, click **BizTalkServerIsolatedHost** and click **OK**.

6.   In BizTalk Explorer, bind the ActionActivationPort of the HWS action to HwsActivatePort.

For more information about configuring HTTP receive locations, see Configuring an HTTP Receive Location.

To create a physical receive port for ActionInterruptPort binding

1.   In BizTalk Explorer, create a new one way receive port do the following:

Use this	To do this
Name field	Type **HwsActivatePort**
Authentication	Click **Not Required**
Tracking Type	Click **<None>**
Inbound Maps	Click **No maps**

2.   Add a new receive location for HwsInterruptPort and do the following:

Use this	To do this
Name field	Type **HwsHttpInterruptReceiveLocation**
Transport Type	Click **HTTP**

3.   In the Address (URI) field, click the ellipsis button and in the **HTTP Transport Properties** dialog box, do the following:

Use this	To do this
Virtual directory plus ISAPI extension	Type **/HwsMessages/BtsHttpReceive.dll?InterruptAndResponse**
Public address	This is optional and you may leave this blank. Optionally, you may type this value: **http://<computer name>/HwsMessages/BtsHttpReceive.dll?InterruptAndResponse**, where *<computer name>* is the computer that runs HWS.
Return content type	Click **text/XML**
Return correlation handle on success	Click **True**

4.   Click **OK**.

5.   In the Receive Handler field, click **BizTalkServerIsolatedHost** and click **OK**.

6.   In BizTalk Explorer, bind the ActionInterruptPort of the HWS action to HwsInterruptPort.

If the action receives Hws_Task response messages, then bind the logical ports in the action to ActionInterruptPort configured previously. You do this because HWS uses the same receive location for interrupt and task response messages. For more information about binding ports, see <u>Port Bindings</u>.

Binding the Action

Bind the receive port that will receive your task responses to the Interrupt and task response receive location of the HWS system. In order to maintain a secure system, all task responses must submit via the SendTaskResponse method of the HWS Web service, then the Web service delivers the message to the appropriate orchestration via the interrupt receive location.

Binding File

You can reuse physical receive port definitions across multiple HWS actions. By saving the physical port definitions in a binding file and reusing it, you can automate the creation and binding of these ports during deployment of the HWS actions. For more information about creating a binding file, see **Exporting Assembly Bindings Using the BizTalk Deployment Wizard**.

Deploying and Registering Actions

After you add logic to your action, you deploy the action in the same way as any orchestration. For more information about deploying orchestrations, see **Deploying Assemblies**.

After you deploy an HWS action and bind all its logical ports, then use the BizTalk Management console to enlist and start the action in your BizTalk host application. For more information about enlisting and starting orchestrations, see <u>Enlisting an Orchestration with the BizTalk Administration Console</u> and <u>Starting an Orchestration with the BizTalk Administration Console</u>.

Finally, use the HWS Administration console to register these actions with the HWS system for constraint creation. For more information about constraints, see <u>About Action Constraints</u>.

Note If you un-deploy an action without un-registering from the HWS Administration console will result in runtime errors when someone tries to instantiate them.

For more information about registering and Unregistering an action, see <u>Registering and Unregistering Actions</u>.

If you un-register an action:

- You would not be able to active this action through the Web service.

- The Web service would not respond to tasks.

- Tracking information for that action may not appear in the tracking database.

Creating Dependent Actions

You can create actions that have dependencies on other actions (known as dependent composition), either in an unplanned fashion or by creating a dependent transition in an activity model. In this section, the "child" action is the action with a dependency on another action that is the "parent" action. When you dependently compose actions, the child action runs automatically when the parent action sends a special type of message, without requiring manual user intervention. This is a useful feature when a user wants to link several actions that they did not activate individually.

Design the Synchronize Schema

In order to support dependent composition, an action must both send and receive a synchronization message. A synchronization message is a message that contains a schema that conforms to the Hws_Synchronize schema. This schema is much simpler than the HWS_Task and HWS_Activate schemas, although it does have a similar structure with an HwsSection block, ActionSection block, and Payloads block. You should not modify the HwsSection and you should place any new elements you require in the ActionSection block. The Payloads block should carry auxiliary data that can be unrelated to the action, but should be carried through and attached to all outgoing messages (if the action receives the synchronization message).

By default, when you create a new HWS action project, the action has an HWS_Synchronize.xsd file. This is the default synchronization message for the action and, by default, the action template is designed to receive a synchronization message of that schema. The most important thing to understand about synchronization messages is that in order for two actions to be dependably compose-able, the parent must send and the child must receive the same schema, with the same namespace.

This presents a problem if you deploy multiple schemas to BizTalk Server that use the same namespace because you will create errors at run time. There are two solutions: either create a separate project for the synchronize message schema and create actions that reference the schema, or create the schema in one action and have the other action reference the assembly for that action.

Send and/or receive the synchronize message

After you design your synchronize message schema, you must send and/or receive a message of that schema type. If you only want your action to be able to be composed as a child of a dependent link, then you only have to support the receipt of this message. If your action is to be able to be composed as the parent of a dependent link, your action must send out the synchronization message.

Since actions send synchronization messages between each other not externally to a user, they are always bound to direct bound ports. In fact, if you examine the HWS template action, at the top left in the ports user interface space, you will see that the action template already has direct bound In and Out ports identified as SendOrReceiveSyncMessage already set up. All that is required is that you send a message that fits your synchronize schema through the Out port, and receive that message type through the "In" port.

Annotate your action

In order for HWS to recognize which synchronization messages your action sends and receives, you must annotate the schema of your actions activate message.

To annotate the schema

1.   In Visual Studio.NET, open the activate schema and edit the properties of the <schema> node.

2.   You will see in the HWS properties an "Incoming Sync messages" property and an "Outgoing Sync messages" property. Edit these properties by clicking on the text field of the property and then clicking the ellipses button that appears to the right.

3.   In the list box that appears, list the namespace of each synchronization message your action can receive in the Incoming property and the namespace of each synchronization message your action can send in the Outgoing property.

4.   After you complete the steps in this section, your action will be able to be dependably composed with other actions that share its synchronization message.

5.   An important final step is to register your action using the HWS Administration console. For more information about registering actions, see Registering and Unregistering Actions. The registration process updates the HWS system about the synchronization message characteristics of an action. If you do not run the registration process after deploying your action after having made the synchronization message changes, HWS will not detect that your message supports dependent composition with other messages.

Correlating Actions with Tasks

When you add an action to an activity flow, such as creating an action instance by calling **AddActionToActivityFlow** for example, HWS generates a new task and gives it a unique TaskID GUID. You can get the TaskID by querying the HWS tracking information by calling **GetActivityFlowInfo**.

HWS uses an HWS_Task message schema to represent an outbound task message and the inbound response message. The HWS tracking system determines whether the message is a task or a response depending on the direction in which it is going.

When an orchestration sends a task message and when the response message returns to the orchestration from recipient of the task, HWS records the messages in

HWS tracking tables. Each task message and the corresponding response correlate to the TaskID.

You can sequentially serialize sending and receiving task messages and responses. Your orchestration sends a task message and blocks until it receives a response, once unblocked it would send another task message and block for the response, and so on. Alternatively, your orchestration could send out multiple task messages first and the wait for the responses to return all at once. If you need to receive more than one response but you do not care about the order in which they return, you can do this with a listen shape that you place in a loop.

Note   You cannot dynamically add branches to a listen shape.

To configure the listen shape, you need to know at design time how many response messages the orchestration will receive.

About Action Templates

You create actions using an action template. This section describes action templates in detail. After you create a new HWS project in Microsoft Visual Studio® .NET (see Creating an HWS Project), you will see an action template, which is an orchestration that implements your action logic.

This section contains:

- Action Template Layout

- Action Template Correlation Types

- Action Template Ports

- Action Template Exception Handling

Action Template Layout

An action template outlines the structure to which the orchestration must conform. You use the action template to insert your custom business logic into the **DoAllActionSpecificLogicHere** group shape of the orchestration. The following figure shows the layout of an action template. The send or receive connectors for messages sent or received by the template are omitted for clarity.

HWS action template layout
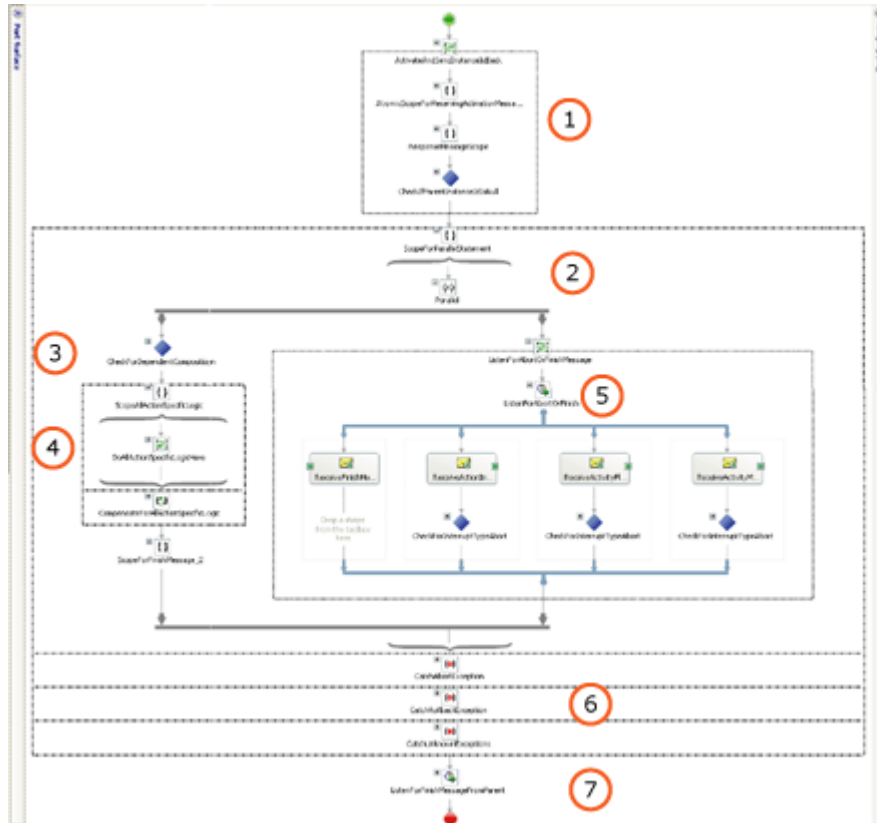
The following explains items 1 to 7 from the figure above:

1.   The action receives an Hws_Activate message over a one-way port that is bound to the HTTP transport. It then creates an instance of the Hws_ActivateResponse message. You use this message to initialize correlation sets used later in the orchestration. To do this the orchestration sends the message to itself over a direct bound port. It uses the SendOrReceiveActivateResponse operation of the ActionDirectBoundOutPort port to send the message and receives it back using ActionDirectBoundInPort port. This section of the template also checks to see if the value for *ParentActionInstanceID* and *ActivityModelInstanceID* are empty globally unique identifiers (GUID). If so, it generates new GUID values for these properties. These values will be empty GUIDs if the action being instantiated does not have a parent action or if the action is not being instantiated as part of an activity model. If there were many such action instances, an equal number of non-unique subscriptions for Hws_Finish and Hws_Interrupt message types would be created. This can affect the overall performance of the system if the numbers grow large. Generating new GUID values prevents creation of non-unique subscriptions and is safe because you do not expect fulfillment of these subscriptions, anyway. Subscription uniqueness guarantees that the routing performance of these messages for valid subscribers does not degrade.

2.   The parallel statement has two branches. The left branch allows for dependent composition of the action to another (tag 3) and provides the site for custom development within the template (tag 4). The right branch allows the action to listen for Hws_Interrupt and Hws_Finish messages (tag 5).

3.   The decision shape checks to see if action activation occurred with the intention of composing it as dependent on another action. It checks for the **IsDependentOnParent** promoted property in the activation message. If so, the action waits for receipt of an Hws_Synchronize message or Hws_Finish message from the parent action. The Receive shapes for the Hws_Synchronize message and the Hws_Finish message use correlation sets based on the instance ID of the parent action. Upon receipt of the Hws_Synchronize message, the execution proceeds to ScopeAllActionSpecificLogic. A receipt of the Hws_Finish message instead of an Hws_Synchronize message from the parent action causes this action to terminate. In this case, before terminating the action sends out an Hws_Finish message to the MessageBox over a direct bound port to indicate its completion.

You can extend the correlation set used on the Hws_Synchronize message to include additional properties promoted out of the Hws_Activate message. Additional correlation sets can be used as well on the receive shape for the Hws_Synchronize message.

4.   The Decision shape (tag 3) collapsed in the previous figure expands to the following:

5. **HWS Action**
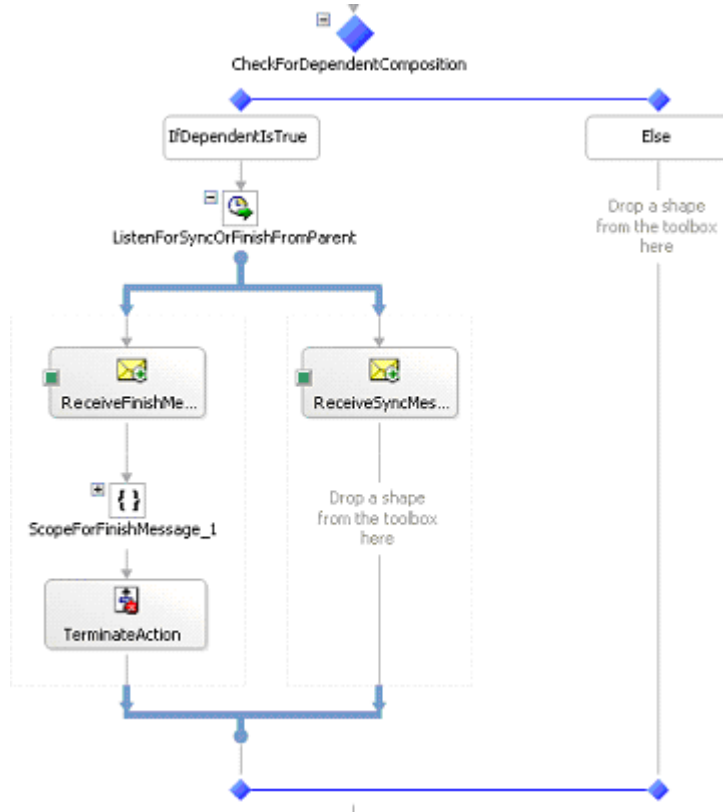
6. This is the site for custom development within the action template. There is a transactional scope and a compensation block defined around the site. This allows you to build custom compensation logic if the action or an Hws_Interrupt message s sent, or if an unknown exception occurs in the action.

7. This section of the action listens for an instance of the Hws_Interrupt message based on multiple subscriptions. It also listens for an Hws_Finish message from itself.

    The action subscribes for the Hws_Interrupt message at three levels of granularity—action instance level, activity flow level, and activity model level. An interrupt message may request an Abort or a Rollback of the action's operations. Receipt of the interrupt message causes an exception of Abort or Rollback to occur within the action. These exceptions are caught within the template (tag 6), and the exception handler calls compensate for the **ScopeAllActionSpecificLogic** scope (tag 4).

    The other message in the listen block is the Hws_Finish message from itself. The action generates this message and HWS sends it to the MessageBox through a direct-bound port in the left branch of the parallel statement that we describe in section 2. This occurs after the **ScopeAllActionSpecificLogic** scope (tag 4) completes. A receipt of this message in the right side branch causes the listen shape **ListenForAbortOrFinish** to complete and the branch to end.

8.   This section of the action contains exception handlers for the Abort, Rollback, and Unknown exceptions.

The action generates the Abort and Rollback exceptions upon receipt of the Hws_Interrupt message. The exception handler for Abort exception builds and sends out the Hws_Finish message indicating that the action is completing and then enters the terminate state. The exception handler for a Rollback exception calls the compensation block **CompensateForAllActionSpecificLogic** before building and sending an Hws_Finish message. It then enters the terminate state.

- The exception handler for Unknown exceptions also calls the compensation block **CompensateForAllActionSpecificLogic** before building and sending an Hws_Finish message and terminating.

Each exception handler sends out an Hws_Finish message to the MessageBox over a direct bound port. This message instructs other dependently composed child actions that have not received an Hws_Synchronize message to terminate. Tag 3 is where dependently composed child actions receive the Hws_Finish message.

9.   This section of the action listens for an Hws_Finish message from a parent action. If the action does not receive a message after five seconds, then a timeout occurs and the action completes. This happens so that the action consumes an Hws_Finish message from a parent action. This condition arises if the parent action sends an Hws_Synchronize message and sends an Hws_Finish message in quick succession. the dependent child action in the listen shape (tag 3) only consumes the Hws_Synchronize message. The child action receives, but does not consume the Hws_Finish message because the listen shape accepts only the first of the two messages. If the child action completes without consuming this message and this pattern continues for other instances as well, a number of orphaned message entries would build up in the MessageBox database. This can adversely affect the overall performance of the system. Having a listen shape avoids the action completing without consuming the Hws_Finish message.

Action Template Correlation Types

The action template has the following six correlation types defined and used within the orchestration:

ActionInstanceInterruptCorrelation

The **ScheduleInstanceID** promoted property in the Microsoft.BizTalk.Hws.HwsPromotedProperties.dll assembly defines this correlation type. The action orchestration defines a correlation variable of this type called **corActionInstanceInterruptCorrelation**. This correlation initializes upon receipt of the Activation message with the **ActionInstanceID** property defined in the HwsSection of the message. You use this variable to correlate the receipt of ActivateResponse and Interrupt messages within the action orchestration.

ActivityFlowInterruptCorrelation

The **ActivityFlowID** promoted property in the
Microsoft.BizTalk.Hws.HwsPromotedProperties.dll assembly defines this correlation
type. The action orchestration defines a correlation variable of this type called
corActivityFlowInterruptCorrelation. This correlation initializes when sending the
ActivateResponse message with the **ActivityFlowID** property defined in the
HwsSection of the message. You use this variable to correlate the receipt of an
Interrupt message within the action orchestration.

ActivityModelInstanceInterruptCorrelation

The **ActivityModelInstanceID** promoted property in the
Microsoft.BizTalk.Hws.HwsPromotedProperties.dll assembly defines this correlation
type. The action orchestration defines a correlation variable of this type called
corActivityModelInstanceInterruptCorrelation. This correlation initializes when
sending the ActivateResponse message with the **ActivityModelInstanceID**
property defined in the HwsSection of the message. You use this variable to correlate
the receipt of an Interrupt message within the action orchestration.

ParentFinishCorrelation

The **ParentInstanceID** promoted property in the
Microsoft.BizTalk.Hws.HwsPromotedProperties.dll assembly defines this correlation
type. The action orchestration defines a correlation variable of this type called
corParentFinishCorrelation. This correlation initializes when sending the
ActivateResponse message with the **ParentActionInstanceID** property defined in
the HwsSection of the message. You use this variable to correlate the receipt of
Finish message from a parent action.

SelfFinishCorrelation

The **ScheduleInstanceID** promoted property in the
Microsoft.BizTalk.Hws.HwsPromotedProperties.dll assembly defines this correlation
type. The action orchestration defines a correlation variable of this type called
corSelfFinishCorrelation. This correlation initializes when sending the
ActivateResponse message with the **ActionInstanceID** property defined in the
HwsSection of the message. You use this variable to correlate the receipt of Finish
message from itself.

SyncCorrelation

The **ParentInstanceID** promoted property in the
Microsoft.BizTalk.Hws.HwsPromotedProperties.dll assembly defines this correlation
type. The action orchestration defines a correlation variable of this type called
corSyncCorrelation. This correlation initializes when sending the ActivateResponse
message with the **ParentActionInstanceID** property defined in the HwsSection of
the message. Use this variable to correlate the receipt of Synchronize message from
a parent action.

Custom Correlation Types

You can define other correlation types based on promoted properties within the messages sent or received by the action. You can extend the **SyncCorrelation** type to include additional properties that you add and promote out of the Synchronize message during action development.

Action Template Ports

The default action template has the following four orchestration ports defined:

ActionActivationPort

This is a one-way port for receiving the activation message. This port defines an operation called **ActivationOperation**, which subscribes to the HWS activate message. The port is internal and is late bound. At deployment, this port should be bound to HTTP transport having /HwsMessages/BtsHttpReceive.dll?Activate as the primary receive location.

ActionInterruptPort

A one-way port exists to receive an interrupt message that causes an abort or rollback to happen in the action. This port has an operation called **InterruptOperation** defined on it that subscribes to the HWS interrupt message. The port is private and is late bound. At deployment, this port should be bound to HTTP transport having /HwsMessages/BtsHttpReceive.dll?InterruptAndResponse as its primary receive location.

ActionDirectBoundInPort

This is a one-way direct bound port for receiving Finish, Synchronize and ActivateResponse messages. The port defines **SendOrReceiveFinishMessage**, **SendOrReceiveSyncMessage**, and **SendOrReceiveActivateResponse** operations for sending these messages. The port is private and is marked for ordered receipt of messages. This port is bound to the MessageBox database for receiving subscribed messages.

ActionDirectBoundOutPort

This is a one-way direct bound port for publishing Finish, Synchronize and ActivateResponse messages to the MessageBox database. The port defines **SendOrReceiveFinishMessage**, **SendOrReceiveSyncMessage**, and **SendOrReceiveActivateResponse** operations for receiving these messages. The port is private and is marked for ordered delivery of messages. This port is bound to the MessageBox database for publishing messages to it directly.

Custom Ports

You can define other ports as necessary to send task messages and receive responses or other application specific messages. Bind the defined ports for receiving response messages to the HTTP transport that has /HwsMessages/BtsHttpReceive.dll?InterruptAndResponse as the primary receive location.

Action Template Exception Handling

In BizTalk Server, if the MessageBox database receives a message from an orchestration and there are no subscribers for that message, then the orchestration receives a persistence exception of type **Microsoft.XLANGs.BaseTypes.PersistenceException** that wraps another exception of type **Microsoft.XLANGs.BaseTypes.PublishMessageException** within it.

A persistence exception may occur for a variety of reasons, but in scenarios where the orchestration would like to ignore this exception if there were no subscribers for the message it sent then it should inspect the inner exception to be of type **PublishMessageException**.

Implement the following sample code inside the exception catch block for type **PersistenceException** as shown:

The HWS action template uses this logic for an HWS_Finish message Send in the **CatchAbortExceptionBlock**.

The Send shape in this case is inside a scope called **ScopeForSendingFinishMessage** and has an exception handler block associated with it called **CatchPersistenceException**. Inside this exception handler block, a check for the inner exception occurs and if the type turns out to be **PublishMessageException**, the exception is ignored; otherwise, it is re-thrown.

Ignoring the **PersistenceException** if the inner exception is **PublishMessageException** may be necessary in scenarios where an HWS action sends out an Hws_Sychronize message type that does not have any subscribers. In such scenarios, the sending orchestration may want to ignore the **PersistenceException** if the inner exception is **PublishMessageException**.

This, however, does not mean that you should ignore **PeristenceExceptions** due to missing subscribers for an application-specific message. It shows how to inspect the exception and take appropriate action as necessary.

About Action Constraints

You use constraints to secure and guide workflow actions in Human Workflow Services (HWS). Using constraints, an HWS administrator can set policies on who can perform actions, as well as who can be targets of these actions. Constraints define the security policy for an action in an HWS deployment.

There are two types of constraints, role constraints and target constraints.

- Role constraints relate to constraints on Activity Models. For more information about role constraints, see Role Constraints.

- Target constraints are a set of clauses, which define who can be a target of an action. For more information about target constraints, see **Step.AddTargetConstraint Method**.

You represent constraints in the following form:

Source Clauses

Source clauses are composed of a set of zero or more clauses that define the set of source actors. Source actors are those actors that can initiate the action.

Action

Action is the identifier for the action to which this constraint applies.

Enacted Clauses

Enacted clauses are composed of a set of 0 or more clauses that define the set of Enacted actors. The enacted users are those on which a transitive business action acts. Examples of this concept are the delegate or escalate action in workflow. When someone does not complete their task, you would escalate the task to their manager. You would do this by initiating the escalate action on the person and target their manager. In this example, the person is the enacted on user while their manager is the target. Using the Enacted clauses, an administrator could set a constraint that would define that when you escalate someone's task, it must go to their manager. This policy enforces the targeting of the appropriate person to rectify the situation and acts as guidance to those using the system.

Target Clauses

Target clauses are composed of a set of zero or more clauses that define the set of target actors. Target actors are receive tasks.

Clause Format

Each clause in a constraint is in the form of:

Fact Retriever properties are data from Fact Retrievers. For more information, see About Fact Retrievers.

The following is an example of a clause:

Number	Source	Action	Enacted On	Target
1	User.Group = "Finance"	Approve		User.Role = "Manager" And User.Level > 5
2	<Allow All>	Escalate	<Allow All>	User.DirectReports contains EnactedOn.UserName

1.   This constraint defines a policy that states that users in the "Finance" group can initiate an approval action and that the target must be someone in the "Manager" role with a level greater than five.

2.   This constraint defines a policy that states that everyone can initiate an escalation on everyone else, and the escalation action's target must go to the target's manager.

The three supported roles are the Initiator (who can create an instance of the activity model), the Observer (who can have visibility into existing activity model instances), and the Owner (who has an elevated level of authority into existing runtime activity model instances.

Since the Approval action is not a transitive action, there is no need to define any clauses for the EnactedOn user.

When constraints are added/modified/removed, once the OK button is pressed, the WMI components will alert the HWS Web Service of the change and it will update itself with the new constraint information. This allows the constraint evaluation against the most up to date information in the Administration database, thereby allowing real time policy enforcement. In fact, every time the fact retrievers have a constraint to evaluate, they retrieve user data to ensure that they evaluate constraints against the most up to date information.

This section contains:

· Multiple Constraints

· Positive and Negative Constraints

· Activity Model Constraints

· Constraints Interaction with Fact Retrievers

· Constraints Data Types

Multiple Constraints

You can define more than one constraint for a given action. When there is more than one constraint for a given action, both constraints are enforced like in the following two example constraints:

The first constraint means that if an actor is in the "Finance" group, they can initiate an approval action to someone in the "Manager" role. The second constraint means that if an actor is in the "Marketing" group, they can initiate an approval action to someone in the "Group Manager" role. However, if an actor is in the "Finance" and "Marketing" groups, they can initiate an approval action to someone in the "Manager" or the "Group Manager" role. Essentially, each constraint is evaluated and for each successful evaluation, the target sets are combined (OR operation) together.

Positive and Negative Constraints

Constraints can be positive or negative and they both have the same form. When constraints are evaluated, negative constraints have precedence over positive constraints. If there is a constraint preventing an actor from performing a specific action or targeting a specific actor, and there is a constraint allowing this same action, then the negative constraint will prevail and the actor cannot perform the specified action. For example:

Positive Constraint

Negative Constraint

In this example, even if Bob was in the Finance group, a constraint states that he cannot initiate the approval action targeting anyone. Therefore, even though he satisfies the positive constraint, he cannot initiate the approval action. Additionally, another negative constraint states that Jim cannot be the target of the approval action. Therefore, even if Jim is in the "Manager" role, he cannot be the target of the approval action

You can specify this set of users as the "Approving Managers" group. This group has no meaning outside of the HWS system. The user system in the deployment only has the notion of roles and levels, but nothing to correlate them. The target group allows the system to define a group to define Managers with a level of greater than five known as "Approving Managers". You can use this target group to initiate an action through the HWS Web service. The HWS Web service returns these groups when it receives a query for a set of targets, and when the constraint for the target group is set to **true**.

The constraints explained so far are base system constraints and are applied for every action performed in the HWS environment. Activity model related constraints are explained below.

Activity Model Constraints

Constraints can also be set specifically to individual activity models. There are three forms of activity model related constraints:

- Step constraints

- Activity model specific system constraints

- Activity model role constraints

Step constraints define who can be the targets of each step in the activity model. Step constraints only have a step ID and a set of target clauses. This defines the set of users who are valid targets for the respective step.

Activity model specific system constraints are the same as system constraints but they only apply for the specific activity model. For a specific action, if there is an activity model specific system constraint, it will be used instead of any base system constraints when the action is performed within the context of that activity model. An example of this is:

Activity Model role constraints define:

- Initiator Constraints – set of actors that can initiate the respective Activity Model.

- Owner Constraints– the set of actors which have Owner rights to the Activity Model (see Activity Model documentation)

- Observer Constraints – set of actors which have Observer rights to the Activity Model (see AM documentation)

Role constraints consists of a single set of clauses which when evaluated define the resulting set of actors. The following is an example role constraint for the Budget Approval Activity Model:

For the constraints above, if the escalate action is initiated in the context of the specified activity model, the target must be someone in the "Director" role. If the escalate action is performed outside the context of this activity model, the target must be someone in the "Manager" role.

For step execution in Activity Models, both the step constraints and the System constraints for the action for which the step contains are evaluated. If there are Activity Model specific system constraints for that action, it will be used instead of the base system constraints. Otherwise, the base system constraints will be used. For initiation of actions within an activity model, activity model specific system constraints will be used if they exist, otherwise, the base system constraints will be used.

Constraints Interaction with Fact Retrievers

Constraints use facts from the fact retrievers. Fact retrievers let the HWS constraints service to use any user data in the organization to create constraints. Whether user data exists in Microsoft stores (SQL, AD, Passport..), third party stores (Lotus Notes, Oracle) or proprietary stores (XML file, custom DB implementation), a Fact Retriever can be implemented to retrieve this data and allow the Constraint Service to use it to evaluate constraints. You can enable multiple fact retrievers within an HWS deployment and you can create constraints that reference facts from different fact retrievers. For example:

If the Role fact was retrieved from Fact Retriever1 and Level was retrieved from Fact Retriever2, then resulting set would be the intersection of the users obtained from each Fact Retriever. If in User Store1 which is queried by Fact Retriever1, Bob, Jim, and Sandy are in the manager role, and in User Store2 which is queried by Fact Retriever2, Tom, Jim, Sandy, and Bill have a level grater than 5, then resulting set from this constraint would be Jim and Sandy because they are the only ones which satisfy both clauses.

If the HWS deployment has no Fact Retrievers enabled, then constraints can still be created using facts from the Intrinsic Fact Retriever. It exposes the following properties:

- User Name

- Day of week (note that 0 is Sunday)

- Time of day

- Current Time

- Current Date

Constraints Data Types

The following data types are used for constraints in HWS. Use this information when converting constraint data types to other formats such as SQL and Win32 data types.

- **Int** - 32 bit integer

- **Real** - 32 bit floating point number

- **DateTime** - determined by the user's locale settings, for example in *yyyy/mm/dd hh:mm:ss*

- **Date** - determined by the user's locale settings, for example in *yyyy/mm/dd* format

- **Time** - in *hh:mm:ss* format

- **UserID** - 256 unicode characters

- **String** - 256 unicode characters

- **Boolean** - True/False

About Activity Models

Activity models are the HWS construct for a predefined workflow. Use activity models to guide or enforce a particular order to a series of actions in a business process. While some processes are very difficult to pre-define, many others do follow a general structure. HWS supports this structure that provides:

- Metadata to better inform the end user on possible next steps

- Fine-grained control of the process flow and its participants

- The ability to combine multiple actions that normally run individually in a dynamic workflow, into a single block of execution that maintains the order

Create and manipulate activity models using the activity model designer API, which you can find in the Microsoft.BizTalk.Hws.WorkflowDesign.dll in the Developer Tools directory of your installation.

For activity model code samples, see Activity Model Designer API and Client Component (BizTalk Server Sample).

There are four major classes in the object model exposed by this API, the **DesignManager** class, the **ActivityModel** class, the **Step** class, and the **Transition** class.

- **DesignManager class**. The **DesignManager Class** provides the environment in which you create activity models. It must be connected to a specific HWS installation to be useful, and once connected, provides information from the HWS system about available actions, activity models, fact retrievers, etc. Use this class to construct the other classes in the object model.

- **ActivityModel class**. The **ActivityModel Class** represents a single activity model. It provides all the activity model wide properties, and serves as a factory/container for the Step and Transition classes that form an activity model.

- **Step class**. The **Step Class** represents the occurrence of an action within an activity model. Each instance of the class is associated with an action type, and it also carries secondary data about the step in the activity model, such as any default parameters and constraints.

- **Transition class**. The **Transition Class** creates the ordering between steps in an activity model, and represents a flow of control from one step to another in an activity model.

This section contains:

- Developing Activity Models

- Dependent and Independent Transitions

- Metadata

- Saving and Retrieving Activity Models

- Step Constraints

- Role Constraints

- Loops

- Default Parameters

Developing Activity Models

The following outlines the development process of an activity model:

- **Create an instance of the DesignManager class**. In order to create a simple activity model, the first thing you must do is create an instance of the DesignManager class. As mentioned previously, this class gathers information about the environment in which you are creating your activity model. Once you create the class, you call the Connect method in order to attach the design manager to your specific HWS installation. None of the other class methods will succeed if the DesignManager is not connected.

- **Create the activity model class**. The next step is to create the activity model class. The DesignManager class serves as a factory for the ActivityModel class, and the only way to obtain a reference to an ActivityModel class instance is by calling one of the Load methods of the DesignManager class (to load a pre-existing activity model), or by calling the CreateActivityModel method to create a new, blank activity model.

- **Add and connect steps to the ActivityModel instance**. Once you have a reference to an activity model, the creation of an activity model involves adding steps to the ActivityModel instance (via the AddStep method) and connecting the created steps in a sequence by adding transitions between the steps (via the AddTransition method). To add a step to an activity model, you must know the Id of the action type you want to associate with that step. The easiest way to obtain a list of actions and their corresponding Ids is to call the GetActionCatalog method on the DesignManager class. The out parameter is an array of all action type registered with the connected HWS system.

Each new step added to the activity model gets an ID. That ID is unique within the activity model, but not unique among all activity models. The id property is read-only, and you use it to refer to that step in various places through the activity model

designer API. The two most prominent places are in the Transition class, and the StepDictionary class. The StepDictionary class is a type specific dictionary class that contains all steps present in the activity model (you can retrieve it by a call to GetSteps). HWS indexes the dictionary by step id, so whenever you have a step Id, and you want the entire Step class for that Id, you simply access the StepDictionary for the activity model at that particular index.

When you add a step to an activity model, it has no initial connection to any Transition objects. You refer to any step that is not the child of any transition objects in the activity model as a root step (it is the root of a tree of one or more actions). The method GetRootActions on the ActivityModel class will return an array step Ids that list all steps not currently the child of any transitions in the activity model.

Dependent and Independent Transitions

There are two different types of transitions in the HWS activity model system, independent and dependent. Independent transitions are transitions that must by explicitly followed by a user calling the HWS Web service AddActivationBlockToActivityFlow method, passing the appropriate parent information and the Id of the step which is the child of the independent transition.

Dependent transitions are transitions that the HWS system automatically follows when the parent action has reached a certain point and signaled this by sending a sync message (See the Creating Dependent Actions topic for more information on sync messages). In order for this to succeed in HWS, you must activate both the parent and child actions of a dependent transition at the same time. Depending on your particular process, you may need on or the other, or both types of transitions throughout your activity model.

By linking two actions with a dependent transition, you create an "activation block". Actually, every step you create in an activity model is an activation block; however, by linking two steps via a dependent transition, you create a non-trivial activation block. HWS defines an activation block as an action that is either a root step, or the child of an independent transition, and all steps linked by a chain of one or more dependent transitions to that first step. In simple terms, an activation block is a collection of actions that must activate at the same time since the parent and child of a dependent transition activate at the same time. It is important to realize, then, that the user who activates that first step in an activation block will have to supply the parameters for the activation message for all steps contained within that activation block.

One restriction present in the HWS activity model is that a step may be the child of at most one non-looping transitions (Looping transitions are discussed below). This means that while a single step can branch out to multiple "branches" of the activity model "tree", it will not be possible to bring those branches back together with transitions to a single step. It is certainly possible to simulate such behavior through some logic inside the action, just not explicitly through designing an activity model.

Metadata

Both the ActivityModel class and the Step class have fields on them to store some basic metadata about the specific occurrence of that class in the activity model. Both classes support a Name and Description field that returns from some of the HWS Web service calls to help give the user context surrounding the information retrieved from the Web service.

Along with Name and Description, the ActivityModel class also contains an Annotation field, which is a string property that may have any data placed in it that you may choose. The data of the Annotation field will be stored along with the activity model, but does not affect how the activity model performs. This field carries such things as layout information for a visual designer application built around the activity model designer API, or perhaps propriety meta-data.

Saving and Retrieving Activity Models

Once you have completed the creation of your activity model, now you save the activity model to your HWS system in order for your users to be able to initiate the activity model. You achieve this is achieved the SaveActivityModelToSystem method on the DesignManager class, which will store the activity model passed into the method in the connected HWS System.

If you wish to retrieve an activity model that resides on the system, you can use the LoadActivityModelFromSystem method to generate an instance of the ActivityModel class with all the data for the specified activity model.

It is possible to stop development on an activity model before the activity model is complete and store the activity model in a temporary file so that work can be continued on the activity model at a later date by using the SaveActivityModelToFile method of the DesignManager class. To resume work on the activity model, simply load the model back from file using the LoadActivityModelFromFile method.

Step Constraints

Step constraints are an extension of the HWS constraint system (See the Setting Up Action Constraints topic) that allows an even finer grained level of control when working within an activity model. HWS system level constraints allow you to restrict or guide the set of users that can initiate or be a target of a certain type of action. Step level constraints enable you to restrict of guide the set of users that you can choose as targets for each step. For example, in an expense report process, you could restrict the initial expense approval step's targets to be a user with a manager level in the org chart. Constraints only affect the steps to which you apply them, even if there are other steps in the activity model that reference the same action type.

Unlike system level constraints, however, it is not possible to grant more authority with step constraints than a user already has. Step constraints evaluate along with system constraints, and only users who satisfy both types of constraints for a given step can be targets.

Role Constraints

Role constraints are a little different from the other types of constraints you have seen to this point. Role constraints do not relate to specific action types or steps, they apply to an entire activity model. In HWS, users fit three special roles.

The roles for an ActivityModel are:

Initiator: A user who can start the ActivityModel.

Owner: A user who is the owner for an ActivityModel. Owner can extend an activity flow by adding an action from a step in an activity model. Owner can also execute a step in an activity model.

Observer: A user who can view the activity flow. An observer of an activity model need not be a participant in the activity flow to view it.

Roles contain users who may be involved in an activity model without necessarily being a participant in the activity model. For a detailed description of activity model roles, see **AddRoleConstraint**.

Add role constraints to an instance of the ActivityModel class to specify the parameters a user must meet to operate under that role's authority.

Loops

Loops may occur in pre-defined workflows. Activity models support the construction of loop transitions, unlike workflows dynamically created at runtime when loops are not possible. Loop transitions indicate that the workflow can return to a previously visited step in the process, just like normal transitions.

There are a few important things to note about looping transitions:

- When a step runs again because of a looping transition, a new instance of that step activates. The loop does not return the workflow to the previously instantiated action, as most likely that action has long since completed.

- A looping transition must be independent. It is not possible to create a dependent looping transition in HWS.

- You must independently activate the child step of a looping transition. This is different from #2 in that #2 specifies the transition that forms the loop must be independent. This rule specifies that the child step of a looping transition must activate independently by all transitions that have the step as a child. This includes the loop transition, and the original transition involving the step in the activity model.

Special Case Involving Loops

The following describes a loop situation you may encounter when you add steps to an activity model.

In this situation, assume that you add a step zero and step one to your activity model. Then you add a transition between step zero and step one and a transition between step one and step zero.

At this point, you have an activity model, which is one big loop. Step zero remains the only root step, since step zero was a root step prior to adding the second transition.

The problem is that you may want step one to be the root. You cannot simply change the root to step one. In order to resolve this situation, you must first delete the transition from step zero to step one, since the activity model is now linear and the only step that is not a child of any transitions is step one. Step one becomes the new root step. Additionally, you must re-add the transition from step zero to step one. Step one remains the root step.

Default Parameters

Default parameters are possibly the most important advanced feature of activity models. They allow the designer of the activity model to specify default values for the activation message of a step, and those default value will be communicated to the user at runtime when the users asks the HWS Web service for the parameters of the step. Users can change those default values if they choose, however if they do nothing, the default values will be returned to the system as specified by the activity model designer.

The benefit of default parameters is the ability of the designer to develop a customized activity model from a catalog of generic actions. The goal of long-term action development in HWS is to accumulate a catalog of generic, re-useable actions. The usual side effect of reusability, however, is that an action may require a large number of configuration parameters to be set. In most cases, the parameters do not relate to the business process and would confuse the end user. Using default parameters, the activity model designer can supply default values for those configuration parameters, and then design a UI that simply does not show the parameters to the user, alleviating the problem.

To retrieve or set the default parameters on a step, you use the GetParameters and SetParameters methods respectively.

About Fact Retrievers

Fact Retrievers are .NET assemblies that implement the IFactRetriever interface. You use fact retrievers to allow the HWS system to leverage user information in the organization. For example, in an organization with an existing infrastructure where user information is kept in some third party or proprietary user store, it would be ideal if the HWS system could leverage this rich information to set up constraints. By

implementing the IFactRetriever interface to retrieve this rich user data, it will allow HWS to use the existing infrastructure and easily integrate with it. You can easily implement a Fact Retriever for your custom user store, whatever form it exists.

For a fact retriever code sample, see <u>Active Directory Fact Retriever</u>.

Use constraints to filter, provide guidance, or enforce restrictions on users, or actors, within an activity flow. The fact store retriever provides the mechanism to retrieve, but not modify, facts about the actors from existing data sources. In turn, HWS provides these facts to the constraint service to enforce the constraints.

When you create a fact retriever, the only requirement is that it must implement the **IFactRetriever Interface**. This provides the Human Workflow Services (HWS) Web services a common interface for executing queries against any data source, such as Microsoft® Active Directory® directory service, Exchange store, Microsoft SQL Server™, flat-file, etc. It is possible to have many fact retrievers that use the same data source. For example, a SQL Server store may have several fact retrievers bound to it, with each fact retriever exposing a different set of properties, or facts. For sample code that demonstrates interacting with a fact retriever programmatically, see <u>Activity Model Designer API and Client Component (BizTalk Server Sample)</u>.

About Action Template Schemas

The action project template creates several schemas. This section explains the schemas and their elements.

This section contains:

- Hws_Activate Schema

- Hws_ActivateResponse Schema

- Hws_Synchronize Schema

- Hws_Task Schema

- Hws_Finish Schema

- Hws_Interrupt Schema

- Tasks for Defining and Updating Schemas

Hws_Activate Schema

You use the Hws_Activate message to supply parameters to an action during its activation. An action can have at most one task message schema associated with it. The Hws_Activate message has three child elements under the HwsMessage node. These are HwsSection, ActionSection and Payloads.

HwsSection

The HwsSection holds definitions of XML elements and attributes that HWS reserves for its own use. Do not modify elements or attributes defined under this section. The following list explains the elements/attributes defined under the **HwsSection** node.

Node Name	Node Type	Data Type	Mix/Max Occurrence	Description
ActionInstanceDescription	Element	String	1/1	The value for this element should carry a desc... instance to activate.
ActionInstanceID	Element	String	1/1	The value for this element is a GUID string th... an instance of the action to activate.
ActionProperties	Record		1/1	This node groups properties that correspond to... track.
ActionProperties\Property	Record	String	0\unbounded	This property is optional. Nodes of this typ... properties to track at the action-instance level.
ActionProperties\Property\Description	Attribute	String	1/1	Description of the property being tracked at acti...
ActionProperties\Property\Name	Attribute	String	1/1	Name of the property being tracked at action-in...
ActionProperties\Property\Type	Attribute	String	1/1	Data type of the property being tracked at actio...
ActionTypeID	Element	String	1/1	The value for this element is a GUID string that... being instantiated.
ActivityFlowDescription	Element	String	1/1	The value for this element should carry a descr... flow of which the action being instantiated is goi...
ActivityFlowID	Element	String	1/1	The value for this element is a GUID string. It ... flow to which the action being instantiated belon...
ActivityFlowProperties	Record		1/1	This node groups properties that correspond ... track, so that they HWS can use them in other a...
ActivityFlowProperties\Property	Record	String	0/unbounded	This property is optional. Nodes of this typ... properties to track at the activity flow level.
ActivityFlowProperties\Property\Description	Attribute	String	1/1	Description of the property to track at the activi...
ActivityFlowPropertie	Attribute	String	1/1	Name of the property to track at the activity-flo...

s\Property\Name				
ActivityFlowProperties\Property\Type	Attribute	String	1/1	Data type of the property to track at activity-flo
ActivityModelInstanceID	Element	String	1/1	The value for this element is a GUID string that model instance to which the initiated action belo
ActivityModelStepID	Element	Int	1/1	The value for this element is an integer that model step associated with this action. Each st model is associated with one action.
ActivityModelTypeID	Element	String	1/1	The value for this element is a GUID string that model of which the action being instantiated is a
HwsMessageType	Attribute	String	1/1	This attribute identifies the message sche Hws_Activate. The default value for the attribute
HwsWebServiceUrl	Element	String	1/1	URL of the Human Workflow Services Web servi
InitiatingActor	Element	String	1/1	The value for this element is the domain\use initiating the action.
IsDependentOnParent	Element	Boolean	1/1	This flag determines if the action being instantia synchronize message from a parent action in current action waits for a synchronize messa action before it proceeds. If false, the current for the synchronize message.
ParentActionInstanceID	Element	String	1/1	The value for this element is a GUID string that instance preceding this action in the activity flow
ParentTaskID	Element	String	1/1	The value for this element is a GUID string t from a preceding action in the activity flow. This ID that led or required the initiating actor to action.

These properties promote by default to refer to and use in the action template.

ActionSection

The ActionSection is customizable by action developers and can contain any action specific parameters and values to deliver to the action during instantiation. This section may contain elements that correspond to human targets for the action. The HWS interceptor does not track the values of elements/attributes in the ActionSection. If you have properties to track, specify them in the instance document under the pre-defined HwsSection\ActionProperties collection.

Payloads

The payloads node in the schema is a placeholder for applications to specify additional information that they may need to include in the other messages sent out by the action.

Annotations

The task message schema carries annotations used by the HWS system. These annotations are defined using HWS extensions to the Schema Editor and appear as properties in the HWS category of the properties for the schema. HWS defines some properties at the schema root-node level and some at the element-node level. For more information, see **Supplemental Node Properties for HWS Schemas**.

Schema node properties

To view these properties, open the task message schema in BizTalk Schema Editor. Select the **<Schema>** node, right-click and select **Properties**. In the Properties window, select the **Categorized** view. The following properties appear under the HWS grouping:

- **Description**: The value in this property is used to describe the action the activation message is associated with.

- **Incoming Sync messages**: This property specifies the target namespace of the synchronize messages that are received by the orchestration associated with the activation message. Select the ellipsis () button; it brings up a dialog box where you can specify the target namespace of synchronize messages received.

- **Outgoing Sync messages**: This property specifies the target namespace of the synchronize messages that are sent by the orchestration associated with the activation message. Select the ellipsis () button; it brings up a dialog box where you can specify the target namespace of synchronize messages sent.

Element node properties

For element nodes defined under the **ActionSection** element of the schema document HWS defines the following property:

- **Target**: This is a Boolean property. The value true indicates that the element node is a human target that is a recipient of one or more task messages sent by the action associated with the activation message. A false value or if a value is not specified, indicates that the node is not a human target.

Remarks

Each task message schema should have a Target Namespace that uniquely identifies it within the set of deployed actions and schemas. You should modify the default value: http://tempuri.org/Hws_Activate_Sample.

An action should not have more than one activation messages associated with it.

An instance of the activate message is received over the ActionActivationPort in the template bound to a one-way port with HTTP receive location /HwsMessages/BtsHttpReceive.dll?Activate.

A task message schema should not refer to other schemas by means of an **include**, **redefine**, or **import** statement. HWS message schemas do not support this.

Hws_ActivateResponse Schema

The action template uses the Hws_ActivateResponse message internally to initialize a set of correlation variables for receiving other messages in the template. The Hws_ActivateResponse message has only one child element under the HwsMessage node. This is the HwsSection.

HwsSection

The HwsSection holds definitions of XML elements and attributes that are reserved for use by the HWS system. Elements or attributes defined under this section should not be modified. Following is the list of elements/attributes defined under the HwsSection node.

Node Name	Node Type	Data Type	Min/Max Occurrence	Description
ActionInstanceID	Element	String	1/1	The value for this element is a GUID string that uniquely identifies an instance of the action activated.
ActivityFlowID	Element	String	1/1	The value for this element is a GUID string, and it identifies the activity flow of which the action is a part.
ActivityModelInstanceID	Element	String	1/1	The value for this element is a GUID string that identifies the activity model instance that the activated action is part of.
HwsMessageType	Attribute	String	1/1	This attribute identifies the message schema to be of type Hws_ActivateResponse. The default value for the attribute is

				Hws_ActivateResponse.
ParentActionInstanceID	Element	String	1/1	The value for this element is a GUID string that identifies the action instance preceding this action in the activity flow.

These properties promote by default for initializing the correlation sets that the Action template uses.

Annotations

The activate response message schema has annotation for Description of the message. The value for this annotation is not available for editing. For more information, see **Supplemental Node Properties for HWS Schemas**.

Remarks

The TargetNamespace of this schema is
http://schemas.microsoft.com/Hws/2003/Hws_ActivateResponse

This schema exists in the Microsoft.BizTalk.Hws.HwsSchemas.dll assembly and the action template references it.

The message exists in the action and HWS sends/receives it over a direct-bound port.

Hws_Synchronize Schema

HWS sends the Hws_Synchronize message from one action to another to unblock the execution of the receiving action. If an action is dependent on its parent, it will start executing but will wait before running any user logic until the parent sends a synchronization message. The receiving action instance is activated with the IsDependentOnParent property in its activation message set to true to be able to wait for the synchronize message. The Hws_Synchronize message has three child elements under the HwsMessage node. These are HwsSection, ActionSection and Payloads.

HwsSection

The HwsSection holds definitions of XML elements and attributes that the HWS system reserves. Do not modify elements or attributes defined under this section. Following is the list of elements/attributes defined under the HwsSection node.

Node Name	Node Type	Data Type	Min/Max Occurrence	Description
ActionInstanceID	Element	String	1/1	The value for this element is a GUID string that uniquely identifies the action instance

				that sends the synchronize message.
ActivityFlowID	Element	String	1/1	The value for this element is a GUID string. It identifies the activity flow of which the action sending the synchronize message is a part.
ActivityModelInstanceID	Element	String	1/1	The value for this element is a GUID string that identifies the activity model instance of which the action sending the synchronize message is a part.
HwsMessageType	Attribute	String	1/1	This attribute identifies the message schema to be of type Hws_Synchronize. The default value for the attribute is **Hws_Synchronize**.
HwsWebServiceUrl	Element	String	1/1	URL of the Human Workflow Services Web service

These properties promote by default for reference and use in the action template.

ActionSection

You can customize the ActionSection, and it can contain any scenario specific parameters and values that you want to deliver to the receiving action.

Payloads

The payloads node in the schema is a placeholder for applications to specify additional information that they may need to include in the other messages sent out by the action.

Annotations

The synchronize message schema has an annotation for Description of the message. This annotation is defined using HWS extensions to the Schema Editor and appears as a property in the HWS category of the properties for the schema. This property is defined at the schema root node level. For more information, see **Supplemental Node Properties for HWS Schemas**.

Schema node properties

To view these properties, open the synchronize message schema in BizTalk Schema Editor. Select the **<Schema>** node, right-click and select **Properties**. In the

Properties window, select the **Categorized** view. The following properties appear under the HWS grouping:

- Description: The value in this property is used to describe the synchronize message.

Remarks

Each synchronize message schema should have a Target Namespace that uniquely identifies it within the set of deployed schemas. You should modify the default value: http://tempuri.org/Hws_Synchronize_Sample.

If the TargetNamespace of the synchronize message is changed or if a new synchronize message schema is added to an action, then the Incoming Sync Messages and Outgoing Sync Messages properties on the activation message (of the action that either sends this synchronize message or receives it) must be updated as well.

Synchronize messages are sent and received over direct-bound ports.

A synchronize message schema should not refer to other schemas by means of an **include**, **redefine**, or **import** statement. HWS message schemas do not support this.

Hws_Task Schema

Use the Hws_Task message schema for messages that you want to send to the participating targets of an action. An action can send task messages of one or more types. It can also send one or more instances of a given task message type. Use the Hws_Task message for submitting responses back to the action also. The Hws_Task message has three child elements under the HwsMessage node. These are HwsSection, ActionSection and Payloads.

HwsSection

The HwsSection holds definitions of XML elements and attributes that HWS reserves for its own use. Do not modify elements or attributes defined under this section. Following is the list of elements/attributes defined under the HwsSection node.

Node Name	Node Type	Data Type	Min/Max Occurrence	Description
ActionInstanceID	Element	String	1/1	The value for this element is a GUID string that uniquely identifies an instance of the action sending or receiving the task message. You should copy this node from an activation message.

ActionTypeID	Element	String	1/1	The value for this element is a GUID string that identifies the action type sending or receiving the task message. You should copy this node from an activation message.
ActivityFlowID	Element	String	1/1	The value for this element is a GUID string. It identifies the activity flow of the action sending or receiving the task message. You should copy this node from an activation message.
ActivityModelInstanceID	Element	String	1/1	The value for this element is a GUID string that identifies the activity model instance of which the action sending or receiving the task message is a part. You should copy this node from an activation message.
ActivityModelStepID	Element	Int	1/1	The value for this element is an integer that identifies an activity model step associated with this action. Each step within an activity model is unique and is associated with one action. You should copy this node from an activation message.
ActivityModelTypeID	Element	String	1/1	The value for this element is a GUID string that identifies the activity model type to which the action being instantiated belongs. You should copy this node from an activation message.
ActorElementXPath	Element	String	1/1	This is the XPath of the target actor to which you send the task message. The XPath is the value of the **Instance XPath** property for the target node that corresponds to the target actor from the activation message of the action. For more information

				about XPaths, see <u>Using XPaths in Message Assignment</u>. You create this node.
HwsMessageType	Attribute	String	1/1	This attribute identifies the message schema to be of type Hws_Task. The default value for the attribute is **Hws_Task**.
HwsWebServiceUrl	Element	String	1/1	URL of the Human Workflow Services Web service. You should copy this node from an activation message.
InitiatingActor	Element	String	1/1	The value for this element is the domain\username of the actor who initiated the action that is sending or receiving the task message. You should copy this node from an activation message.
TargetActor	Element	String	1/1	The value of this element is the ID of the actor who is receiving the task message or is sending the response for it. You create this node.
TaskDescription	Element	String	1/1	The value for this element should carry a description of the task assigned to an actor or that of the response from the actor. You create this node.
TaskID	Element	String	1/1	The value for this element is a GUID string that uniquely identifies each task that the action assigns to an actor. The same TaskID value should used in the task message that the actor sends back to the action in response. You create this node.
TaskProperties	Record		1/1	This node groups properties that correspond to a task to track. You create this optional

				node.
TaskProperties \Property	Record	String	0/unbounded	This property is optional. Nodes of this type carry values for properties to track at the task-message level.
TaskProperties \Property\Description	Attribute	String	1/1	Description of the property to track at the task-message level
TaskProperties \Property\Name	Attribute	String	1/1	Name of the property to track at the task-message level
TaskProperties \Property\Type	Attribute	String	1/1	Data type of the property to track at the task-message level.

For more information about additional nodes, such as TaskStatus, PercentageComplete, and NumberofResponses, see Adding Action Logic and Tasks.

ActionSection

The ActionSection is customizable by action developers and can contain any scenario specific parameters and values to deliver to the participating targets. You can also use it to define any parameters that the targets can supply in their responses.

Payloads

The payloads node in the schema is a placeholder for applications to specify additional information that they may need to include in the other messages sent out by the action.

Annotations

The task message schema carries annotations used by the HWS system. These annotations are defined using HWS extensions to the Schema Editor and appear as properties in the HWS category of the properties for the schema. These properties are defined at the schema root node level. For more information, see **Supplemental Node Properties for HWS Schemas**.

Schema node properties

To view these properties open the task message schema in BizTalk Schema Editor. Select the **<Schema>** node, right-click and select **Properties**. In the Properties window, select the **Categorized** view. The following properties appear under the HWS grouping:

- **Description**: Use the value in this property to describe the task message schema.

- **Target XPath**: This property specifies the XPaths of the target nodes in the activation message to whom this task message goes. Select the ellipsis () button; it brings up a dialog box where you can specify the Instance XPath values of the target nodes from the activation message. The Instance XPath property of a target node can be retrieved by opening the activate message schema and navigating to the node of interest. The Instance XPath property appears in the Properties window for that node.

Remarks

Each task message schema should have a Target Namespace that uniquely identifies it within the set of deployed schemas. You should modify the default value: http://tempuri.org/Hws_Task_Sample.

An task message schema should not refer to other schemas by using an **include**, **redifine**, or **import** statement. HWS message schemas do not support this.

Hws_Finish Schema

Actions use the Hws_Finish message internally and send it when the action completes. The Hws_Finish message has only one child element under the HwsMessage node. This is the HwsSection.

HwsSection

The HwsSection holds definitions of XML elements and attributes that are reserved for use by the HWS system. Elements or attributes defined under this section should not be modified. Following is the list of elements/attributes defined under the HwsSection node.

Node Name	Node Type	Data Type	Min/Max Occurrence	Description
ActionInstanceID	Element	String	1/1	The value for this element is a GUID string that uniquely identifies an instance of the action that was activated and is now sending the finish message.
HwsMessageType	Attribute	String	1/1	This attribute identifies the message schema to be of type Hws_Finish. The default value for the attribute is **Hws_Finish**.

Annotations

The activate response message schema has annotation for Description of the message. The value for this annotation is not available for editing. For more information, see **Supplemental Node Properties for HWS Schemas**.

Remarks

The TargetNamespace of this schema is
http://schemas.microsoft.com/Hws/2003/Hws_Finish

This schema exists in the Microsoft.BizTalk.Hws.HwsSchemas.dll assembly and the action template references it.

The message constructs in the action and sends/receives over a direct-bound port.

Hws_Interrupt Schema

The Hws_Interrupt message interrupts a running instance of an action. You can send an interrupt message to an individual action instance, to an entire activity flow or to an entire activity model instance. There are two kinds of interrupts – Abort and Rollback. The Hws_Interrupt message has only one child element under the HwsMessage node. This is the HwsSection.

HwsSection

The HwsSection holds definitions of XML elements and attributes that HWS reserves. Do not modify elements or attributes under this section. Following is the list of elements/attributes defined under the HwsSection node.

Node Name	Node Type	Data Type	Min/Max Occurrence	Description
HwsMessageType	Attribute	String	1/1	This attribute identifies the message schema to be of type Hws_Interrupt. The default value for the attribute is **Hws_Interrupt**.
HwsWebServiceUrl	Element	String	1/1	URL of the Human Workflow Services Web service
InterruptLevel	Record	N/A	1/1	Holds one of the three interrupt-level elements defined in the Choice group

InterruptLevel/<Choice>	Choice Group	N/A	1/1	Choice group that holds values for one of ActionInstanceID, ActivityFlowID or ActivityModelInstanceID
InterruptLevel/Choice/ActionInstanceID	Element	String	1/1	The value for this element is a GUID string that uniquely identifies an instance of the interrupted action.
InterruptLevel/Choice/ActivityFlowID	Element	String	1/1	The value for this element is a GUID string and it identifies the interrupted activity flow.
InterruptLevel/Choice/ActivityModelInstanceID	Element	String	1/1	The value for this element is a GUID string that identifies the interrupted activity model instance.
InterruptType	Element, Restricted	String	1/1	The value for this element is restricted to either Abort or Rollback.
RequestingActor	Element	String	1/1	The value for this element is the ID of the actor requesting the interrupt.

Annotations

The interrupt message schema has annotation for Description of the message. The value for this annotation is not available for editing. For more information, see **Supplemental Node Properties for HWS Schemas**.

Remarks

The TargetNamespace of this schema is
http://schemas.microsoft.com/Hws/2003/Hws_Interrupt

This schema exists in the Microsoft.BizTalk.Hws.HwsSchemas.dll assembly and the action template references it.

An instance of the interrupt message is received over the ActionInterruptPort in the template bound to a one-way port with HTTP receive location /HwsMessages/BtsHttpReceive.dll?InterruptAndResponse.

An interrupt message at activity-flow level or activity-model instance level submits to each action currently active in the activity flow or the activity model instance.

An interrupt type Abort causes the interrupted action to terminate without compensating for the work done by the action. A Rollback interrupt type causes the action to compensate for the work already completed by the action.

Tasks for Defining and Updating Schemas

This section describes the tasks you do to define and update schemas.

This section contains:

- Defining/Changing the Schema Target Namespace

- Updating the Action Name

- Updating the Action Assembly Name

- Defining a Human Target for an Action

Defining/Changing the Schema Target Namespace

Use this procedure to set the **Target Namespace** property.

To change the target namespace

1.   Open the schema in BizTalk Schema Editor.

2.   In the BizTalk Schema Editor dialog, right-click the **<Schema>** node in the tree view and click **Properties**. The Properties dialog appears.

3.   In the Properties dialog, the Target Namespace displays under the General category for the <Schema> node.

4.   In the Target Namespace field, type the name of your new target namespace and click **OK**.

If you are updating the target namespace of an activation message and you have target actors in it to which task messages go, then update the "Target XPaths" property on the each of those task message schemas because a change in the activation message's Targetnamespace alters the XPath to nodes marked as targets. See "Annotation" section in Hws_Task schema documentation for information on how to do this.

- If you are updating the target namespace of synchronize message, then the Incoming Sync Messages and Outgoing Sync Messages properties on the activation message of the action that either sends this synchronize message or receives it needs to be updated as well.

- If you are updating the target namespace of the Hws_Activate message that has element nodes in the ActionSection marked as Target = true, steps documented in the Notes for "Defining a human target for an action" must be adhered to.

Updating the Action Name

Use this procedure to update the action name.

To update the action name

1. In Visual St Studio.NET, open the action .odx file in the Orchestration Designer.

2. Right-click on the design surface where there is no shape underneath and click the **Properties** window.

3. In the **Properties** pane, click **TypeName** under the BizTalk category.

4. In the **Action Name** field, type the new action name and click **OK**.

Updating the Action Assembly Name

Use this procedure to update the action assembly name.

To update the action assembly name

1. In Visual Studio.NET, in the Solution Explorer window, right-click on your project node and click **Properties**.

2. In the **<Project-name> Properties** page dialog box, click **General properties** under **Common Properties**.

3. In the **Action Assembly Name** field, type the new assembly name and click **OK**.

Defining a Human Target for an Action

You can mark an element node under the ActionSection in the activation message as a human target.

To mark an element node as a human target

1.   In Visual Studio.NET, open the task message schema of the action with the BizTalk Schema Editor.

2.   Navigate to the node in the ActionSection you want to identify as the human target.

3.   Right-click the node and click **Properties**.

4.   In the **Properties** pane, click the **Categorized** view and scroll down to the HWS grouping.

5.   Under the property named **Target**, click **True**, and click **OK**.

Creating Action Template Schemas

The Human Workflow Services (HWS) schemas represent the protocol message schemas that you must implement to make use of HWS. These schemas consist of two parts, an HWS-specific part and an action-specific part.

The HWS Web service requires the HWS-specific part of the schemas to guarantee a level of service to the client applications. Do not modify this part of the schema. This part of the schema also contains promoted properties that HWS uses to correlate received messages and actions.

However, you should modify the ActionSection element to add any parameters that the action may receive as input from an actor. HWS provides sample action schemas and you must modify these for any actions you develop.

The following table lists the schemas you must customize for each action you create.

Schema name	Description
Hws_Activate.xsd	Corresponds to the activation message that initiates an action and sets up the composition type for the action. The composition type is either independent or dependent. This schema provides parameters for the action to be executed. If multiple actions share the same receive port for an activation message, then the target namespace or schema for that activation message must be unique.
Hws_Task.xsd	Corresponds to the task that HWS assigns to an actor from the action. This schema exposes all task properties to the actor.
Hws_Synchronize.xsd	Corresponds to the synchronize message that notifies an action that has dependent composition. This schema must expose all task properties that an action passes to another dependent action.

HWS does not support modification of the following schemas:

Schema name	Description
Hws_Interrupt.xsd	Corresponds to the interrupt message that an action may receive from the administrator or action owner.
Hws_Finish.xsd	Corresponds to the message that an action generates immediately prior to completion.
Hws_ActivateResponse.xsd	Corresponds to the task response that the actor generates in response to a task request. This schema exposes all task-specific properties that the actor needs to communicate to the process. For example, a task may allow a user to accept, reject, or annotate that task.

Specifying action parameters

If you need to specify additional action parameters within the action orchestration, in addition to adding any sections in the *ActionSection* section of the schema, you must add those parameters as either:

- **Promoted fields**. Use these if you use the field to link correlated items or leverage by other actions.

- **Distinguished fields**. Use these if you use the field only inside the current action you develop.

The target namespace of the schema should also be modified to reflect the organization that will distribute the action. For more information, see Promoting Properties.

After you modify the HWS schemas, you can do one of the following:

- Create a separate assembly for the schemas and add a reference to this schema assembly in the action project.

- Copy these schemas to the action project and add a reference to the .xsd files into the project.

The Microsoft® Visual Studio® .NET project includes the necessary schema files and .dll references. You can copy those into a separate DLL if you wish.

Defining Business Processes with ODBA

You use the Microsoft BizTalk Server 2006 Orchestration Designer for Business Analysts (ODBA) to create business process orchestration diagrams that show the general flow of your business process.

With ODBA, you can:

- Create the structure of your business process orchestration diagram.

- Validate the structure of your business process orchestration diagram.

- Export the orchestration diagram to an .odx file that a solutions developer can import into the BizTalk Server 2006 Orchestration Designer.

- Import an orchestration implemented by the developer as a business process orchestration diagram into ODBA.

- Define a Business Activity Monitoring (BAM) activity by adding data of interest and milestones to a business process orchestration diagram.

- Navigate to a BAM Web page and access live BAM data from within the ODBA business process orchestration diagram.

In This Section

- What Is a Business Process Orchestration Diagram?

- How to Install and Start ODBA

- Creating a Business Process Orchestration Diagram

- How to Add Non-ODBA Shapes and Connectors

- How to Validate a Business Process Orchestration Diagram

- How to Export a Business Process Orchestration Diagram to BizTalk Server

- How to Use ODBA to View a BizTalk Orchestration

- Creating a BAM Activity Definition with ODBA

What Is a Business Process Orchestration Diagram?

A business process orchestration diagram is a graphical representation of a business process you create using shapes and connectors from the ODBA template in Visio. You use the ODBA shapes and connectors to diagram actions, process order, decisions, and so on.

The following figure shows a business process orchestration diagram.
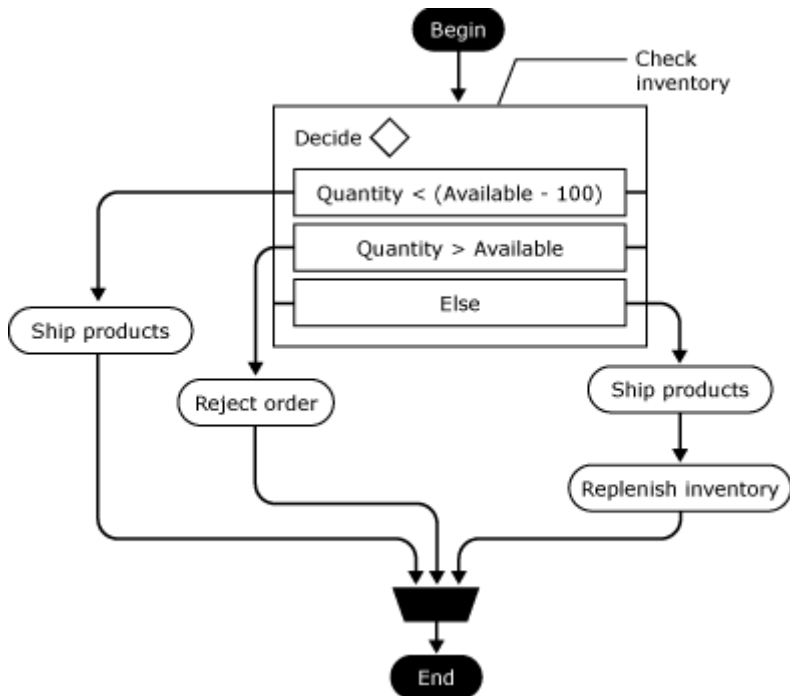
When you open Microsoft Visio with ODBA, the business process orchestration diagram stencil, the design pane, and the Visio menu and toolbars are visible. The ODBA stencil contains all of the business process orchestration diagram shapes. Use the design pane as the area on which to create your business process orchestration diagram. For information about installing and starting ODBA, see How to Install and Start ODBA.

In addition to using ODBA shapes in your business process orchestration diagram, you can use shapes from other Visio stencils to document your process better. For example, to add annotations or indicate processes outside of BizTalk. You use the ODBA connector tool to connect ODBA shapes to each other, and non-ODBA connectors to connect non-ODBA shapes to each other and to connect non-ODBA shapes to ODBA shapes. For information about adding and connecting non-ODBA shapes in a business process orchestration diagram, see How to Add Non-ODBA Shapes and Connectors. For information about connecting ODBA shapes to each other, see How to Add a Connector (ODBA).

When you export a business process orchestration diagram that contains non-ODBA shapes and connectors, the non-ODBA shapes and connectors are not included in the exported file. For information about exporting a business process orchestration diagram as an .odx file, see How to Export a Business Process Orchestration Diagram to BizTalk Server. For information about exporting a business process orchestration diagram as an .xml file, see How to Export a BAM Activity Definition from ODBA.

You can use business process orchestration diagrams in three ways:

Scenario 1

1.   You create a business process orchestration diagram and export it to BizTalk 2006.

2.   A solutions developer imports the business process orchestration diagram as an .odx file into BizTalk Server Orchestration Designer, and uses it as the basis for creating a BizTalk Server orchestration.

Scenario 2

1.   A solutions developer creates a BizTalk Server orchestration in Orchestration Designer.

2.   You import the BizTalk orchestration into ODBA to review it.

Scenario 3

1.   You create a business process orchestration diagram and use the Business Activity Monitoring (BAM) features in ODBA to define data of interest and milestones.

2.   You export the business process orchestration diagram from ODBA as a BAM activity definition (.XML) file.

3.   You import the BAM activity definition file into Excel as a BAM activity. You use the activity to create BAM views for business end users.

How to Install and Start ODBA

Orchestration Designer for Business Analysts (ODBA) for BizTalk Server 2006 is available as a separate Web download. If you have a previous version of ODBA installed, the ODBA setup process automatically removes all components of the older version before it installs the latest version.

In the latest version of ODBA, you can open business process orchestration diagrams created with an earlier version of ODBA.

ODBA Installation Considerations

Note the following considerations for installing ODBA:

- You do not have to install BizTalk Server 2006, or have access to a computer running BizTalk Server 2006, to use ODBA.

- ODBA does not distinguish diagrams by the version of ODBA you used to create the files. To prevent overwriting diagram files, make sure each file has a unique name or path.

- You cannot open a business process orchestration diagram created with ODBA for BizTalk Server 2006 on a 32-bit machine directly in ODBA for BizTalk Server

2006 on a 64-bit machine. You must open the diagram in ODBA 2006 on a 32-bit machine, save the diagram, and then open it in ODBA 2006 on a 64-bit machine.

- If you have existing business process orchestration diagram (.vsd) files, you must use the same language of Microsoft Office Visio 2003 and ODBA to open them that you used to create the original files. For example, if you create a .vsd file in the German version of ODBA, you must use the German versions of Visio and ODBA to open it.

Step 1: Install prerequisite software

Before you can install ODBA, you must have the following software installed on your computer:

- Microsoft Windows 2000, Windows XP with Service Pack 1 (SP1), or Windows Server 2003 operating system. To install the latest service packs, see http://go.microsoft.com/fwlink/?LinkId=25070.

- Microsoft .NET Framework 2.0. For information, see http://go.microsoft.com/fwlink/?LinkId=25072.

- Microsoft Office Visio 2003 with Service Pack 2 (SP2). For information, see http://go.microsoft.com/fwlink/?LinkId=25074.

Step 2: Download and install ODBA

After you install the required software on your computer, download ODBA from http://go.microsoft.com/fwlink/?LinkId=47543.

Step 3: Start ODBA

You can start ODBA from the Start menu, or you can start Visio 2003 and then open a new ODBA drawing.

To open ODBA from the Start menu

1.   Click **Start**, point to **Programs**, point to **Microsoft BizTalk Server 2006**, and then click **Orchestration Designer for Business Analysts**.

To open Visio and select an ODBA drawing type

1.   Click **Start**, point to **Programs**, point to **Microsoft Office**, and then click **Microsoft Office Visio 2003**.

2.   In Visio, on the **File** menu, point to **New**, point to **BizTalk**, and then click **Orchestration Designer for Business Analysts**.

To enable the ODBA template in Visio 2003

1.   In Visio, on the **Tools** menu, click **Options**.

2.   In the **Options** dialog box, on the **Security** tab, select the **Enable COM Addins** check box, select the **Enable Automation events** check box, and then click **OK**.

3.   Close Visio, and then reopen it.

Creating a Business Process Orchestration Diagram

You use Orchestration Designer for Business Analysts (ODBA) in Microsoft Office Visio 2003 to create business process orchestration diagrams. ODBA includes a set of shapes and connectors that you can use to create business process orchestration diagrams. Shapes represent the different tasks and activities in your business process orchestration diagram. Connectors represent the work flow from one activity shape to the next.

For information about business process orchestration diagrams, see What Is a Business Process Orchestration Diagram?

In This Section

- How to Add a Begin shape (ODBA)

- How to Add an Action Shape (ODBA)

- How to Add a Fork Shape (ODBA)

- How to Add a Join Shape (ODBA)

- How to Add a Decide Shape (ODBA)

- How to Add a Group Shape (ODBA)

- How to Add a Loop Shape (ODBA)

- How to Add an End Shape (ODBA)

- How to Add a Connector (ODBA)

- How to Fix ODBA Shape Layout

- How to Save a Business Process Orchestration Diagram

How to Add a Begin shape (ODBA)

The Begin shape represents the beginning of a process flow.

Use rules

Note the following rules when you use a Begin shape:

- Every business process orchestration diagram must have a Begin shape. A business process orchestration diagram may have only one Begin shape.

- You can not delete the Begin shape or add more Begin shapes to the page. The Begin shape is available automatically on the page when you open ODBA or create a new ODBA diagram in Visio.

- In a business process orchestration diagram imported into BizTalk Server, a Begin shape appears as a Begin shape.

Connection rules

The Begin shape can have only one outgoing ODBA connection, and no incoming connection.

To add a Begin shape

1. Open ODBA. For instructions, see How to Install and Start ODBA.

    When you open ODBA, the Begin shape is at the top of the design pane.

    The following figure shows the Begin shape.
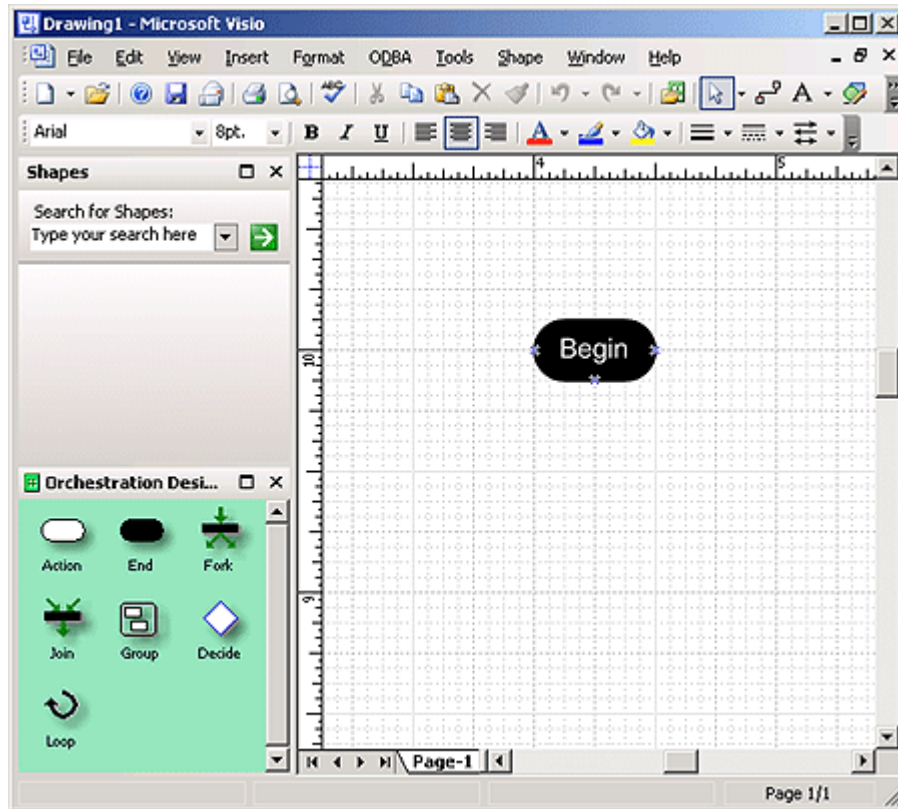
How to Add an Action Shape (ODBA)

The Action shape represents a single business activity or task that happens within the business process orchestration diagram.

Use rules

Note the following rules when you use an Action shape:

- You can have as many Action shapes as you want in the business process orchestration diagram.

- In a business process orchestration diagram imported into BizTalk Server, Action shapes appear as Group shapes because the action shape may represent a business activity or task that has multiple implementation steps.

Connection rules

This shape must have one incoming ODBA connection and one outgoing ODBA connection.

To add an Action shape

1.   From the stencil, drag an Action shape to the design pane and drop it on the
        design surface below the Begin shape.

        The following figure shows a diagram with an Action shape.
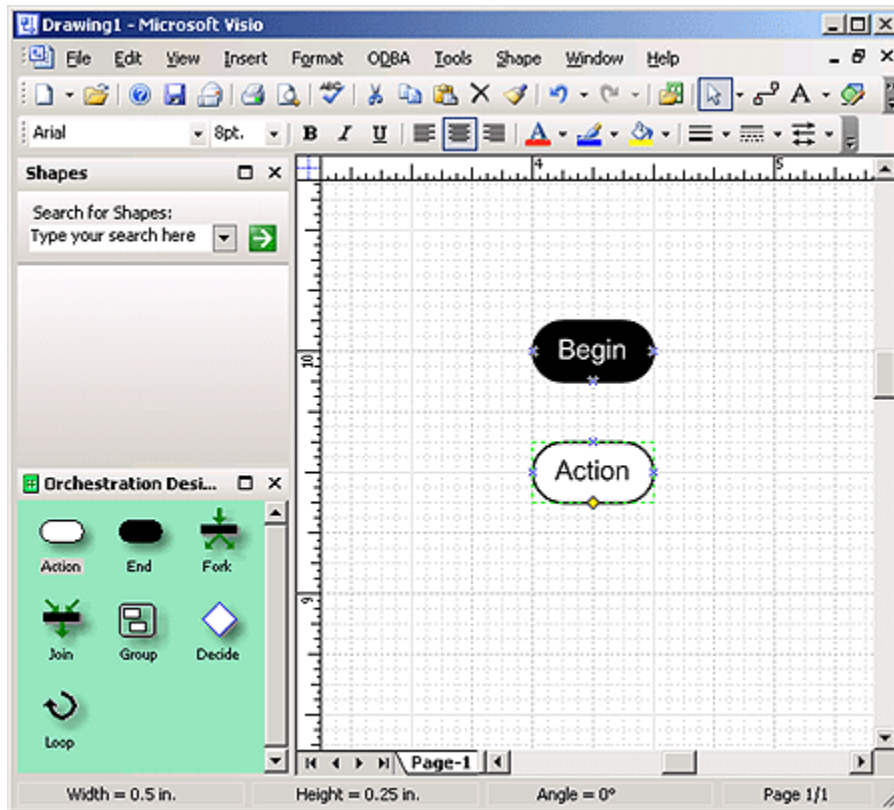
        To give the Action shape a new name, double-click the Action shape, and then
        type the name.

2.   To connect the Begin shape to the Action shape, in the design pane, click the
        Begin shape and drag the yellow diamond at the lower edge of the Begin shape
        to the upper edge of the **Action** shape.

        The following figure shows the Begin shape connected to the Action shape.
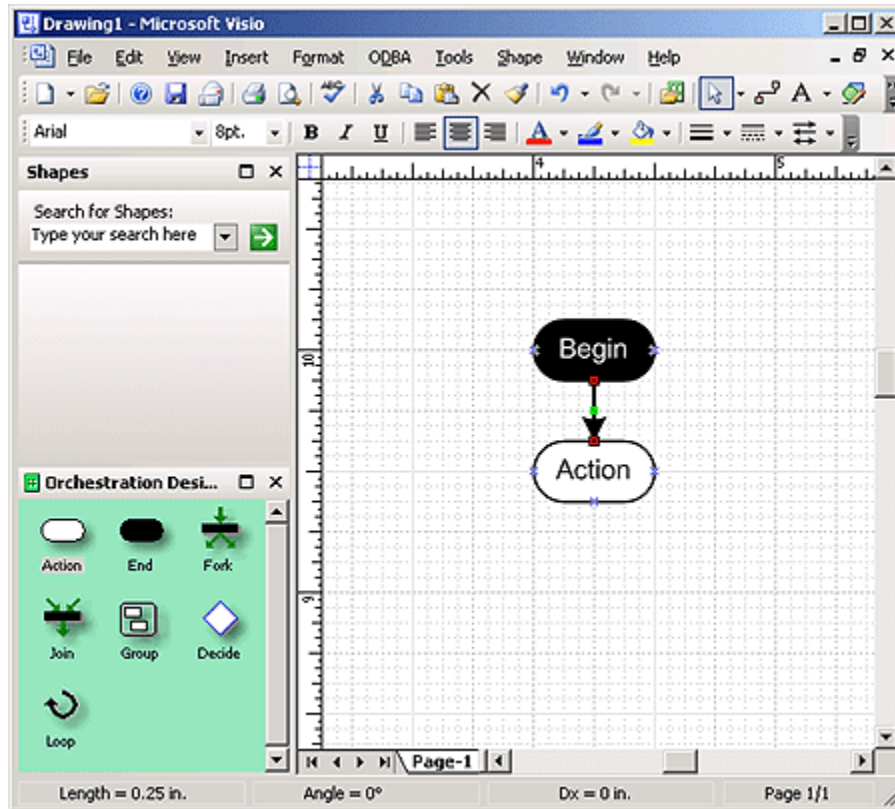
How to Add a Fork Shape (ODBA)

The Fork shape allows parallel flows within a business process orchestration diagram.

Use rules

Note the following rules when you use a Fork shape:

- Parallel flows illustrate different tasks that happen at the same time.

- You can create multiple flows from the lower edge of the Fork shape.

- Flows are expected to execute in parallel until they reach a Join shape or an End shape.

- Drag connectors from the center bottom of the Fork shape. ODBA automatically spaces multiple connectors at the bottom of the Fork shape. If you attach connectors to the corners or sides of the Fork shape, the connectors cause validation errors.

- In a business process orchestration diagram imported into BizTalk Server, a Fork shape, Join shape pair appears as a Parallel shape.

Connection rules

The Fork shape must have one incoming ODBA connection at the upper edge and multiple outgoing ODBA connections at the lower edge.

To add a Fork shape

1. From the stencil, drag a Fork shape to the design pane and drop it on the design surface below the Begin shape.

2. In the toolbar, select the ODBA connection tool.

   The following figure shows the ODBA connection tool.

3. On the design surface, point to the bottom center of the Begin shape.

   The connector at the bottom center of the Begin shape changes to a red square.

   The following figure shows the Begin shape with an active connector.
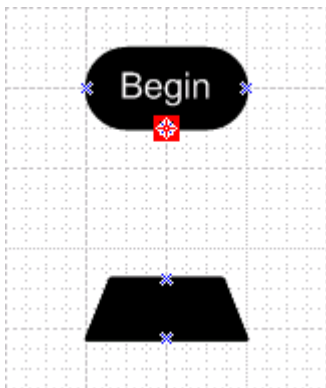
4. Click and drag the connector at the bottom center of the Begin shape to the connector at the top center of the Fork shape.

   The connector at the top center of the Fork shape changes to a red square.

   The following figure shows the Begin shape connected to the Fork shape.
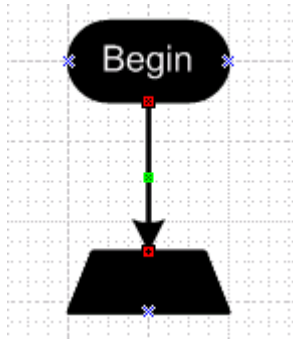
5.   From the stencil, drag the shapes to which you want to join the Fork shape to the design pane and drop them below the Fork shape.

        The following figure shows the Fork shape with the three action shapes.
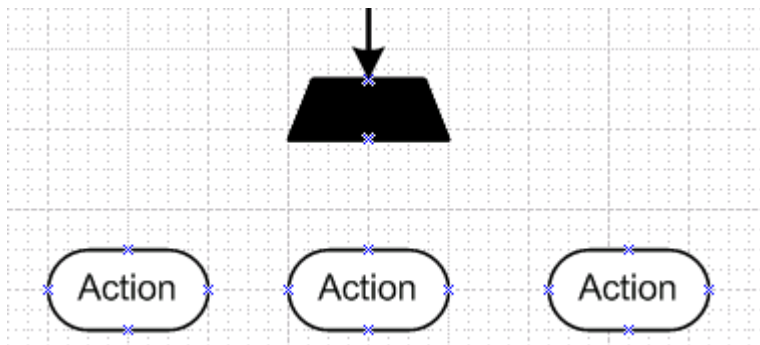
6.   In the toolbar, select the ODBA connection tool.

7.   Click and drag the connector at the bottom center of the Fork shape to the connector at the top center of the center Action shape.

        The following figure shows the Fork shape connected to the center Action shape.
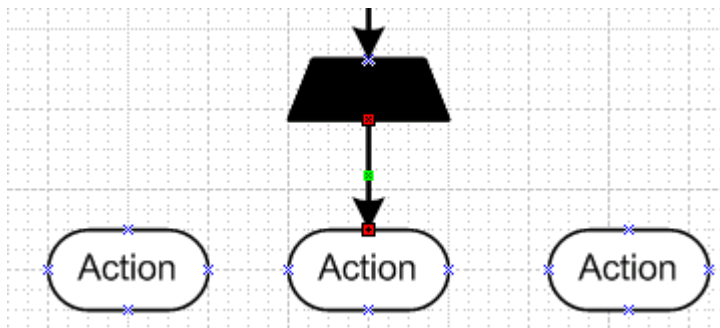
8.   To connect the Fork shape to the left Action shape, point to the connector at the center bottom of the Fork shape (this is the same connector you used to connect to the center Action shape), and then click and drag to the connector to the top center of the left Action shape.

        ODBA automatically adjusts the connections at the bottom of the Fork shape.

The following figure shows the Fork shape connected to the left and center Action shapes.
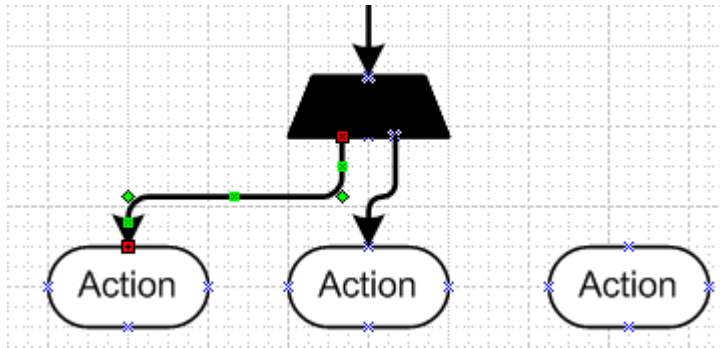
9.   To connect the Fork shape to the right Action shape, point to the top of the connection to the center Action shape at the bottom of the Fork shape, and then click and drag to the connector at the top center of the right Action shape.

ODBA automatically adjusts the connections at the bottom of the Fork shape.

The following figure shows the Fork shape connected to all three Action shapes.
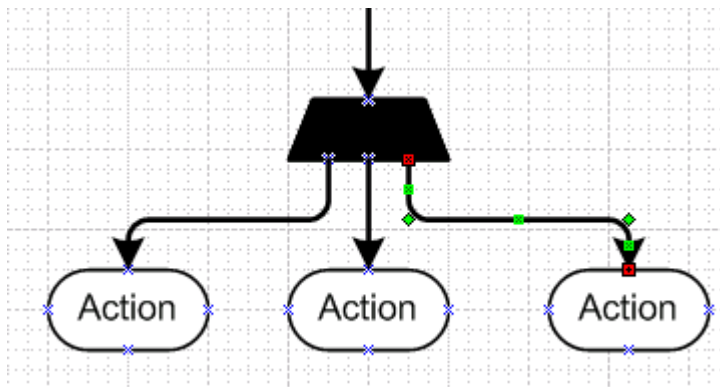
How to Add a Join Shape (ODBA)

The Join shape represents the end of multiple flows that start from a Fork or a Decide shape.

Use rules

Note the following rules when you use a Join shape:

- All the flows that reach the Join shape must start from the same Decide or Fork shape. In other words, flows from different Fork or Decide shapes can't end at the same Join shape.

- The Join shape should be the place where different branches of a Decide or a Fork shape come back together.

- All flows from one Decide or Fork shape must ultimately connect to one Join shape.

- A Join shape connects to the flows from one Decide or Fork shape.

- If you have a Join shape that accepts flows from more than one Decide or Fork shape, the Orchestration Checker lists this as a logic error.

- If you have many flows from a Decide or Fork shape that you want to organize by connecting them to more than one Join shape, use a Decide or Fork shape for the flows for each of the Join shapes.

- Drag connectors to the center top of the Join shape. ODBA automatically spaces multiple connectors at the top of the Join shape. If you attach connectors to the corners or sides of the Join shape, the connectors cause validation errors.

- In a business process orchestration diagram imported into BizTalk Server, a Fork shape—Join shape pair appears as a Parallel shape.

Connection rules

The Join shape accepts multiple incoming connections at the upper edge and a single outgoing connection at the lower edge.

To add a Join shape

1.   From the stencil, drag a Join shape to the design pane and drop it on the design surface below the shapes you want to join.

2.   The following figure shows three Action shapes and the Join shape.
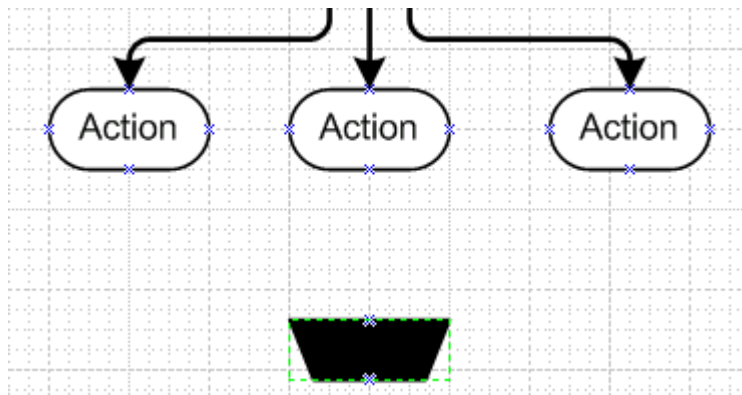
3.

4.   To connect the Action shapes to the Join shapes, from the center bottom of each Action shape, drag a connector to the center top of the Join shape.

ODBA automatically spaces multiple connectors at the top of the Join shape.

The following figure shows the Join shape connected to the Action shapes.
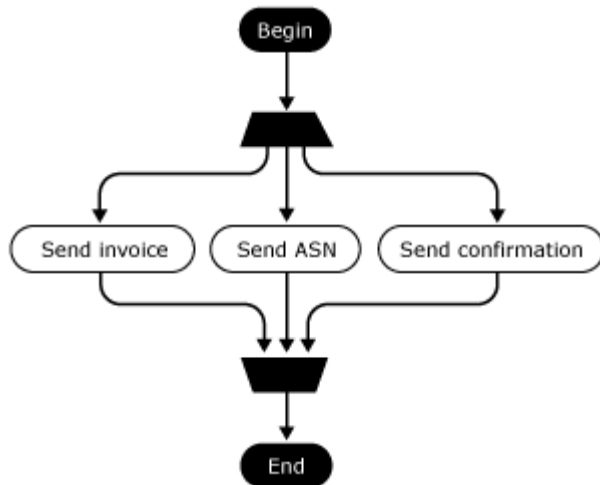
How to Add a Decide Shape (ODBA)

You use the Decide shape to make decisions based on various conditions and perform activities based on the conditions.
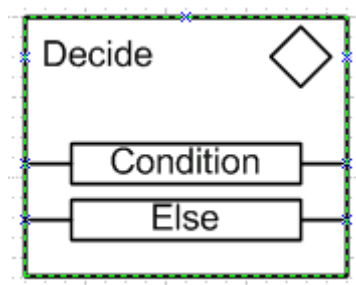
Use rules

Note the following rules when you use a Decide shape:

- The Decide shape must have at least one condition in addition to the Else condition. Conditions are represented by condition shapes inside the Decide shape.

- Double-click the Decide shape to add condition shapes, edit condition shape names, or change the order of the condition shapes.

- Condition shape names can have a maximum of 511 characters. If you add a condition with a name longer than 511 characters, and you export the business

process orchestration diagram to BizTalk Server, the solution developer will not be able to open the exported file in Orchestration Designer.

- Each condition shape, including the Else shape, in a Decide shape must have one outgoing connection that starts on either the left or right side connector of the condition shape.

- Before you can connect a condition shape, make sure you select the condition shape, not the Decide shape. When you select a shape, the control handles for that shape appear. The control handles for condition shapes are on the edge of the Decide shape.

- All conditions in a Decide shape must connect to the same Join shape.

- If you have a complex process with multiple conditions in a Decide shape, and you want to connect some of the conditions to one Join shape and other conditions to connect to a different Join shape, create one Decide shape with the conditions for the first Join shape, and a separate Decide shape for the conditions that connect to the other Join shape.

- You can connect a condition shape in one Decide shape to another Decide shape. You cannot connect a condition shape in one Decide shape to a condition shape in another Decide shape.

- In a business process orchestration diagram imported into BizTalk Server, a Decide shape appears as a Decide shape.

Connection rules

The Decide shape must have one incoming connection at the upper edge. Each condition or Else shape in a Decide shape must have exactly one outgoing ODBA connection. The outgoing connection can start at either side of the condition shape. All the flows from the decision's conditions must merge at the same Join shape.

To add a Decide shape

1.   From the stencil, drag a Decide shape to the design pane and drop it on the design surface below the Begin shape.

2.   To add a condition, double-click the Decide shape.

3.   In the Decide Shape Properties dialog box, click Add to add a new condition, or in the Conditions list, select a condition, and then click Edit.

4.   In the Condition Properties dialog box, in the Condition name box, type a name for the condition.

5.   Click OK to close the Condition Properties dialog box.

6.   When you are finished adding conditions, click OK to close the Decide Shape Properties dialog box.

7.   In the design pane, connect each condition shape, including the Else shape to another ODBA shape.

The following figure shows the Decide shape condition shapes connected to Action shapes.
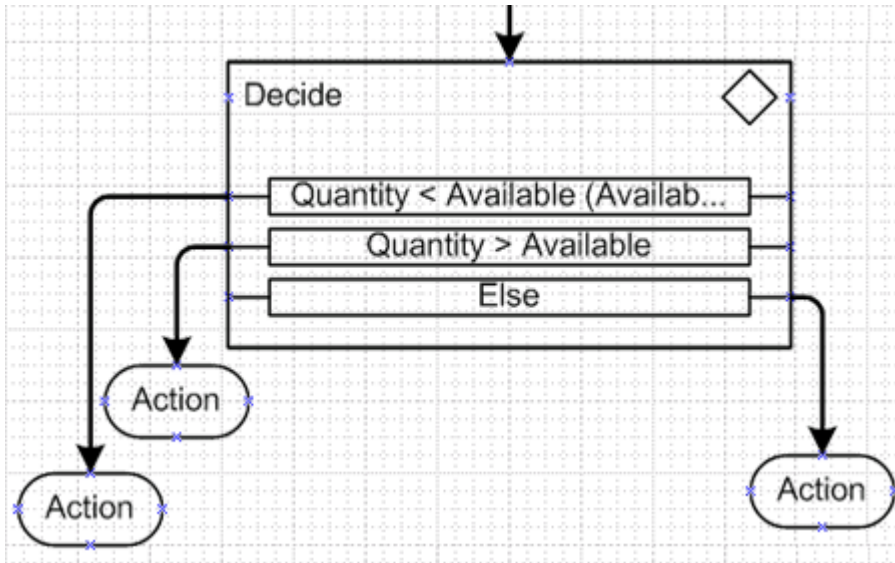

8.   From the stencil, drag a Join shape to the design pane and drop it on the design surface below the Action shapes to which you connected the condition shapes.

9.   Connect the Action shapes to the Join shape.

The following figure shows all conditions in the Decide shape connected to the same Join shape.
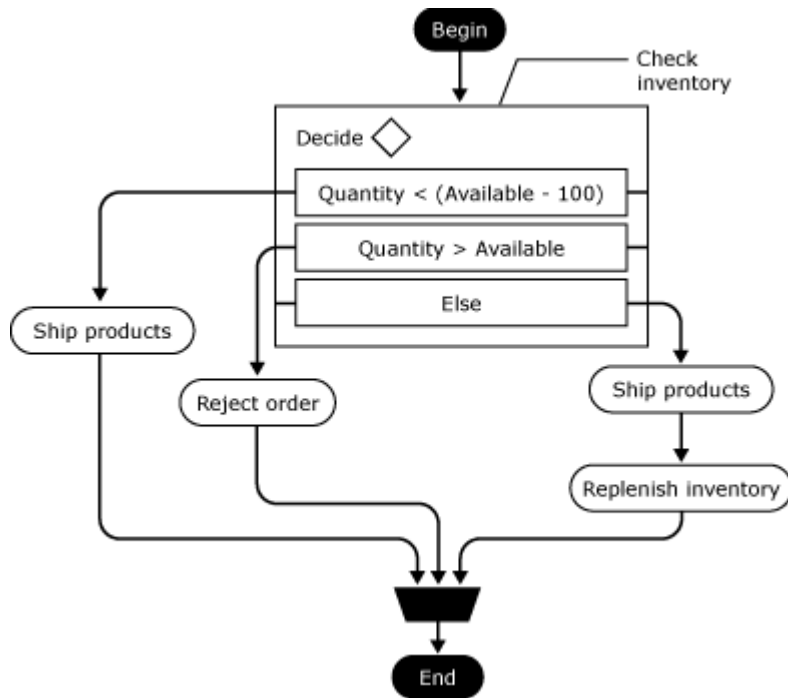
How to Add a Group Shape (ODBA)

The Group shape represents a group of related activities in a business process orchestration diagram that happen in sequence.
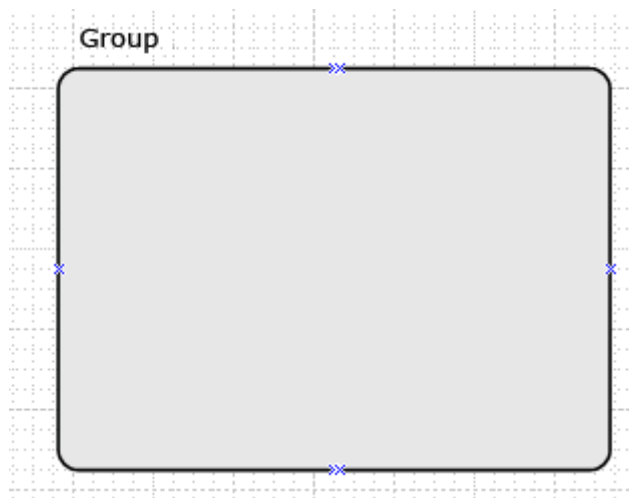
Use rules

Note the following rules when you use a Group shape:

• When you create a Group shape, you can place one or more other shapes within it.

• Shapes must be inside of the Group shape to be part of the group.

- Resizing a Group shape will move the shapes it contains. If resizing the Group shape moves the interior shapes outside of the group shapes, you must manually move the shapes back inside the Group shape to maintain a valid business process orchestration diagram.

- Deleting a Group shape deletes all shapes it contains.

- Group shapes can contain a caption. For example, the "Complete Order" caption in the figure.

- You can drag a caption anywhere in the design pane.

Connection rules

Note the following rules when you connect a Group shape:

- Each Group shape must have at least one incoming and one outgoing ODBA connection attached to it.

- The Group shape must have one connection flowing from the top of the group to the first activity or action within it.

- Each shape inside the Group shape must have appropriate connections. For information about connecting ODBA shape, see the topic for the shape (listed in See Also section of this topic).

- You cannot connect shapes contained within a group directly to shapes that are not contained within the same group.

- At the bottom of the group, the final activity or action in the group must be connected to the inner bottom connector of the group.

- In a business process orchestration diagram imported into BizTalk Server, a Group shape appears as a Scope shape.

To add a Group shape

1. From the stencil, drag a Group shape to the design pane and drop it on the design surface below the Begin shape.

2. To give the Group shape a new name, double-click the Group shape, and then type the name.

3. Add ODBA shapes to the group by dropping them inside of the Group shape.

4. Add the following connections:

   - Add a connector that connects the preceding shape to the top of the Group shape.

- Add a connector that connects the top of the Group shape to the top of the first shape inside of the Group shape.

- Add a connector that connects the bottom of each shape inside of the Group shape to the top of the next shape inside of the Group shape.

- Add a connector that connects the bottom of the last shape inside of the Group shape to the bottom of the Group shape.

- Add a connector that connects the bottom of the group shape to the top of the next shape in the diagram.

The following figure shows a diagram with a Group shape.
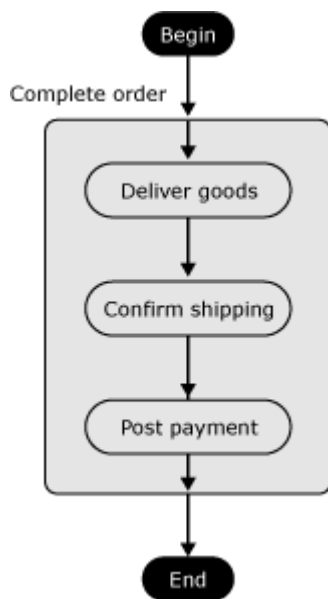
How to Add a Loop Shape (ODBA)

The Loop shape represents a set of repeating activities in a business process orchestration diagram.
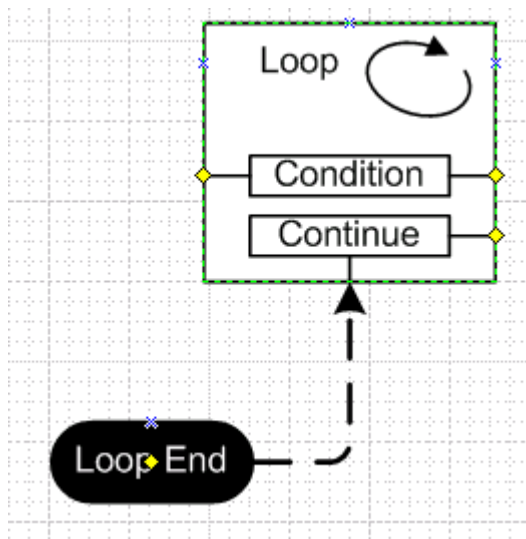
Use rules

Note the following rules when you use a Loop shape:

- Use a Loop shape to repeatedly perform a sequence of operations while a condition is satisfied, for example, processing multiple items in a purchase order.

- When the condition is no longer satisfied, for example, when there are no more items remaining in the purchase order, the Loop shape exits from the Continue sub shape.

- You can only add one condition for the loop.

- The outgoing flow from the condition must ultimately stop at the Loop End shape.

- To make room for a series of processes leading to the Loop End shape, click inside the Loop End shape. A yellow diamond appears inside the shape. Point to the diamond and drag the Loop End shape where you need it on the design pane.

- In a business process orchestration diagram imported into BizTalk Server, a Loop shape appears as a Loop shape.

Connection rules

Note the following rules when you connect a Loop shape:

- The Loop shape must have one incoming connection at the upper edge.

- The Condition and Continue shapes must each have exactly one outgoing ODBA connection.

To add a Loop shape

1.   From the stencil, drag a Loop shape to the design pane and drop it on the design surface below the Begin shape.

2.   From the ODBA stencil, drag a shape (or shapes) to the design surface that represents the action the Loop will perform until the Loop condition is met.

3.   Connect the Loop condition to this shape, and connect the shape to the **Loop End** shape.

       The following figure shows a diagram with a Loop shape.
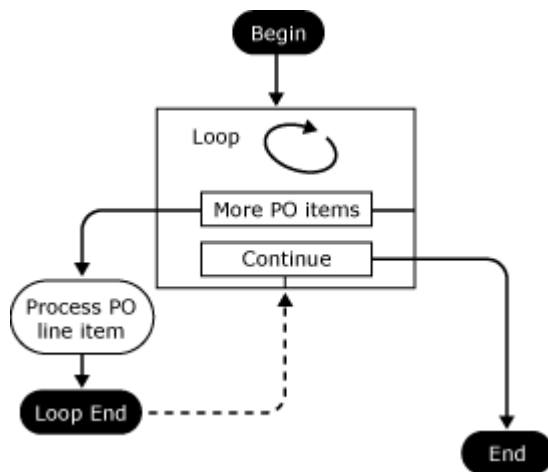
How to Add an End Shape (ODBA)

The End shape represents the end of a business process orchestration diagram flow.

Use rules

Note the following rules when you use an End shape:

•   You can add multiple End shapes to a business process orchestration diagram.

•   To validate a business process orchestration diagram and export it to BizTalk Server (generate an .odx file) the diagram must have only one End shape.

       For information about validating a business process orchestration diagram, see How to Validate a Business Process Orchestration Diagram.

       For information about exporting a business process orchestration diagram to

BizTalk Server, see <u>How to Export a Business Process Orchestration Diagram to BizTalk Server</u>.

- In a business process orchestration diagram imported into BizTalk Server, an End shape appears as an End shape.

Connection rules

An End shape can have only one incoming ODBA connection and no outgoing ODBA connections.

To add an End shape

1. From the stencil, drag an End shape to the design pane and drop it on the design surface below the Begin shape.

   The following figure shows a diagram with an End shape.
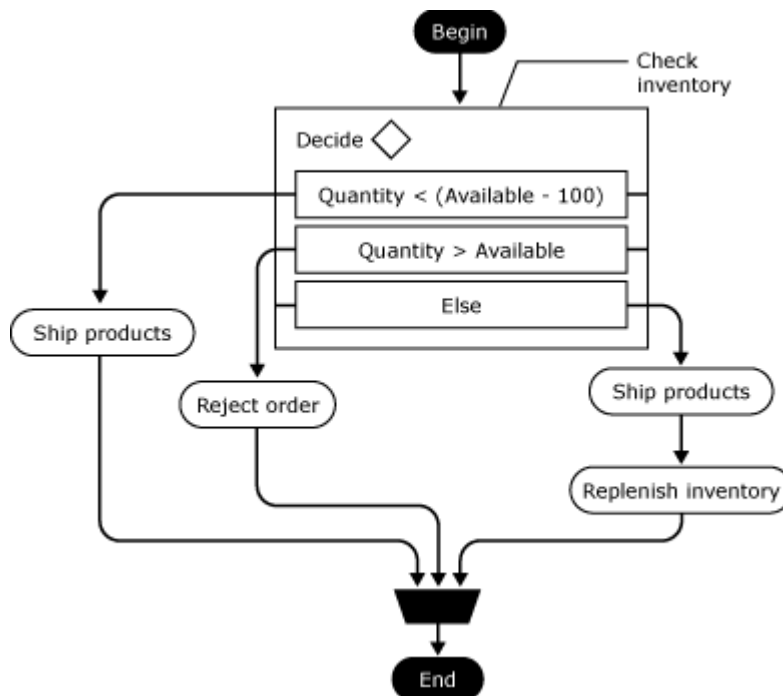
How to Add a Connector (ODBA)

Connectors are lines that represent the flow of control from one shape to the next in the process. In order for a connector to be valid, it must originate from a valid starting point in the source shape and extend to a valid end point in the destination shape. Different shapes have different rules that specify how you can connect them.

Connection Tools

In Orchestration Designer for Business Analysts (ODBA), there are two tools you use to connect shapes:

- **Pointer tool**. Use the pointer tool to connect two shapes by dragging the connection point of one shape to the connection point of another shape.

  The following figure shows the pointer tool.

- **Connector tool**. Use the connector tool to draw a line between the connection points of two shapes.

  The following figure shows the connector tool.

The pointer tool and the connector tool appear in the Visio toolbar when you install ODBA. Connections you create with these tools only attach to valid connection points.

The method you use to connect shapes in your business process orchestration diagram depends on which of the tools is active in the toolbar.

Connection Points

When you add ODBA shapes to the design pane, the shapes have an x indicating acceptable connection points. If you don't see the connection points, on the **View** menu, click **Connections Points**.

The following figure shows the Begin shape connection points.
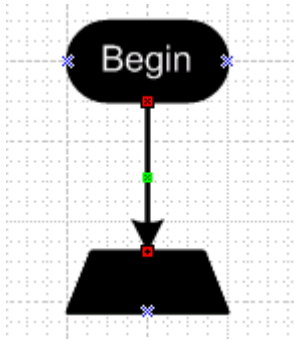

The following figure shows the Fork shape connection points.

You use ODBA connectors to connect ODBA shapes to each other. You use connection points on the top and bottom of ODBA shapes to connect the shapes to other ODBA shapes.

The following figure shows the Begin shape connected to the Fork shape.

If you want to connect a non-ODBA shape to an ODBA shape, you must use a Visio (non-ODBA) connector. You use connection points on the left and right sides of ODBA shapes to connect the shapes to non-ODBA shapes. Non-ODBA shapes and connectors are not included when you export a business process orchestration diagram file. For information about using non-ODBA shapes and connectors in your diagram, see How to Add Non-ODBA Shapes and Connectors.

To use the pointer tool to connect shapes

1.   In the ODBA toolbar, click the pointer tool.

2.   In the design pane, click the shape from which you want the connection to start.

        A yellow diamond, the control handle, appears at the lower edge of the shape.

        The following figure shows the Begin shape control handle.
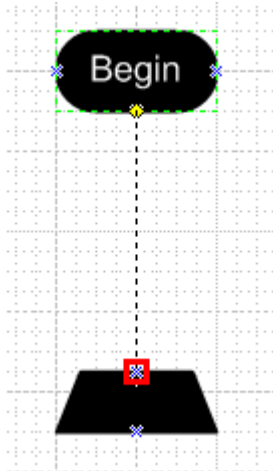

3.   Drag the control handle to the destination shape.
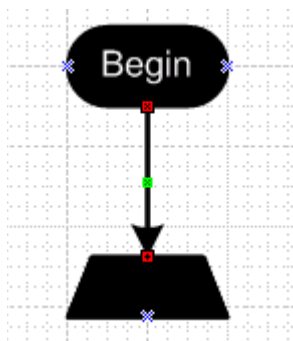
        The mouse pointer changes to a crosshair.

        When you move the crosshair over an **x** on the destination shape that accepts the connection, a red square appears.

        The following figure shows the Fork shape with the red square connection point.

4.   To complete the connection, release the mouse over the x with the red square.

5.   The following figure shows the Begin shape connected to the Fork shape.

6.

To use the Connector tool to connect shapes

1.   In the ODBA toolbar, click the connector tool.

2.   In the design pane, move the mouse pointer over an x on the shape from which you want the connection to start.

A red square appears.

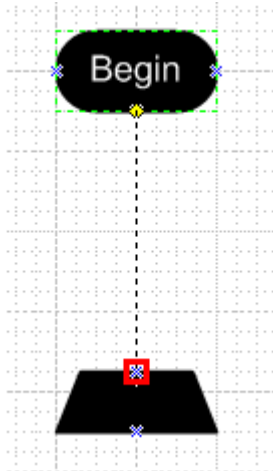The following figure shows the Begin shape with an active connector.

3.   Click the red square and drag to an **x** on the destination shape.

The mouse pointer changes to a crosshair.

When you move the crosshair over an **x** on the destination shape that accepts the connection, a red square appears.

The following figure shows the Fork shape with the red square connection point.

4.   To complete the connection, release the mouse over the x with the red square.

The following figure shows the Begin shape connected to the Fork shape.
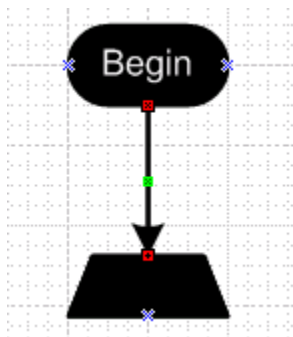
How to Fix ODBA Shape Layout

You use the Layout Workflow tool in ODBA to help you with the workflow logic and layout of your business process orchestration diagram.

To fix process diagram layout

1.   On the ODBA menu, click Layout Workflow.

ODBA checks the workflow logic of the business process orchestration diagram and fixes layout issues.

How to Save a Business Process Orchestration Diagram

You do not have to save a business process orchestration diagram to export it as an .odx file or as a BAM definition file. However, we recommend that you save your business process orchestration diagram in Visio, especially if you are still working on it.

To save your business process orchestration diagram as a Visio file

1.   In Visio, click **File**, click **Save As**, browse to a save location, type a filename, and then click **Save**.

How to Add Non-ODBA Shapes and Connectors

In addition to using ODBA shapes in your business process orchestration diagram, you can use shapes from other Visio stencils to document your process better. For example, to add annotations or indicate processes outside of BizTalk. You use the ODBA connector tool to connect ODBA shapes to each other, and non-ODBA connectors to connect non-ODBA shapes to each other and to connect non-ODBA shapes to ODBA shapes.

When you export a business process orchestration diagram that contains non-ODBA shapes and connectors, the non-ODBA shapes and connectors are not included in the exported file.

To access non-ODBA shapes in Visio

1.   On the **File** menu, point to **Shapes**, point to the type of shapes you want to use, and then click the shape group.

To add non-ODBA shapes to a business process orchestration diagram

1.   From a non-ODBA stencil, drag and drop shapes into the diagram.

To connect non-ODBA shapes to ODBA shapes

1.   From a non-ODBA stencil, drag a connector to a connection point on the non-ODBA shape, and when the connector attaches itself to the shape, drop the connector.

2.   Drag the other end of the connector to a non-ODBA connection point on an ODBA shape.

How to Validate a Business Process Orchestration Diagram

You use the Orchestration Checker tool to check for logic errors in your business process orchestration diagram. You must fix all logic errors in your diagram before you export it. When you export your diagram to BizTalk Server, ODBA checks your diagram for logic errors.

If you export the diagram to BizTalk Server, your diagram must pass all of the logic tests to ensure that a solutions developer can successfully import it and use it in BizTalk Server.

You can run the Orchestration Checker tool directly or, Orchestration Checker launches automatically when you export a business process orchestration.

When you run Orchestration Checker, it goes through your business process orchestration diagram and locates any logic errors, for example, a path of actions with no End shape. The Orchestration Checker dialog box lists any errors and provides a description of each error and a suggestion for fixing each error.

When you run the Orchestration Checker tool, it checks for the following:

1.   The diagram contains a Begin shape

2.   All ODBA shape end points that must be connected are connected, and none are connected to more than one ODBA connector.

3.   All ODBA connectors point in the correct direction. Arrow heads flow into the top of shapes, not into the bottom.

4.   Both ends of an attached ODBA connector are either in the same group, or not in any group.

5.   The branch for each Loop shape ends in the loop end shape.

6.   Each Join shape has a matching Fork or Decide shape. Each Join shape only joins flows that originate from one Fork or Decide shape.

7.   All branches of a Fork or Decide shape end in the same Join shape.

To check a business process orchestration diagram

1.   On the **ODBA** menu, click **Check Orchestration**.

       When the checking logic process completes the validation, the **Orchestration Checker** dialog box lists any logic errors.
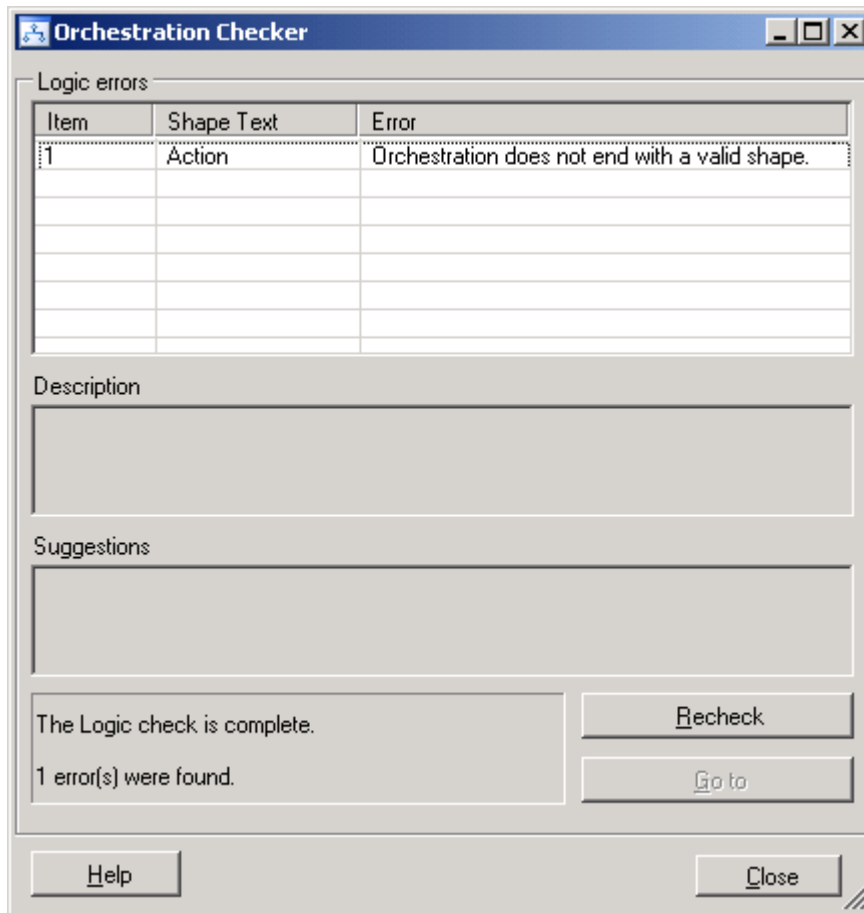
2.   In the **Orchestration Checker** dialog box, click any of the errors to display the description of the error, as well as any suggestions to correct the error.

3.   Click the **Go To** button to highlight the shape in the business process orchestration diagram that caused the error.

4.   If your process has logic errors, correct the errors, and then in the **Orchestration Checker** dialog box, click **Recheck**.

5.   When you have fixed all of the logic errors, click **Close**.

You can now export your business process orchestration diagram to an .odx file for the developer to use

How to Export a Business Process Orchestration Diagram to BizTalk Server

After you have corrected any logic errors in your business process orchestration diagram, you can export the document as an .odx file. A solutions developer can import the .odx file into Orchestration Designer and then use the diagram as a starting point for implementing an orchestration.

Considerations

Note the following considerations when you export a business process orchestration diagram to BizTalk Server:

- When you export a business process orchestration diagram to BizTalk Server, ODBA launches the Orchestration Checker tool to validate the workflow..

- If your business process orchestration diagram contains non-ODBA shapes or connectors, they are not included in the exported file. The workflow checker presents an error message saying that non-ODBA shapes will not be exported to BizTalk.

- ODBA does not support the full round-trip of business process orchestration diagrams (.odx files) from ODBA to Orchestration Designer, and back again. For example:

  1. In ODBA, you create a business process orchestration diagram. You save the diagram as a Visio (.vsd) file and export it to BizTalk Server. The export process generates an .odx file.

  2. In Visual Studio, a solutions developer adds the .odx file to a BizTalk project and updates it. The solutions developer saves the orchestration (.odx) file.

  3. In ODBA, you open the updated .odx file. The updated .odx file has no physical relationship to either the business process orchestration diagram (.vsd file) you saved, or to the original .odx file you exported from ODBA.

  4. In ODBA, you update the business process orchestration diagram and export it to BizTalk Server (.odx file). In Visual Studio, the solutions developer must re-add the .odx file to the BizTalk project. The newly added .odx file has no physical relationship to the original orchestration.

To export an orchestration diagram

1. In the **ODBA** menu, click **Export to BizTalk**.

2. In the **Export to BizTalk** dialog box, from the **Pages to export** list, select the check box of the first page you want to export, and then click **Browse**.

3. In the **Save ODX** dialog box, do the following:

Use this	To do this
Save in	Browse to the folder in which you want to save the ODX file.
File name	Type a file name for the ODX file.
Save as type	From the drop-down list, select **odx file (*.odx)**.

4.   Click **Save**.

5.   Repeat steps 2 and 3 to save any additional pages as ODX files, and then click **OK**.

   The **Orchestration Checker** dialog box opens automatically and checks the validity of each ODX file. You must correct any validation errors before you can complete the export process.

   When the export process is complete, you can send the exported .odx file to the solutions developer.

Import a business process orchestration diagram into BizTalk Server

A solutions developer adds an exported business process orchestration diagram to a BizTalk project. The solutions developer uses the business process orchestration diagram as the basis for building a BizTalk orchestration. The solutions developer must modify the diagram so that as an orchestration it compiles.

To import a business process orchestration diagram into a BizTalk project

1.   In Visual Studio, in Solution Explorer, right-click a BizTalk project, point to **Add**, and then click **Existing Item**.

2.   In the **Add Existing Item** dialog box, select the exported business process orchestration diagram (.odx) file, and then click **Add**.

How to Use ODBA to View a BizTalk Orchestration

You can open BizTalk orchestrations in Orchestration Designer for Business Analysts (ODBA). Opening an orchestration in ODBA allows you to view the orchestration without installing BizTalk Server 2006.

Although you can export business process orchestration diagrams as .odx files, and BizTalk Server orchestrations are .odx files, ODBA does not support the full round-trip of business process orchestration diagrams (.odx files) from ODBA to BizTalk Server, and back again.

To view a BizTalk orchestration in ODBA

1.   Obtain the orchestration (.odx file) from the solutions developer.

2.   On the **ODBA** menu, select **Import from BizTalk**.

3.   In the **Import ODX** dialog box, navigate to the location of the .odx file, select the .odx file, and then click **Open**.

   ODBA imports the .odx file as a business process orchestration diagram (.vsd).