**1.What is Integration?**

Integration is nothing but integrating different applications software like SAP, Oracle, PeopleSoft, Seibel etc. which belongs to different vendors.

Integration is binding of different applications, which may belongs to different vendors together to achieve effective communication and sharing of data.

**2.Starting IS?**

We can run IS 2 ways 1)application 2) service

Application:IntegrationServer/bin/server.bat/sh

Service: windows administrative tools/services/Integration Server we can start

Windows –Integration Server\bin\server.bat –switch –switch -

Unix– IntegrationServer/bin/server.sh –switch –switch –

By default webmethods Server Port numbers

IS: 5555

Broker: 6849

TN: 5555 (TN runs in Integration server, hence it is having same port)

MWS: 8585

BPM:

Reason For Integration Server Doesn’t Up

cause of internal DB Down(Not all casess50-50- chance)

cause of DEFAULT Package disable

there is no space in File System

cause of Corrupted Config.XML

Soap 1.1 vs 1.2

how to resubmit a service in MWS

if u want to resubmit in MWS , That service should be enabled Audit logging

in MWS

Difference Between Top down Approach and Bottom up approach

**Topdown:** first design then code development

**Bottom up:** first code development then Design

Difference between 8X and 9X

No developer in 9X

No lock file in 9x

SFTP package open source in 8X , in 9X SFTP comes as Inbuilt package

For system maintenance we can disable all packages in 8x , in 9 we have a option in Is admin QC mode

Command Centralintroduced: if u have 10 IS u can do (add) group of operations and set of operations

And monitoring also we can do here

Introduced global Variable

**3.How to increase the IS performance.**

* Enabling The Cache Result
* Dropping the unwanted variables,
* To Avoid Save pipeline
* Delete Flow Steps What Ever Disabled

4.**What is clustering ?**

We can run multiple virtual server nodes on one Single Integration Server

**5.If we want to run the server with some other port number, what do we need to do?**

1. Go to IS admin.
2. Click Ports tab under security
3. Go to add port & select (webMethods/http)
4. Go to change primary port select the port & update it

**6.What are the different ports in IS?**

HTTP (5555), HTTPS, FTP, FTPS, FILE POLLING, SMTP

**7.By default, the webMethods Integration Server has an HTTP listener assigned to which port?**

5555

**8.How do I change the JVM used by Integration Server?**

To change to the JDK used by webMethods you will need to edit the IntegrationServer\bin\server.bat or IntegrationServer/bin/server.sh file used to start up Integration Server. Edit the file and change the following line to point to the JDK path SET JAVA\_DIR=C:\opt\j2sdk1.4.2

**9.The Integration Server requires access to the Java classes for each JDBC driver that it will use. Typically, where must such Java classes be placed?**

You should place the Oracle\_14 jar files in Location SoftwareAG\IntegrationServer\lib\jars

Difference Bet weenBaisc Notification and Insert notification?

In branch step I am getting null/empty value/space how do u handle these type of senarion through branch step ?

Can you use any service in catch block before get lost error?

Trigger subscribed the Document from broker , Service Execution Failed, I don’t want to loosetheDocument ?

Transaction logging framework?

Error handling Framework?

What mean by trigger Que Capacity and Refill Level ?

Scenario: I have a database , using pub-sub model convert the data and send to target system using FTP Location ?

**10.How can the webMethods Integration Server logging date format be changed?**

By editing the watt.server.dateStampFmt parameter in the C:\SoftwareAG\IntegrationServer\config\server.cnf file

**11.If the webMethods Integration Server is started with from the server root directory with this command, "bin\server.bat -debug 9 -log none", what does this tell the server to do?**

Start in level 9 debug mode and write all server log information to the screen.

**12.How many kinds of log files in WebMethods? Where do they exist?**

* Server: Operations and errors that occur on Integration Server, such as starting ofIntegration Server subsystems and loading of packages belonging to IS or other webMethods products
* Session: Sessions opened on IS by clients and Developer users
* Error: Stack trace information about all errors that occur in IS, including exceptionsthrown by services
* Guaranteed delivery: Guaranteed delivery transactions
* Service: Services that run in IS

**13.What are extended settings?**

Extended settings are specific setting that are applied to IS

Extended settings are basically done for specifying values to some of the internal keys of Integration server. Like we can specify java compiler at this setting and whenever we compile any java service IS will take this compiler only for compiling the java code.

**14.How do I debug the Developer IDE itself?**

Start the developer up in debug mode, similar to the Integration server: cd pathToWMInstall/Developer/bin developer.bat -debug 10

**15.How to Restore a Session on a Server?**

Developer gets disconnected from the server if the server goes down or if there is a problem in the network. Do not close the developer. If you close the developer you won’t be able to save the changes. Once the server comes up or the network problem is resolved, you will be automatically connected to the server and then you can restore your session.

Refreshing the session is different from restoring a session. Restoring a session allows you to save changes to an element you were working with when the Integration Server shuts down unexpectedly.

Sometimes a server might shut down before you can save your work. Developer preserves any unsaved work as well as lock information, despite the loss of the connection to the server. When the server restarts, you can restore your session and save your changes to the server.

**16.How to open a session on a different IS?**

In package Navigator we have a icon Add or Modify IntegrationServer,

And in designer Windows>Preference>IntegrationServer>Add

we can Add different IntegrationServer we can start new Session.

**17.Can you delete a package from developer if not how can you delete a package in webMethods?**

Yes we can delete (Safe Delete) the packages from the developer –

But this goes to the package recovery area.

To delete it permanently we can use the IS Admin page Package Management (Delete instead of Safe Delete)

**20.How do you export package from one version to another version?**

In packages we can have Manifest.v3file in that file u can change version

In package properties>packageSeetting>packageversion>1.0 u can change here

To install the new version Install Inbound Releases in the Package Management of IS Admin page

In package properties>packageSetting>packageversion>1.1

We can change here

**21.How do you deploy packages?**

Place the packages to be installed in the inbound folder. Install Inbound Releases in the Package Management of IS Admin page

**22.The Package Management interface of the webMethods IS Administrator can be used to create package dependencies. Package dependencies can be used to?**

Ensure that specific webMethods IS packages are loaded before the depending package loads

**23.If I have to move packages from one IS to another, which process would you suggest, is it through wmDeployer or some other processes like publish/subscribe?**

Yes the best way to move (migrate) packages is through WmDeployer.

**24.Should we and how can we archive derived files?**

Yes we can archive the derived files by two ways

* We can do in IS Admin Page Packages>management>Archie when u archive package the package will placed in outbound folder
* In developer file-->export as a ZIP file.

**25.What Is an Element?**

Element is nothing but a item which is existed in package navigator panel , like flowservice,folder,javaservice ……

Except package and server

**26.Can we multi-select elements to lock or unlock in the Navigation Panel?**

Yes but selection cannot contain

1. server
2. folder or package and its content
3. package and any other element

**27.How to Move Flow Steps?**

Arrow buttons on the editor panel to move the selected flow steps. Up down left or right.

**28.How to find dependents of a selected element on the server?**

Right click on the element for which you have to find the dependents in the navigational panel and click on the option find dependents.

Startup & Shutdown Services?

Startup Service: We Can See Startup Service In Package Properties

Startup Service Is A Service Integration Server Automatically Executes When Packages Loads Into Is Memory

When Startup Service Executes: I) When Is Starts/Restart

Ii)When Reload The Package.

Iii)When Deploy To The Target Server

Iv)If The Package Is Disabled , When U Enable The Package

Shutdown Service: When Package Release/Unload Package From Is Memory Shutdown Service Executed

When Shutdown Service Executes:

I) When Is Shutdown/Restart

Ii)When U Disable The Package

Iii)When U Reload The Package

Replicate service: when u archive the package Replicate service will execute

**29.By how many ways can we invoke a service?**

i)usinginvoke step

ii)using transformer

iii)through Http server

iv)inwmpublic/pub/remote/invoke using this service

v)using is/package

vi)using java service

1.using invoke step within webMethods IDE  
2.using Transformers in Map Step  
3.using remoteInvokebuiltin service can invoke flow service  
from other webMethods Environment  
4.using service.doInvoke() in Java services   
5.using webServices,consumingwebService provider   
operations.  
6.usingwebmethods supported protocols like http,https.  
Ex: http://hostname:portnumber/Invoke/namespace  
7.using Publish subscribe mechanism in webmethods  
8.In BPM using CAF application, invoking the flow service   
through a web service call  
9.using Scheduler in webMethods  
10.using File polling in webMethods  
11.using Startup & Shutdown services

**30.How to invoke a service from a browser?**

Use a URL in the form: http://servername:port/invoke/folder.subFolder.subsubFolder/serviceName (the package name is not part of the URL in any way)

**When you create and save the FLOW "my.pack:myFlow" in the "MyPack" package, where will you find the code?**

In the "MyPack\ns\my\pack\myFlow\flow.xml" file

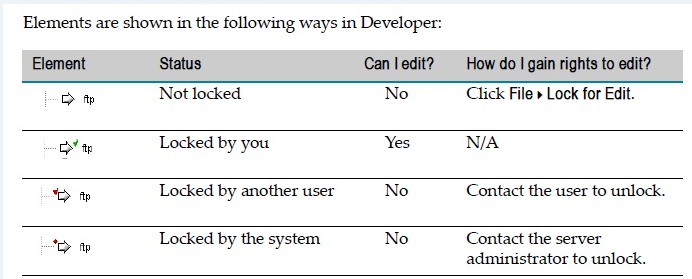
What Is a Lock?

Restrict To Modifying The Element From Other Users, When You Want To Modify The Element That Element Should Locked By You

2 Types Of Locking User Lock And System Lock .

When U Create The New Elements It Is Automatically Locked By The User , Because It Restrict By The User When U Want To Modify The Element It Should Be Locked.

System Lock: for Read-Only Files system lock will Apply It Is Automatically Done By The Server Side Files.



**How to Know Who Has an Element Locked?**

To know who has locked the service or a document or etc right click that particular service or document and select LOCK PROPERTIES from the menu displayed.

IS Admin will also have the lock information?

**How to remove a system lock from an element?**

System locks can be removed by making the server side files of the element as readable. Right Click on the element in developer which is system locked and choose the lock properties.

It will display the server side files for the element. Make the files as readable and click the refresh button in the developer. You will find that the element is no more locked.

**When to Lock an Element?**

When you want make changes to the elements.

**When to Unlock an Element?**

When you are taking a build while deployment you have to unlock the elements.

**What Happens When a Client Runs a Service with ACLs?**

The IS checks the ACL assigned service. If the client is a member of an allowed group and is not a member of a denied group, the server executes the service. If the client is not a member of an allowed group, the IS denies the request to invoke the service and stops executing

**How ACLs Affect locking?**

ACLS are used to give the authorization to the particular user groups. If you give ACL to administrator then the admin group users who are there in that group they can have the access to that particular service. Otherwise we cannot use it. This is called ACL locking.

**How to Create an ACL?**

Create a User after add that user in to particular group

After select ACL and Add group into Allowed list

Read or Write or Execute or List ACLS

**Is Source Control the Same as Locking?**

No. Locking is one component of a full source management solution. Source control is designed to allow the administrator to archive and maintain multiple file revisions. Many source control applications also allow the administrator to tag revisions, run revision history reports, and other functions. Lock functionality allows the user or administrator to control who is working on a particular element at a particular time.

**What Is Flow Diagram View?**

Flow diagram view, like flow tree view, is a view of a flow service that Developer displays in the editor.

Audit Logging:

When you execute the service we can have Internal database, we can audit the service logs ,

Is it is success at that time u need to audit,

If it is Failure u need to audit

If it is Success r failure u need to audit

When the service is top-level service at that time only u need to audit

And including pipeline data you need to audit

**What is the difference between drop and delete pipeline variable?**

Drop: runtime we can drop the variable at run time

Delete: delete The Variable purely Design Time

Drop pipeline is an explicit cleanup. It is a request for the pipeline to remove a variablefrom the available list of variables and make the object it refers to available for garbage

collection by the Java Virtual Machine.

Delete is purely a design time operation to remove the variable from the current view. It isonly of use if you have created a variable that you didn't mean to create. If you delete a variable that was there because it was previously in the pipeline when you change the view in developer you will see the variable appear again.

**How to create a link between variables?**

Before this, service at which you are working must be locked. You have to map the Input Pipeline variable A to Output Pipeline variable B.

1. Create any new Flow service
2. Add some Input and Output variable as required
3. Insert any previously created service or insert any Map
4. Click at Pipeline In area then click on the variable A,
5. Click at Pipeline Out area then click on the variable B,
6. Now click over link button.

**Advantage of MAP flow step**

Performs specified editing operations on the pipeline (such as mapping variables in the pipeline, adding variables to the pipeline, dropping variables from the pipeline, initializing input values for a flow service – hard coding set value).

**What is a conditional map and how do you realize it**

In a conditional map or link, the link or map is executed (copies the value) only if the condition defined evaluates to true. It can be realized by applying conditions.

When the link appears blue it indicates conditions or index values for arrays have been applied to the link between variables.

Conditions cannot be added to the links between implicitly linked variables.

**What is the difference between implicit mapping and explicit mapping?**

Implicit map - Developer implicitly links variables whose names are the same and whose datatypes are compatible. Developer connects implicitly linked variables with a gray link. In a MAP step, Developer only displays the links between pipeline variables and

transformers. Developer does not display any implicit linking for a MAP step.

Explicit map - In cases where the services in a flow do not use the same names for a piece ofinformation, use the Pipeline tab to explicitly link the variables to each other.

Name and Structure transformations are realized using Explicit map. Developer connects explicitly linked variables with a solid black line.

**What Are Transformers?**

Transformers are the services you use to accomplish value transformations on the Pipeline tab. You can only insert a transformer into a MAP step. You can use any service as a transformer. This includes any Java, C or flow service that you create and any built-in services in WmPublic, such as the pub.date.getCurrentDateString and the pub.string.concat services. By using transformers, you can invoke multiple services (and perform multiple value transformations) in a single flow step.

**Do you suggest the usage of transformers? When and why should we use transformers and flow services? How are they different from each other?**

Mapping is the process of performing transformations to resolve data representation differences between services or document formats. By linking variables to each other on the Pipeline tab, you can accomplish name transformations and structural transformations.

However, to perform value transformations you must execute some code or logic. Developer provides two ways for you to invoke services:

You can insert INVOKE steps or you can insert transformers onto the Pipeline tab.

Transformers are the services you use to accomplish value transformations on the Pipeline tab.

When we are performing only one data operation we can go ahead with directly invoking the corresponding flow service. If we want to perform the multiple operations in a single step, it is good to go with transformers in a map step. This increases the memory usage and readability.

Both invoking Flow services and transformers can be used to provide the same functionality, but there are certain differences.

In case of transformers the order of execution is not strict (i.e. the order of transformer execution) is not determined and multiple transformers can be executed at the same time. And also transformers cannot be used whenever the underlying data structure dimension is

different (for ex: can't execute a transformer on a document when iterating over a loop on a document list)

Only transformers can accomplish multiple value transformations in a single flow step.

Service out parameters of the transformer service don’t get added to the main pipeline. Whereas in case of invoke the service out parameters flow to the main pipeline also.

Transformers can be used for any element under a document list on which we are looping over but cannot be used for child document list which is under the parent document list on which we are looping over.

**What are Structural transformations?**

Splitting one field into several or merging fields, reordering portions of a message (changing the structure) are known as structural transformations.

Other transformations are Name and Value.

**When do you use REPEAT, EXIT flow steps?**

REPEAT step is used when we want a particular step to be repeated provided with some conditions.

EXIT step is used to exit from any loop or from the whole flow step.

**What is the difference between a LOOP and a REPEAT?**

For Loop we need to give the input as document. For repeat we need to give the input as number.

Loop will exit by default after completing all iterations corresponding to number of documents in document list or elements in array. Repeat will loop for the specified number of times on Success / Failure of child steps

**I have a loop A, under which I have a child loop B, under which I have a Branch with few services with a condition. If a condition is satisfied, then I need my branch to exit from loop B, what should I do?**

Place an Exit Step inside the branch condition and set the property exit from to "Loop"

**What is the BRANCH operation?**

Branch operation conditionally executes an operation based on the value of a variable at run time

**How to use SEQUENCE as the Target of a BRANCH? How will you implement if-else and switch cases in webMethods using flow steps?**

Set evaluate label property of branch step to true. Then set the Label property of sequence with the value on which it needs to be processed. [if-else]

In addition, we can also use the branch in the context of switch case also, for that we have to set the evaluate label property of the branch to false and the set the value for the SWITCH property of the branch as the variable based on which you want to switch and set the label property of the branch as the possible value for the variable you gave in switch. [switchcase]

**When creating a BRANCH flow element, what is the purpose of the "scope" field on the properties tab?**

To restrict pipeline access to only the data in a particular document in the pipeline. Default scope is access to the entire pipeline.

**How to get the current index of the List in a LOOP**

There is a special variable on the pipeline called $iteration which will be incremented as the loop operator works up through the list.

**Difference between input and output array**

Input Array – It contains a list of elements which the LOOP step at run time passes over each

one of them individually at a time and makes them available as input for one or more steps

under the Loop. String list - String table - Document list - Object list

Out Array - If your LOOP step produces an output variable, the server can collect that output

for each iteration of the loop into an array in the pipeline.

**For a REPEAT operation to execute as long as the specified repeat condition remains true, the count parameter needs to be set to:**

The count parameter needs to be set to "-1”.

**If you specify REPEAT COUNT as 10, how many times does the loop repeat?**

The loop will repeat 11 times. (n+1)

0 – The child steps wont be repeated

-1 – repeats as long as the specified Repeat on condition is true. (Repeat on – Success or

Failure)

**What is the default behavior, if a EXIT does not specify a "from"?**

$loop will be assumed, and a com.wm.lang.flow.FlowException will be thrown if the EXIT is not in a LOOP

The EXIT will throw anjava.lang.NullPointerException.

**When creating Flow services, what is the purpose of a SEQUENCE operation?**

The purpose of Sequence operation is to group a subset of Flow operations so that they are treated as a unit.

**When using the SEQUENCE Flow operation, if exit-on is set to SUCCESS, what condition will cause the entire SEQUENCE to fail?**

When all of the child operations fail.

**How we can catch exception error on run time mode using Flow language in WebMethods??? How we can send error description as an output parameter?**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Sequence | | | ---------- | main block | ------ Succeed |  |
| |-- | Sequence ------------- | try block------ | | | | Failure |
| |-- | Sequence ------------- | catch block----- | | | | Done |
|  | |-- | | InvokegetLastError | | service |  |

**How to handle exceptions in webMethods? What is Nested Sequencing?**

To handle exceptions in webMethods, we use three sequence steps – two child sequence steps nested under a parent sequence step as below -

First sequence (exit on success)

* Second sequence (exit on failure)
* Third sequence (exit on done)

Second and third sequences are child steps of first one. All the business logic will come inside second sequence.

What we have to do when exception is there, this logic will come in third sequence.

**In the following case what is the outcome**

Sequence (Success)

Sequence (Success)

Sequence (Failure)

It will work if there is no exception. In case there is an exception, it is never handled as the flow never reaches the Catch Block.

**What is scope?**

Scope Property is used to restrict flow step to access only the elements of aIData Object. For Example if document x has elements x1,x2,x3,lets and we set the scope as X for a MAP flow - we can do operations only on x1,x2,x3 when the control is in MAP step.

**What Are Input and Output Parameters?**

Variables having a name and datatype which serve as inputs and resulting outputs to services.

**What are the different data types?**

String - String of characters

String list - A one dimensional String array String table - A two dimensional String array

Document - A data structure that is a container for other variables. The contents of a document (IData object) are stored as key/value pairs where the variable name is the key Document list - A one dimensional array of IS document types (IData [ ]or Values [ ])

Document reference - A document whose structure is defined by an IS document type

Document reference list - A document list whose structure is defined by an IS document type. Object - A data type that does not fall into any of the data types described in the aboverows, and is not declared to be one of the basic Java classes supported natively by Integration Server

Object list - An array of Objects of unknown type

**What is the use of Document Reference?**

Document reference - A document whose structure is defined by an IS document type.

An IS document type contains a set of fields used to define the structure and type of data

In a document (I Data object). You can use an IS document type to specify input or output parameters for a service or specification.

* IS document types improve accuracy, because there is less opportunity to introduce a typing error typing field names.
* reduces the effort needed to declare input or output parameters or the effort/time needed to build other document fields.
* make future changes easier to implement, because you can make a change in one place (the IS document type) rather than everywhere the IS document type is used

**What is a specification?**

A specification is a ‘free standing’ IS element that defines a set of service inputs and outputs. If you have multiple services with the same input and output requirements, you can point each service to a single specification rather than manually specify individual input and output fields in each service.

* Reduces the effort required to build each flow.
* Improves accuracy, because there is less opportunity to introduce a typing error when defining a field name.
* Makes future specification changes easier to implement, because you can make the change in one place (the specification) rather than in each individual service.

**What is a canonical form? Why do you go for it?**

A canonical document is a standardized representation that a document might assume while it is passing through the webMethods Integration Platform. A canonical document acts as the intermediary data format between resources. For example, in an implementation that acceptspurchase orders from companies, one of the steps in the process converts the purchase order document to a company’s standard purchase order format. This format is called the 'canonical' form of the purchase order document. The canonical document is published, delivered, and passed to services that process purchase orders.

**What Is Data Validation?**

Data Validation is the process of verifying that run-time data conforms to a predefined structure and format. It also verifies that the run-time data is a specific data type and falls within a defined range of values.

**What Is Data Validated Against?**

During validation, run-time data is compared to a blueprint or model. The blueprint or model is a formal description of the structure and the allowable content for the data. The blueprint identifies structural and content constraints for the data being validated. The validation engine in webMethods Integration Server considers the data to be valid when it conforms to the

constraints specified in the blueprint. A blueprint can be an IS schema, an IS document type, or a set of input and output parameters.

**What Does an IS Schema Look Like?**

The Appearance and content of IS Schema depends on whether you generate an Is Schema from XML schema or DTD. If you create an IS schema from xml schema, the resulting IS schema displays type definitions, element declarations and attribute declarations.

If you create an IS schema from DTD, the resulting IS schema displays element type declarations.

**How to Create an IS Schema?**

An IS schema can be created by following these steps:

1. First create a webMethods document
2. Run the service pub.schema:createXSD
3. Give the input parameter the fully qualified name of your document. The output parameter Xsd /source has the text schema representation of your document. Save this to a file

**What is the difference between Schema and SchemaDT?**

Schema is a flat file schema. SchemaDT is a IS Document Type created/generated from this schema blue print.

**What is a flat file schema and what are its uses?**

Flat file schema can be used to validate the incoming flat files. Schema provides definition for flat files against which IS can use to validate flat files.

Flat file schema is a blueprint which has rules for the flat file. IS will validate the flat file against this Flat file schema.

**What is a dictionary?**

A flat file dictionary contains record definitions, field definitions, and composite definitions that can be used in multiple flat file schemas.

Dictionary is nothing but a naming layout of flat files. By using dictionary we can define the structure of flat files, it can contains record, field, and composite definitions.

**The Developer was used to create a Java Service named myService in a folder named myFolder in the Default package. What is the best way to hide the source code of the Java service?**

Configure an Access Control List which only allows members of authorized groups and assign it as the Read ACL for the service

**What is a node? How will you get a value from an xml (which service)? What are some of the XML services?**

A node is the result of a parsing operation. It is an element-based representation of an XML document. The node expresses a document in a tree like structure that allows the data

Within it to be efficiently addressed and linked into services.

pub.xml:loadXMLNode- Retrieves an XML document via HTTP or HTTPS, parses it, and produces anXML node.

pub.xml:xmlStringToXMLNode- Converts an XML document (represented as a String, byte[ ], orInputStream) to an XML node.

pub.xml:xmlNodeToDocument- Converts an XML node to a document (an IData object).

pub.xml:documentToXMLString- Converts a document (IData object) to an XML string.

**Which service can be used to get value for a particular element in an XML document?**

pub.xml:queryXMLNode- Queries an XML node.

**If you want to split the one xml file into two xml files, then how will you handle that scenario?**

pub.xml:QueryXMLNode

The fields parameter specifies how data is extracted from the node to produce an output variable. This output variable is called a “binding,” because the fields parameter binds a certain part of the document to a particular output variable. At run time, this service must include at least one fields entry. The service must include at least one entry in fields. The result of each query you specify in fields is returned in a variable whose name and type you specify.

**In how many ways we can invoke a flow service within another flow service?**

Developer provides two ways for you to invoke services:

You can insert INVOKE steps or you can insert transformers onto the Pipeline tab.

pub.remote:invokeinWmPublicpackage

Invokes a service on a remote webMethods Integration Server. (remote-invoke)

The remote server is identified by an alias, which is configured on the Remote Servers tab in the Integration Server Administrator. Connection and authentication to the remote server is managed transparently to the caller of this service.

All current pipeline inputs are passed to the remote service.

**What are the input parameters for the doinvoke service?**

public static IDatadoInvoke(NSName name, IData input)

Parameters:

name - Fully-qualified service name (folder:service). input - Input values for service.

Returns: output IData from called service.

public static Values doInvoke(NSName name, Session session, Values input)

Parameters:

name - Fully-qualified service name (folder:service). session - Current session object.

input - Input values for service. Returns: output Values from called service.

public static IDatadoInvoke(NSName name, Session session, IData input)

Parameters:

name - Fully-qualified service name (folder:service). session - Current session object.

input - Input values for service. Returns: output IData from called service.

**What is a database service? Can you give an example?**

Services in pub.db in WmPublic package. It provides for the storage and retrieval of data in a database. pub.db:connect - pub.db:query - pub.db:insert - pub.db:getTablesetc

We can write advanced database Services using techniques such as connection pooling, transactions and stored procedures.

**What is the service you use in error sequence?**

pub.flow:getLastError - Obtains detailed information about the last exception that was trappedwithin a flow.

pub.flow:debugLog - Writes a message to the server log.

Application Error - Expected System Error - Unexpected System Error

The BRANCH operator is handy for managing logic and application errors. By creating BRANCH logic with "case" or "if-else" structure, a developer can determine the exact error condition and then take the appropriate action.

The EXIT operator can be used to exit from an entire Flow or just a section of Flow. The developer can designate that the exit is an exception by signaling "FAILURE" or that the error has been caught by signaling "SUCCESS".

webMethods Integration Server also ships with Event Manager. Event Manager is very useful as an umbrella for error notification because it catches exception events for the entire server. Its broad scope, however, can work against you as your system can quickly flood with unwanted notifications. Because of this, Event Manager must be configured for filtering so that it only catches errors that are required by the developers.

Most common error reporting method is email notification. An error should be classified and reported to any relevant applications as well as delivered via email to developers. This approach helps to deal with critical issues quickly.

A generic Error Handling service can be written to manage these tasks. It should reside in a utility-level package from which the entire corporate implementation will benefit. Then, a customized Error Handling service should be written to add an additional abstraction layer. This step provides freedom to change the error notification logic without changing the underlying business integration logic or performing additional regression tests.

1. Using the Event Manager, assign ErrorHandler.event:exceptionHandler to Exception Event.
2. Set the appropriate filters for Exception Events. Remember that the Exception Event will not fire if the exception-throwing service is not included in the Filter field.

Throwing errors

Using Flow, though, a developer must process the error differently. Using Flow, a developer will call the EXIT operator and provide a detailed message for the failed action. Following a standard naming convention is good practice for using EXIT because it helps to parse and then route the error message to the proper recipient or target tracking system.

Once assigned, the service will be invoked with every exception satisfying the filter. The default service behavior of ErrorHandler.event:exceptionHandler sends email to an intended recipient. This service can be modified to accommodate issue tracking software.

Another technique for handling Flow exceptions is to create a Try-Catch block.

**When Is a Copy of the Input Pipeline Saved in the Audit Log?**

You need to set the properties allow pipeline in auditlog properties.

**What happens when the pub.flow:tracePipeline service is invoked?**

The Integration Server logs the name-value pairs in the pipeline at that time

**What is the primary function of the built-in pub.flow:savePipeline service?**

Save the current pipeline to a named memory location on the Integration Server

**How to preserve existing pipeline before a restorePipeline step?**

Set the "$merge" variable in restorePipeline or "merge" in restorePipelineFromFile to be true. This will ensure that everything in the pipeline before a restorePipeline step is preserved.

**How to convert document list to document? If document list contain documents and documents contain strings field?**

INVOKE pub.document:documentListToDocument

**After a default installation, in order to use the pub.file:getFile service, what needs to be done?**

pub.file:getFile does not require any modifications to the Integration Server.

**How do you create WebService connector and WSDL file?**

A Web service connector. A Web service connector is a flow service that invokes a Web service located on a remote server. Developer automatically generates a Web service connector when it creates a Web service descriptor for a consumer Web service. Developer can also create a Web service connector from an existing WSDL.

A Web service descriptor is an IS namespace element that contains the definition of an IS Web service. A WSD describes either a ‘provider’ or a ‘consumer’ Web service.

A provider Web service allows an external user to invoke an existing IS service as an ‘operation’ of the Web service.

A consumer Web services are external Web services that can be invoked from within the local Integration Server.

**What is the primary purpose of a Web Service Connector?**

The purpose of Web Service Connector is to invoke a Web Service on a different web server

Why do we use end point aliases in web services?what is its uses?

if you do end point Alias then when you deploy your code from one server to other server then no need to change the code level IP address just change in the config level

**What is UDDI Registry Tab?**

UDDI Registry is used to connect to and disconnect from a UDDI Registry. Once you have opened a UDDI Registry, you can display, filter, and publish Web services in that registry. Within the UDDI Registry tab, Web services are sorted in alphabetical order. Simply select a Web service to view more information about the service.

**Explain about WSDL and its use in a wM Integration scenario**

A Web service is a software application that can be accessed remotely, using XML based languages to communicate. From a Web service or a WSDL, you can create a consumer Web service descriptor and connector. Developer can invoke the connector to run the remote Web service.

From an existing IS service or WSDL, you can create a provider Web service descriptor. You can then publish the Web service descriptor to a UDDI Registry so that the IS service it describes can be invoked by an external user as an “operation” of the Web service.

**What is java service?**

A service in Integration Server developed using Java and the Integration Server Java API.

**How can we develop a java service?**

We can use the developer IDE or any independent IDE to create a Java Service by making use of the Integration Server Java API.

**How Java Services Are organized on the webMethods Server?**

All Java services in same folder are stored as methods in one java class which has the name of the folder.

**What happens if we save a java service?**

On saving, the Java Service is compiled.

A Java service is a public static method in a Java class file on webMethods Integration Server.

* The service name represents the Java method name.
* The interface name represents the fully qualified Java class name.

**When coding IS Services, how can a variable of type Document Type be represented in Java?**

Variable of type Document Type be represented as "IData”

**How can we extract the output variables from the java service? What is IData Object? Difference between IData and IDataCursor**

The IData object is the universal container that services use to receive input from and deliver output to other programs. It contains an ordered collection of key/value pairs on which a service operates. An IData object can contain any number of key/values pairs (elements). The keys in an IData object must be Strings. The values can be any Java objects (including IData objects).

Services Take IData Objects as Input and Return IData as Output. For getting data from and putting data into IDataelements:

* Position the cursor at the IData element.
* you get or set the data in that element

The class that you can use to position a cursor in an IData object is IDataCursor. The IDataCursor class contains methods for performing basic cursor operations such as placing the cursor at the first, last, or next element in the object. There is getValue or setValue methods to read or write the value of that element, respectively. This class also provides methods for inserting new elements, getting key names, and

deleting elements.

static

{

myObject = IDataFactory.create(); IDataCursormyCursor = myObject.getCursor();

myCursor.insertAfter("VA", new Double("0.045")); myCursor.insertAfter("MD", new Double("0.05"));

myCursor.insertAfter("DE", new Double("0.0"));

}

webMethods Integration Server Java API Reference

**What is the difference between Java Service and Flow Service?**

Java is the native language for services. A Java service is developed using Java and the IS Java API whereas a Flow Service is written using the proprietary FLOW language (GUI-based tool for developing FLOW services)

Although flow services are written as XML files, they are maintained in a format that can only be created and understood by Developer. You cannot create or edit a flow service with a text editor unlike the java services.

- It is more Service Oriented in its approach as it leverages a lot of built in services.

**When do you use a flow service and a java service?**

Flow is not designed to be used to create user interfaces, carry out complex mathematical functions or to create general purpose application business logic.

Java services are used in situations where Flow needs to be extended to accomplish a task not provided in a built-in service or to leverage business logic contained in a third-party java library. Java services are most often used inside of Flow services to provide a needed function or transformation.

It is not a best practice to build a large percentage of an integration application using IS java services. Doing so does not leverage the productivity advantages of Flow and does not provide significant portability advantages.

**How do we invoke another service in a Java service and in which package does this method reside?**

You can invoke flow Service or even java service within a java service using

Service.doInvoke(). It is present in the package com.wm.app.b2b.server.Service.

Alternative and an easier way is generate code using webMethods's function: Tools > generatecode - For calling this service from another service

**If a Service suddenly hangs, how would you know this?**

Can be seen in IS.

**How do you setup a Scheduled Service?**

In IS Administrator page Click the Server > Scheduler we can Create a scheduled task for a specific service.

**What Happens When a Breakpoint is encountered?**

When you execute a service that contains a breakpoint or call a child service that contains a breakpoint, the service is executed up to, but not including, the designated breakpoint step. At this point, processing stops and will not resume until you select another one of Developers debugging commands.

Note: if you want Developer to stop at subsequent breakpoints, you must select the Trace Into command.

What Happens When an Event Occurs?

There are various types of events that occur in Integration Server i.e. Session start, session end, session timeout, alarm, exception etc. These can be seen in Developer Event Manager. Go

to Tools menu in the menu bar on the developer and click on Event Manager. We can invoke a service when a particular event occurs in this event manager. E.g. we can invoke a particular service when an exception occurs in any of the IS services. This is helpful when we want to implement custom/generic exception handling framework.

**What Are Event Handlers?**

Event handlers are the services that you write to perform some actions when a specific event occurs.

**How to create DSP pages**

A dynamic server page (DSP) is a document embedded with special codes (tags) that instruct the webMethods Integration Server to perform certain actions when an HTTP (or HTTPS) client requests the document.

DSPs are used to construct browser based applications. Because they are HTML based, they can be used to build complex user interfaces that includes any valid construct (e.g., forms, cascading style sheets, JavaScript) recognized by the client’s browser.

They allow you to conditionally execute a service based on run-time input. Eg: you might build a DSP that actually contains several different HTML pages, and use the %switch% tag to select among them.

To create a DSP, you must compose it with a text editor and then save it on the Integration Server with extension .dsp - Place the DSP file in the pub directory of the package in which you want the dsp to reside.

**What is an Output Template?**

Output templates allow you to insert output values from a service into a document that youDefine. Output templates are similar to DSPs. In fact, the tags you use to compose DSPs are the same ones you use to compose output templates. However, DSPs have one additional tag—the %invoke% tag—that distinguishes them from output templates. This tag allows a

DSP to invoke a service.

IS\packages\packageName\templates directory

To assign an output template to a service we bind the name in the properties panel of a flow service.

**How do we perform testing in webMethods? What is unit testing and how do you do unit testing?**

Unit testing helps ensure quality in the software development process. A webMethods service makes a good candidate as a unit for testing because each is treated as a black box of business logic. Each service has an expected set of pre-conditions and post-conditions.

The implementation of a unit test is simplified by using an open-source standard like JUnit. With a simple Java test client, for example, JUnit can execute services for the creation of a unit test library.

JUnit can be used to test both webMethods Flow and Java services. However, this requires a utility capable of accessing webMethods services as a Java client.

The Java client allows individual JUnit tests to invoke webMethods services . Furthermore, the Java client allows access to the IData pipeline so that control is granted over input and output parameters.

**GH Tester for webMethods**

**How to test the business process?**

We can design, simulate, test and deploy processes—then in real-time monitor those processes using webMethods Business Process Management Suite (BPMS).

**How is security handled in Wm?**

ACL (Access control lists)

Alias for remote-invoke

Security in Trading Networks

* Communicating Securely Using SSL
* Protecting Access to User Interfaces
* Protecting Partner Profile Passwords
* Protecting Access to Trading Networks Processing
* Certificates for Verifying, Signing, Encrypting, and Decrypting Documents

**Why do we use certificates?**

Digital certificates are electronic credentials that are used to certify the identities of individuals, computers, and other entities on a network.

Digital certificates function similarly to identification cards such as passports and drivers’ licenses. For example, passports and drivers’ licenses are issued by recognized government authorities, whereas digital certificates are issued by recognized certification authorities(CAs).

A digital certificate contains an entity's name, address, serial number, public key, expiration date and digital signature, among other information. When a Web browser like Firefox, Netscape or Internet Explorer makes a secure connection, the digital certificate is automatically turned over for review. The browser checks it for anomalies or problems, and pops up an alert if any are found. When digital certificates are in order, the browser completes secure connections without interruption.

Trading Networks supports x.509 certificates for verifying the digital signature of documents sent by a partner. Certificates are also used to verify, sign, encrypt and decrypt.

**What is point-to-point messaging?**

With point-to-point (P2P) message passing the sending application/client establishes a named message queue in the JMS broker/server and sends messages to this queue. The receiving client registers with the broker to receive messages posted to this queue. There is a one-to-one relationship between the sending and receiving clients.

**What is pub-sub messaging?**

Publish/subscribe (or pub/sub) is an asynchronous messaging paradigm where senders (publishers) of messages are not programmed to send their messages to specific receivers (subscribers). Rather, published messages are characterized into classes, without knowledge of what (if any) subscribers there may be. Subscribers express interest in one or more classes, and only receive messages that are of interest, without knowledge of what (if any) publishers there are. This decoupling of publishers and subscribers can allow for greater scalability and a more dynamic network topology.

Pub/sub is a sibling of the Message Queue paradigm, and is typically one part of a larger Message-Oriented Middleware solution. Most messaging systems support in their API (e.g. JMS) both the pub/sub and Message Queue models.

**What is JMS?**

Java message service is a java frame work specification for messaging between applications.

**What is a Broker?**

The Broker is a high-speed message router. It enables asynchronous, message-based solutions that are built on the publish-and-subscribe model.

The role of the Broker is to route documents between information producers (publishers) and information consumers (subscribers). In an integration solution, webMethods components such as the Integration Server and the Workflow Server are both publishers and subscribers.

The Broker maintains list of subscribers that are interested in receiving certain types of documents. When the Broker receives a published document, it queues the document for the subscribers of that document type. Subscribers receive the documents from their queues, which usually triggers an action on the subscriber’s system that processes the document.

A Broker contains document types, client groups, and client state objects. The primary job of the broker is to manage queues for message delivery

**What is the difference between Synchronous and Asynchronous communication? How do you achieve it?**

In synchronous communication, the initiating party (sender) waits for a response (from receiver) before moving on to other tasks. Achieved using request/reply mechanism.

1) the publisher waits for a response 2) one or more repliers will deliver (as opposed to publish) a reply to the requestor. This is a way to obtain essentially synchronous interaction over an asynchronous mechanism.

In asynchronous communication, the initiating process will initiate the process, such as starting a thread or sending a message, and will not wait for a response but instead will move on to other processing. When a response is desired/required, the response comes via another avenue to complete the interaction. Achieved using publish/subscribe mechanism.

This is designed to decouple applications. A system will indicate an interest to a broker in one or more types of events. This is referred to as a subscription. Publishers publish events, which have a specific type. When the broker receives events of that type, it copies the event to all subscribers. This technique allows publishers to not care what systems are consuming data, and allows subscribers to not care what system(s) produced the data.

**What is Publish-and-Subscribe Model?**

The publish-and-subscribe model is a specific type of message-based solution in which messages (carrying documents) are exchanged anonymously through a message broker. Under this model, applications that produce information make that information available in specific types of recognizable documents that they publish to the Broker. Applications that require information subscribe to the specific types of documents that they need.

**What is a territory?**

A territory is a group of inter-connected Brokers that functions as one logical Broker. All Brokers in a territory maintain the same set of document types and client groups, however, they each support their own set of clients. Brokers that operate in a territory are peers.

A territory enables you to scale a Broker to handle large numbers of clients by spreading the client population across multiple host machines.

For additional flexibility, you can create multiple territories and link them together using gateways. A gateway is a connection that allows specified documents to flow between twoterritories.

An integration platform can contain multiple Brokers that operate in groups called Territories. A Territory Gateway is a connection between two territories.

Rules Concerning Territories

* A Broker that is not part of a territory does not have knowledge of any other Broker.
* Brokers within a territory have knowledge only of the other Brokers currently in the same territory.
* Once a Broker leaves a territory, it loses knowledge of all other Brokers in the territory.
* A Broker can be a member of only one territory at a time. To change from one territory to another, a Broker must leave the first territory and then join the second.
* Operations on document types and client groups affect all Brokers in the territory.
* All Brokers in a territory share the same client groups and document types. In effect, they appear to operate under a single configuration.
* A territory cannot be empty. To create one, you must find a Broker that does not belong to any other territory.

**What are client groups?**

Client groups contain a list of clients. It has a set of client properties and permissions. ACL can be associated with it. It can configure documents its client could publish and subscribe to. A Broker contains document types, client groups, and client state objects.

**Do you require a broker if you are sending the document locally. If not what type of delivery should you mention?**

Local publishing refers to the process of publishing a document within the Integration Server. Only subscribers located on the same Integration Server can receive and process the document. In local publishing, the document remains within the Integration Server. There is no Broker involvement.

In the Document Type properties we need to define Publishable as True and Broker doc type as Publishable Locally Only [Indicates that instances of the publishable document type can be used in local publishes only. This publishable document type does not have a corresponding Broker document type]

pub.publish:publish

pub.publish:publishAndWait

A Broker/local trigger is trigger that subscribes to and processes documents published/delivered locally or to the Broker.

**How do you do the broker settings and what is the default Broker port number?**

A Broker contains document types, client groups, and client state objects. They are actually data structures that encompass a collection of objects upon which processing threads within Broker Server operate. It partitions the publish and subscribe capabilities of a Broker Server into separate domains or namespaces

In My webMethods: Messaging > Broker Servers. Brokers tab Add Broker

Broker Name, Description, If you want this Broker to act as the default Broker enable the

Default Broker option.

* Document Type Logging
* Dead Letter Queue : The dead letter queue is a queue in which Broker places dead letters. A

dead letter is a document for which the Broker has no subscribers.

The default port number is 6849.

**What is guaranteed delivery in broker and how do you ensure it?**

Guaranteed document delivery ensures that a document, once published, is delivered at leastonce to the subscribing triggers.

At least once processing and exactly once processing are types of guaranteed processing. Inguaranteed processing, the Integration Server ensures that the trigger processes the document once it arrives in the trigger queue. The server provides guaranteed processing

for documents with a guaranteed storage type.

A characteristic of a delivery mechanism that assures delivery of a document or event. The transport used by Enterprise Server and the guaranteed delivery facility within Integration Server provide guaranteed delivery

Broker is used to distribute your Integration Servers for asynchronous communication and ensure guaranteed delivery.

pub.publish:publishAndWait - sends a document to any and all resources interested in that document type and waits to receive a reply from the processing resource.

pub.publish:deliverAndWait - sends a document to a specific resource and waits for a reply.

Guaranteed delivery is a facility of wM IS that ensures guaranteed, one-time execution of services. It protects transactional requests from transient failures that might occur on the network, in the client, or on the server.

A transient failure is a failure that can correct itself within a specified period of time. If a request cannot be delivered to the server due to a transient failure, the request is resubmitted. If the problem corrected itself, the request is successfully delivered on a subsequent attempt.

You can determine what constitutes a transient error by specifying a time-to-live (TTL) period for a guaranteed delivery transaction and, optionally, the number of times a transaction should be retried. If you do not specify the TTL or retry

value, the configured defaults are used. You can use guaranteed

You can use guaranteed delivery when you invoke a service from a client or from within another service. You can only use the guaranteed delivery capabilities with stateless (that is, atomic) transactions. As a result, guaranteed delivery capabilities cannot be used with multi-request conversational services.

**What is the difference between the Guaranteed document delivery and Guaranteed document processing?**

The document storage type setting determines how instances of that document are persisted. Documents published to Broker can be stored as volatile documents or guaranteed documents.

Volatile documents are stored in local memory only. These documents are lost if the Brokerhost experiences a service interruption or the Broker Server is restarted.

* higher performance than guaranteed storage
* suited for documents that have a short life span or are not critical

Guaranteed documents are persisted to disk so that they can be recovered in the event ofa power failure or a server restart.

* Default setting
* lower performance than volatile storage
* safest and preferred for critical documents

The Integration Server returns acknowledgements for guaranteed documents only. IS does not return acknowledgements for volatile documents

Guaranteed document delivery ensures that a document, once published, is delivered at leastonce to the subscribing triggers. Guaranteed document processing ensures that a trigger makes one or more attempts to process the document.

**What is a trigger? What is its use and what are the different types of triggers?**

Triggers subscribe to documents and invoke services that process the documents.

Subscriptions to specific documents are created via Trigger

Triggers define

* What Documents to listen for
* What to do when the Document arrives

Delivery does not require a trigger to be present on the client in order for the client to receive the document.

Trigger Conditions

* Which Document Type(s) to subscribe to
* What Filters to use when subscribing
* Which Service(s) to invoke on receipt

**What are Filters in Triggers**

To further refine a condition for the publishable document types, Filters are used.

Filters are used to specify conditions based on the contents of a document

* All elements of a document are available for filtering
* Use regular expressions to set conditions

Eg: Do not consume orders where Sender is ‘IBM’

**What are Join types in trigger. Difference between AND, OR, XOR options in Triggers?**

A single Trigger may subscribe to multiple document types using multiple conditions in a

trigger. The trigger will only invoke the service when the Join condition is met:

* AND : All of the subscribed document types have been published
* OR : Any of the subscribed document types have been published
* XOR : Only one of the subscribed document types has been published

**How will Broker identify a particular Trigger?**

Based on the published Document and the defined conditions.

**What is the difference between concurrent / parallel and serial processing in trigger?**

Message processing determines how the Integration Server processes the documents in the trigger queue. Trigger properties panel Processing Mode – Serial / Concurrent

Serial Processing - IS processes the documents in the trigger queue one after the other. ISretrieves the first document in the trigger queue, determines which condition the document satisfies, and executes the service specified in the trigger condition. It waits for the service to finish executing before retrieving the next document from the trigger queue.

Processes documents in the trigger queue in the same order in which it retrieves the documents from the Broker. Maintains publication order.

Concurrent Processing - IS processes the documents in the trigger queue in parallel. (ISprocesses as many documents in the trigger queue as it can at the same time) IS does not wait for the service specified in the trigger condition to finish executing before it begins processing the next document in the trigger queue. You can specify the maximum number of documents the IS can process concurrently. Faster performance than serial processing.

**If two documents match in the trigger then how do you process them?**

Make use of Joins to handle the processing.

**How does IS connect to the Broker?**

* Open the Integration Server Administrator.
* In the Settings menu of the Navigation panel, click Messaging.
* Under Broker Configuration, click Broker Settings.
* Click Edit Broker Settings.

Broker Host - Broker Name - Client Group etc

Brokers use the pub-sub model to connect to the IS. A pub-sub solution on an Integration Server is made up of services and triggers. Services are units of logic that publish information to the Broker. Triggers subscribe to documents and invoke services that process the documents.

**How many brokers can be connected to IS?**

Only 1 Broker can be connected to 1 IS

**If a broker is connecting to IS and involved in publish-subscribe mechanism, if the published document is changed, then how will it reflect in broker? When changes are made to the document that was already published to broker then how will be the document updated in the broker?**

Each publishable document type on your Integration Server has a synchronization status to indicate whether it is in sync with the Broker document type, out of sync with the Broker document type, or not associated with a Broker document type.

Updated Locally - Updated on Broker - Updated Both Locally and on the Broker - Created Locally - Removed from Broker - In Sync with Broker

Synchronization Actions needs to be performed to match the documents :

Push to Broker - Update the Broker document type with information from the publishabledocument type.

Pull from Broker - Update the publishable document type with information from the Brokerdocument type.

Skip – skips the synchronization action

Synchronizing Multiple Document Types Simultaneously

* Sync All Out-of-Sync Document Types
* Sync All Document Types

**Can you see the document in Broker?**

Yes, document can be seen in Transactions of Broker Admin page (wM 6.x)

Messaging > Broker Servers > Brokers Broker Details - transactions tab (wM 7.x)

**If document fails in broker how to send it again?**

We can resubmit the document using webMethods Monitor.

Suspend and retry later option provides a way to resubmit the document programmatically.

**What is Message store / Document store? How does webMethods handle message queuing?**

The Integration Server uses document stores to save published documents to disk or to memory while the documents are in transit or waiting to be processed. The Integration Server maintains three document stores for published documents.

* Default document store
* Trigger document store : contains documents waiting to be Processed by triggers until

successfully processing. Trigger queues associated with each trigger in this store

- Outbound document store : contains documents waiting to be sent to the Broker. The Integration Server places documents in the outbound document store when the configured Broker is not available.

**Where does the document get stored when the subscriber of that document is not connected to the Broker?**

The document gets stored in the dead letter queue.

The dead letter queue is a queue in which Broker places dead letters. A dead letter is a document for which the Broker has no subscribers / that are addressed to non existent clients.

**When there are ten subscribers connected to a broker for a similar document type, how will you publish to**

**only a particular subscriber?**

We use the Deliver or Deliver & Wait messaging models. Documents are published to a specific client, and a reply may or may not be expected from that client

pub.publish:deliver

pub.publish:deliverAndWait

**What is an adapter and what is an agent?**

An Adapter is a gateway between the Integration Server System and other resources. It is hosted on the IS and communicates with IS and Resources. They are out of the box ready to run components – no coding is required – only configuration needed.

Two models of operation – Invocation model and Notification model.

An Agent is a type of webMethods Enterprise Server adapter that, rather than connecting to aresource of some sort, such as a database, provides functionality on its own. Examples include the Integration Logic Agent, Rules Agent and Data Transformation Agent.

**What is the difference between Adapter Service and Adapter Notification?**

Adapter service connects to an adapter’s resource and initiates an operation on the resource.Adapter services are created using service templates included with the adapter.

Adapter notification enables an adapter to receive event data from the adapter’s resource.

There are two types of adapter notifications:

* Polling notifications : which poll the resource for events that occur on the resource.
* Listener notifications : which work with listeners to detect and process events that occur

on the adapter resource.

Listener is an object that connects to an adapter resource and waits for the resource todeliver data when an event occurs on the resource. Listeners work with listener notifications to detect and process event data on the adapter resource.

**What do you mean by Notification and where is it used?**

Adapter notifications notify webMethods components whenever a specific event occurs on an adapter’s resource. The adapter notification publishes a document when the specified event occurs on the resource.

For example, if you are using the JDBC Adapter and a change occurs in a database table that an adapter notification is monitoring, the adapter notification publishes a document containing data from the event and sends it to the Integration Server.

An adapter notification can have an associated publishable document type.

**How do you map data into a database?**

By mapping the data to the input document in Adapter service

**How webMethods Developer Supports Tables?**

There is an element called Adapter Service in Web methods using this element we can create adapter service like insert, update and delete, select the records from table. Before that we have to create Adapter Connection in Integration server under JDBC Adapter.

**What is connection pooling?**

A connection pool is a cache of database connections maintained by the database so that the connections can be reused when the database receives future requests for data. Used to enhance the performance of executing commands on a database as a new connection does not have to be established.

A connection pool is a collection of connections with the same set of attributes. The Integration Server maintains connection pools in memory. Connection pools improve performance by enabling adapter services to reuse open connections instead of opening new connections.

**How should I organize connection pools?**

If you have adapter notifications and adapter services then you will need to have two separate connections. Otherwise you may get strange errors about transactions and the like.

You should also avoid having connection pools shared across different functional areas, even if they are pointing to the same database. The reason for this is that tuning the size of the pool becomes quite difficult if you have multiple types of usage of a pool. You are also unable to easily change the database settings for one without impacting on the other.

One approach that seems to work quite well is to have separate pools for each package (generally not a hard and fast rule though), as your packages should generally be divided up according to functional area too.

**How do you configure an adapter connection for oracle**

In Administrative page JDBC Adapter will be there. By giving parameters like Datasource class, server name, database name port number and after enabling it.

Oracle JDBC Thin Driver

For NO\_TRANSACTION and LOCAL\_TRANSACTION - oracle.jdbc.pool.OracleDataSource

For XA\_TRANSACTION - oracle.jdbc.xa.client.OracleXADataSource

**What are the parameters you specify for connection pooling in configuration of a JDBC adapter?**

We need to specify Package name, Folder Name, Connection Name, Transaction Type, Data Source, Server Name, Database name, Port number, User Name, pass word.

**What is a Transaction?**

Transaction is nothing but a database operation like update

**What are Implicit and Explicit Transactions?**

With implicit transactions, the Integration Server automatically manages both local and XA Resource transactions without requiring you to explicitly do anything. That is, the Integration Server starts and completes an implicit transaction with no additional service calls required by the adapter user.

With explicit transactions when you need to explicitly control the transactional units of work. To do this, you use additional services, known as built-in services, in your flow. WmART Package has 4 transactions, they are:

pub.art.transaction:startTransaction

pub.art.transaction:commitTransaction

pub.art.transaction:rollbackTransaction

pub.art.transaction:setTransactionTimeout

**What are transaction types in JDBC Adapter? What are different transaction types while configuring an**

**adapter connection? What is an XA transaction?**

No\_Transaction: The JDBC Connection itself will take of the Transaction Management (Start,

Commit or End)

Local\_Transaction: The Developer needs to take care of the Transaction Management and need tospecify the Start, Commit or End Transaction Points in the Code. For this we have Built-In Services in WmART package

XA\_Transaction

The JDBC Adapter supports Automatic (NO\_TRANSACTION), Controlled (LOCAL\_TRANSACTION), and 2-phased (XA\_TRANSACTION) transactions.

An XA transaction, in the most general terms, is a "global transaction" that may span multiple resources. A non-XA transaction always involves just one resource. An XA transaction involves a coordinating transaction manager, with one or more databases (or other resources, like JMS) all involved in a single global transaction. Non-XA transactions have no transaction coordinator, and a single resource is doing all its transaction work itself (this is sometimes called local transactions).

In Automatic transaction mode, the adapter handles the transaction boundaries. In Controlled transaction mode, users have control over the transactions. Using XA JDBC drivers, the adapter

|  |  |  |
| --- | --- | --- |
| can | support 2-phase transactions | executed across multiple databases. The adapter works with |
| the | webMethods Integration Server | to act as the transaction manager for 2-phase transactions. |

**What is NO transaction type? When will it commit those transactions?**

When we use this type of transaction, it will automatically commit the operations / transactions.

After insertion of data into the database the commit will automatically be done.

**In JDBC adapter what was the driver you used? Was the driver you used was given by vendor or you have configured your own driver?**

Oracle JDBC Thin Driver. Data Source - oracle.jdbc.pool.OracleDataSourceojdbc14.jar placed in \webMethods7\IntegrationServer\lib\jars

Driver provided by the vendor.

**Why do you assign a particular number to the minimum connections (what happens if you give 0 & 1?) What is the number of minimum connections in a JDBC Adapter?**

When you enable a connection, the Integration Server initializes the connection pool, creating the number of connection instances you specified in the connection’s Minimum Pool Size field when you configured the connection. Whenever an adapter service needs a connection, the Integration Server provides a connection from the pool. If no connections are available in the pool, and the maximum pool size has not been reached, the server creates one or more new connections (according to the number specified in the Pool Increment Size field) and adds them to the connection pool.

If the pool is full (as specified in Maximum Pool Size field), the requesting service will wait for the Integration Server to obtain a connection, up to the length of time specified in the Block Timeout field, until a connection becomes available. Periodically, the Integration Server inspects the pool and removes inactive connections that have exceeded the expiration period that you specified in the Expire Timeout field.

Minimum Pool Size - If connection pooling is enabled, this field specifies the number ofconnections to create when the connection is enabled. The adapter will keep open the number of connections you configure here regardless of whether these connections become idle.

Assigning 0 will create no instance of the connection. By default the Minimum Pool Size is set to 1 which creates a single instance of the connection on the initialization of the connection pool.

Maximum Pool Size of 10 means that a maximum of 10 connections can be existing in theconnection pool at a single time.

**In select template (in JDBC Adapter service) if you mention MAX=0 what happens? If you mention MAX=10 what happens?**

Maximum Row – This field is used to specify the maximum number of records to retrieve from thedatabase. The default value of 0 (no limit) retrieves all records.

A value of 10 will return a maximum of 10 records from the database.

What is the purpose of using context factory?

A context factory is a way to communicate various preferences and properties that define the environment in which naming and directory services (JNDI) are accessed.

For example, a context might require specification of security credentials in order to access the service. Another context might require that server configuration information be supplied.

**What are the configuration parameters of JMS adapter?**

Connection Type, Connection name, package name, folder name, Transaction type, JNDI Initial Context Factory, JNDI provider URL, QueueConnectionFactory JNDI Name, user name, password

**What is mentioned in JNDI Initial context factory?**

The fully qualified class name for the Initial Context Factory used by your JNDI provider. For example: weblogic.jndi.WLInitialContextFactory

For SAP Web Application Server 6.30, specify com.sap.engine.services.jndi.InitialContextFactoryImpl

**What is the JNDI provider URL?**

The Java Naming and Directory Interface (JNDI) is a Java API for a directory service thatallows Java software clients to discover and look up data and objects via a name. The Directory Service implementations may make use of a server, a flat file, or a database.

The URL to the starting context within the JNDI directory service. For example: t3://MyHost.MyCompany.com:7001

For SAP Web Application Server 6.30, specify

SAP\_Netweaver\_Hostname:SAP\_J2EE\_Engine\_port\_number

By default, the SAP\_J2EE\_Engine\_port\_number is 50004

**Syntax of Insert Statement in Oracle**

INSERT\_INTO {table\_reference | [THE] (subquery1)} [(column\_name [, column\_name]...)]{VALUES (sql\_expression [, sql\_expression]...) | subquery2} [RETURNING] [row\_expression [, row\_expression]... INTO {variable\_name| :host\_variable\_name} [, {variable\_name | :host\_variable\_name}]...];

INSERT INTO bonus SELECT ename, job, sal, comm FROM emp

WHERE comm>sal \* 0.25;

INSERT INTO emp (empno, ename, job, sal, comm, deptno)

VALUES (4160, 'STURDEVIN', 'SECURITY GUARD', 2045, NULL, 30);

INSERT INTO dept

VALUES (my\_deptno, UPPER(my\_dname), 'CHICAGO');

What is the difference between DB2 and oracle?

Oracle directly supports a Java application server where you can deploy Java servlets while with DB2 you would need to employ a Java translator like Tomcat

**MQ Adapter configuration Settings**

- Connection Type, package, Queue manager, host, TCP/IP port, server connection channel, CCSID, user, password, queue name.

- MQ adapter templates: put, get, peek, request /response.

**What is difference between Webspehere MQ Connection and Webspehere MQ transaction Connection?**

In MQ connection all Transactions are considered to be one whereas in MQ Transaction all transactions are considered separate.

**Explain trading networks?**

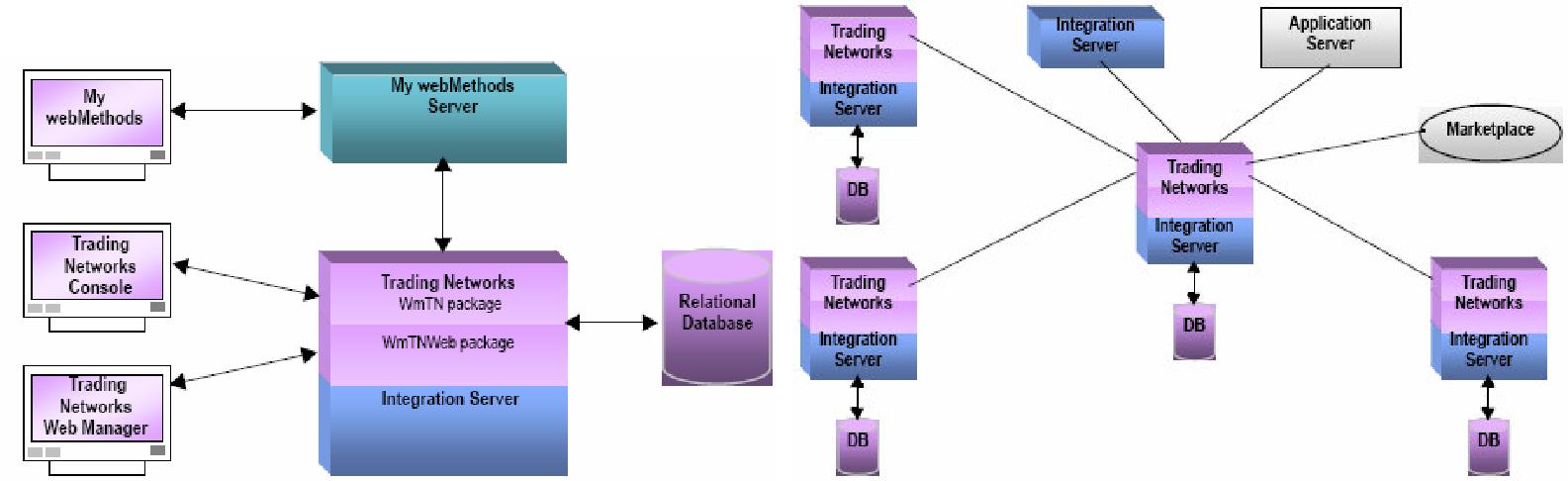
A trading network is a set of organizations that have agreed to exchange the business documents. A webMethods Trading Networks is a component that runs on webMethods Integration Server. Trading Networks enables your enterprise with other companies and market places to form a business-to-business trading network. The organizations in your network are referred to as trading partners(partners). A trading partner can be any system, within or outside your enterprise, that produces or consumes business documents.

TN is a business document gateway that can that can recognize and process a variety of documents.

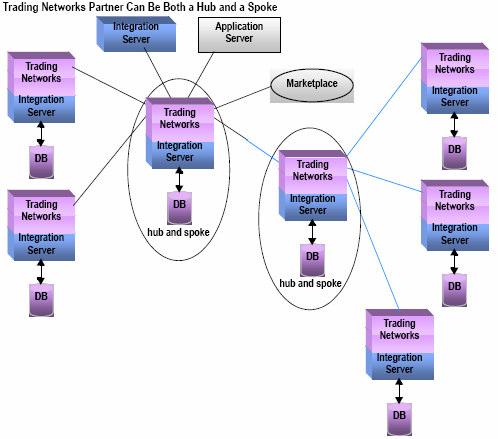
The business documents maybe purchase order, order status, acknowledgements etc as well as business domain specific documents.

Components are server – My webMethods server – TN console – TN web manager. WmTN and WmTnWeb are the packages on IS. All information pertaining to TN is persisted on a RDBMS.

Trading Networks does not require that all partners in the network use webMethods Trading Networks.



The partner in the center is referred to as the hub of the network. The other partners are referred to as spokes. The hub hosts the network and the spokes participate by interacting with the hub.



**Explain about TN console?**

TN console is a Java based GUI, is a Trading Networks user interface primarily for designing how you want Trading Networks to recognize documents using TN document types and how Trading Networks processes documents.

It is also used to perform deprecated functions which have been migrated to MWS like transaction analysis – tasks and activity log viewing - profile management.

TN Web Manager is a web based UI of Trading Networks. It offers some functions of TN console and other administrative actions. It is now deprecated.

**What are different icons we see on TN console?**

Some icons in the Selector Panel are:

* Document Types
* Processing Rules
* Agreements
* Trading Partners [Deprecated]
* Enterprise [Deprecated]
* Transaction Analysis [Deprecated]
* Tasks [Deprecated]
* Activity Log [Deprecated]

**What is a TN Document Type?**

TN document types specify the types of business documents that you want to exchange with your partners. It maybe cXML (commerce –XML standards created by Ariba), xCBL (xml Common business library), OAG (Open Applications Group), and EDI, or any customized specifications. Pre processing actions may be specified too.

They include –

* Identification information
* Extraction information
* Pre-processing options

**How will you create custom attributes while creating a document type in TN?**

Document attributes identify selected content from the documents that pass through yourtrading network. Trading Networks maintains two types of attributes— system attributes (SenderID – ReceiverID – DocumentID - User Status – GroupID – ConversationIDetc) and customattributes

Document Attributes are referenced in TN document types indicating that you want that piece of information extracted from documents that match the TN document types [Extraction information]

Custom attributes are attributes that you define to identify any other content that you are interested in extracting from documents. For example, to extract the purchase order number from documents, you might define a document attribute named – PO\_Number

To create custom attributes - In Trading Networks Console: View > Document Types. Select Types > Document Attributes. We can create a new attribute by defining the Name, Type, Description. This can be enabled or disabled.

**How do you create XML Document Type in TN?**

TN XML document types define how Trading Networks recognizes XML documents, where to locate attributes within an XML document, and how to pre process the XML documents.

We can create one by defining

* Identification information
* Extraction information
* Pre-processing options and other options like IS document type, IS Schema etc

**How to install EDI DocumentType in TN?**

EDI Module provides the TN document types for EDI documents, and you can install them from the EDI Module home page. You only need to install the TN document types for the types of EDI documents you plan to exchange.

**What is DUNS number?**

Data Universal Numbering System, abbreviated as DUNS. It is a unique numeric identifier (9 digits) associated with single business entity. It is a system developed and regulated by Dun & Bradstreet (D&B). The DUNS database has over 57 million entries.

**What is a processing rule? How would you configure the routing rules?**

Processing rules indicate how you want Trading Networks to process documents as they traverse your trading network.

* Sender ID, Receiver ID, Document type along with other details are specified
* Pre processing actions are specified (digital signature verification, valid structure etc). The pre processing actions in the processing rule to override the actions that are specified in the TN document type
* Processing Rule Actions namely Execute a service – Alert email – Change user status – Deliver document by

**What is default processing rule? Which processing rule executes first when more than one processing rule matches the document?**

If a document matches more than one processing rule, Trading Networks uses the first Processing rule it encounters.

Hence, the order in which you maintain your processing rule is important because Trading Networks checks for a matching processing rule in that order. Keep rules with specific criteria before rules with general criteria. You should also set up a default processing rule

that you want Trading Networks to use when a Document does not match any of the other processing rules. Place the default-processing rule last in the list.

**Can we have two processing rules with same sender and same document type?**

Yes we can, hence we should order the processing rules based on their specificity.

**How many types of delivery methods are there in TN to send a document?**

Immediate Delivery

Scheduled Delivery

Queue for polling

Receiver Preferred Protocol

**What are the transport types in Routing rules?**

Immediate Delivery or Scheduled Delivery of documents is performed using the following transport types –

Primary/Secondary E-mail

Primary/Secondary FTP

Primary/Secondary FTPS

Primary/Secondary HTTP

Primary/Secondary HTTPS

**What is a Trading Partner Profile. What are the different elements in creating Partner Profiles?**

Trading Networks maintains your profile (called the Enterprise profile)as well as of the partners in your TN

TP profile contains the contact information & other information (host details, certificates, etc) of the Trading Partners that Trading Networks needs to exchange documents with your partners.

Profiles need to be activated to enable document exchange.

**What are extended fields in TN? What is meant by extended criteria?**

Fields within a partner profile. Extended fields are created to maintain information about trading partners that is not covered by the standard fields.

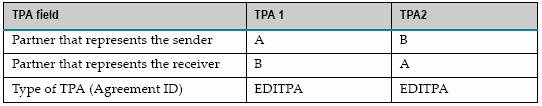
**What is TPA? Does TN use TPA every time when a document coming in to the TN?**

Trading partner agreements for pairs of partners contains specific information for two trading partners, where one partner represents a sender and the other represents the receiver.

Each TPA must have a unique combination of the following:

* Partner that represents the sender
* Partner that represents the receiver
* Type of TPA, represented by the “Agreement ID”

Multiple TPAs for a pair of Trading Partners can exist



It is optional to create TPA. Trading Networks does not use TPAs for its own processing. For example, Trading Networks does not use TPAs when determining the processing rules to use for a document. webMethods EDI Module use TPAs to perform processing. We can create applications that use stored TPA information at run time to tailor how documents are exchanged.

TPA vs. Profile information – TPA is transaction specific and can vary for specific kinds of document exchange (digital signature etc) whereas a profile contains information about the partner that does not vary with each document being exchanged

Primary goal of the TPA function in Trading Networks is to offer users a flexible and efficient way to define these transaction specific parameters.

**What is the first parameter for TPA?**

* The partner that represents the sender for the TPA.
* The partner that represents the receiver for the TPA.
* An agreement ID to identify the type of TPA (webMethods EDI Module agreement ID : EDITPA)

**What are the types of status in TPA? What is the difference between proposed status and agreed status?**

TPAs have two types of statuses

— Agreement status

— Data status

Agreement status indicates the status of the TPA agreement between the receiver and sender.

* Proposed : TPA is in Draft stage and modifications are done
* Agreed : TPA is final. Data statuses take effect. TPA cannot be deleted
* Disabled : TPA is inactive and will not be used

Data status determines whether you can modify the TPA data (can change the variables in the IS document type)

* Modifiable : TPA data can be changes
* Non-modifiable : Cannot change the TPA data

**How can we import TPA’s from one environment to another?**

We can export the Trading Network objects like Processing Rules, TPAs, TN Document Types, Document Attributes, Partner Profiles etc from one environment and import them to another. File Export / Import

**What is Transaction Analysis?**

Transactions are the documents that have passed through Trading Networks. We can view the information of these transactions using the Transaction Analysis in TN console or Monitoring > Integration > B2B >Transactions in MWS.

The information pertaining to the documents is stored in the database. We can see

* Attributes that have been extracted from documents
* Content of documents
* Status of documents that Trading Networks is in the process of delivering

In addition to viewing this information, we can also resubmit and reprocess documents.

Resubmit - TN sends the document back to recognition processing as a new document. The TNdetermines the TN document type for the document, extracts the document attributes, selects a processing rule, and processes the document.

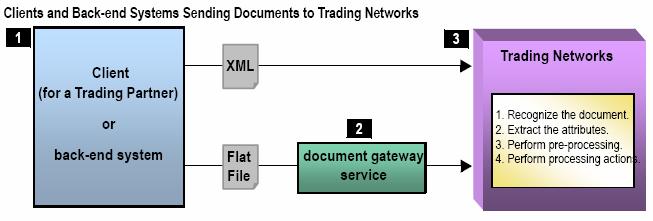
Reprocess - Trading Networks sends the document back to processing rule selection. The TN uses the TN document type it already has saved for the document as well as the document attributes it already has saved for the document. It simply selects a new processing rule and processes the document again.

**What is conversation ID?**

A conversation is a case in a Business Process that involves a series of related documents being exchanged by two or more trading partners. All documents from a specific trading partner contain the same Conversation ID.

A conversation ID is a system attribute that identifies a value within a document that is common to all documents that are part of the same business process.

**How do you send a document from developer to TN? What is the wm.Tn.receive service output?**



For XML documents, the client should invoke the wm.tn:receive service. For flat files, the client should invoke a document gateway service you created, which in turn, invokes the wm.tn:receive service.

Document gateway services are the entry points for flat files into Trading Networks. That is, rather than sending flat files directly to Trading Networks, your trading partners send their flat files to a document gateway service.

It provides hints to Trading Networks to indicate the TN flat file document type to use for the flat file document. The service provides these hints in the TN\_parms variable, which is located at the root of the pipeline.

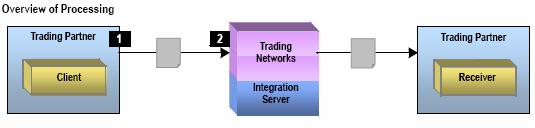
The output of the wm.tn:receive is a business document - bizdoc - The document that Trading Networks received (the document passed in the node input variable) formatted as an IS document (IData object).

**What is an Inbound and outbound document?**

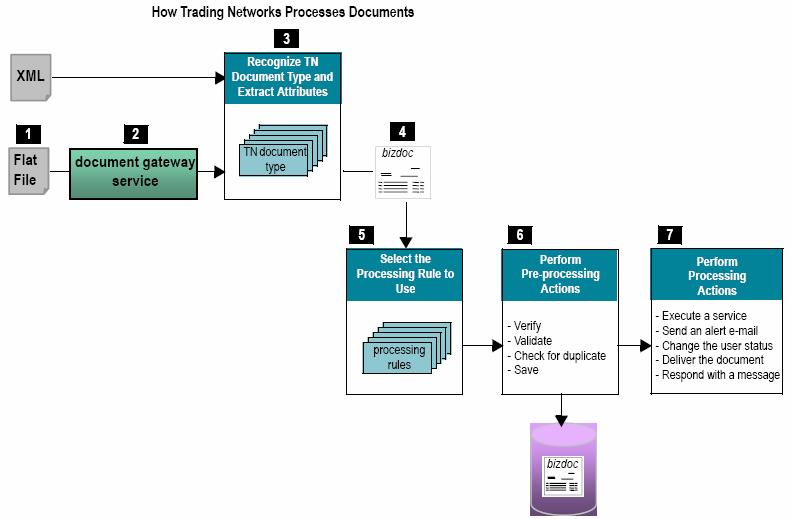
Inbound – Partners to Trading Network (TN Server in IS)

Outbound - Trading Network (TN Server in IS)to Partners

**How do you process the incoming document in Trading Networks?**



The run-time processing of the documents is done by defining Trading Networks objects at design time.



3>Trading Networks performs recognition processing. In recognition processing, TradingNetworks recognizes the type of document using TN document types that you set up.

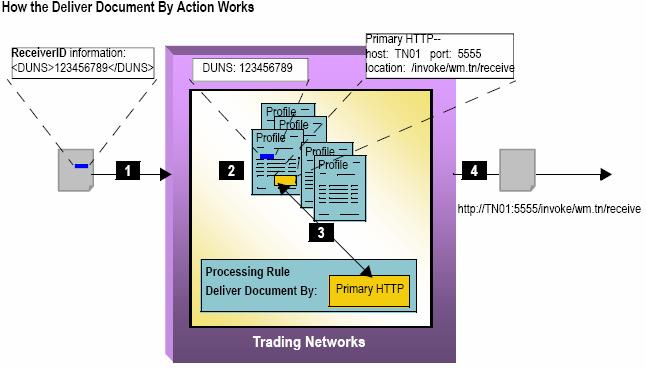
4>The output of the Trading Networks recognition processing is a BizDocEnvelope in thepipeline in the bizdoc variable.

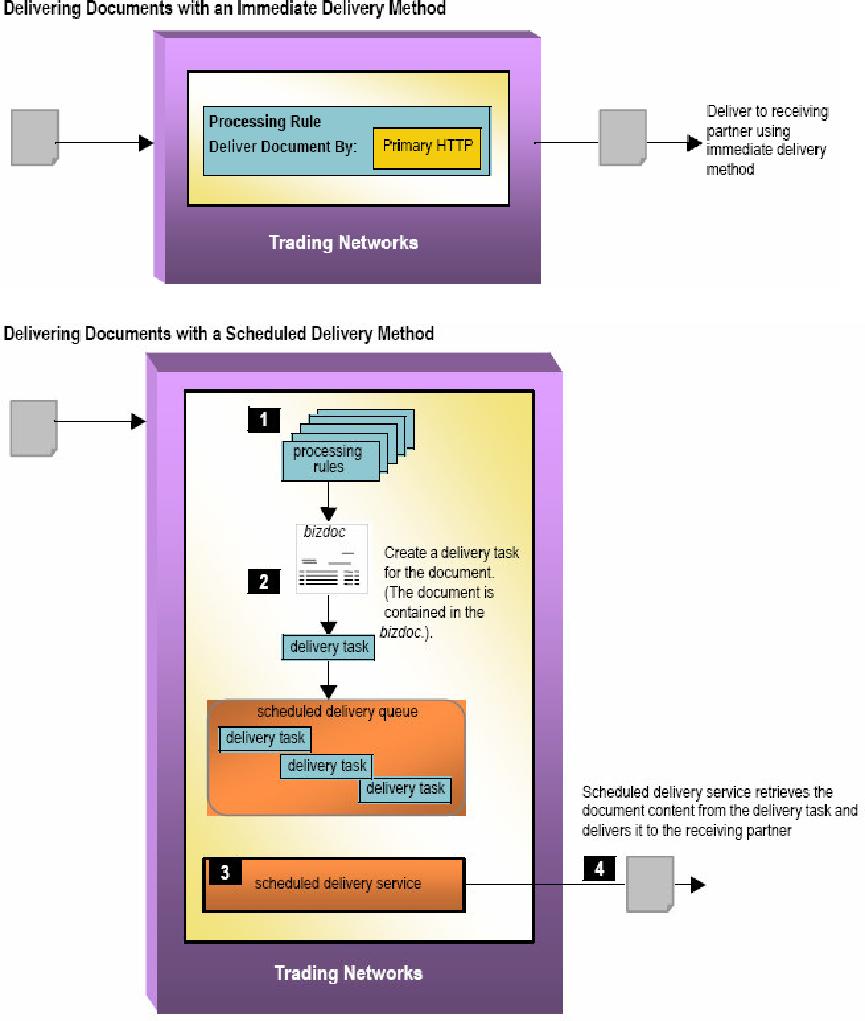
**How do you process the outbound document in Trading Networks?**

When a processing rule includes the Deliver Document By processing action, Trading Networks determines the delivery method to use to deliver the document to the receiving partner.

The Delivery methods can be set to anyone of the following:

* Immediate Delivery
* Scheduled Delivery
* Queue for polling
* Receiver Preferred Protocol





There are two kinds of scheduled delivery queues –

Public - to schedule the delivery of documents that are aimed at multiple different receivingpartners.

Private - to schedule the delivery of documents that are aimed at one specific tradingpartner. You define a private queue in the profile of the partner to receive the documents.

**What is the difference between immediate delivery and scheduled delivery?**

Immediate delivery - Trading Networks attempts to immediately deliver a document directly tothe receiving partner.

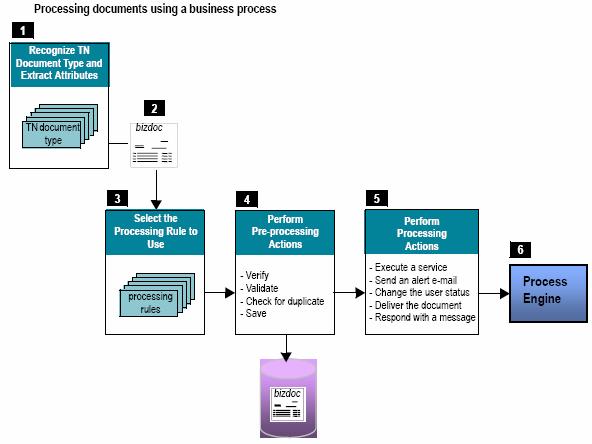
Scheduled delivery is a way to batch multiple documents that are acted on (delivered) atscheduled times. It is more efficient to deliver a batch of documents at a time rather than deliver them immediately as they arrive.

**If a document is sent to trading networks, then can we use it in modeler?**

Yes this can be done.

If Trading Networks was instructed to extract the Conversation ID system attribute and it has a value, Trading Networks passes the document to the Process Engine.

For a document to be used in a business process (Modeler), the document must be sent to the Process Engine.



**What is the difference between routeBizdoc and submit service?**

wm.tn.route:routeBizdoc- Submits for processing a document (submits a bizdoc -BizDocEnvelope) that has already been recognized by Trading Networks. This service does not check the identity of the sender against the currently logged in user.

Only invoke this service from within processing rules or services - do not expose directly to trading partners.

wm.tn:submit- Submits a document that has already been recognized to Trading Networks forprocessing. This service ensures that the sender of the document matches the current user.

**What is EDI?**

EDI stands for Electronic Data Interchange. It is computer to computer exchange of business documents between companies in specific computer record formats based on widely accepted standards. The standards are flexible to fit to the needs of a Business.

Over 80,000 companies have made the switch to EDI to improve their efficiencies. Companies have EDI implementation guide where they modify the EDI standards for their needs. There is cost reduction by automating B2B processes with EDI.

Drawbacks - EDI adapter software is too expensive and needs to be changed with changes in standards. VAN subscription costs. EDI is static so there has to be interaction between partners for standardized Business Process.

**What is EDIINT? What are the different protocols supported for transferring the EDI documents?**

EDIINT – EDI over the Internet enables the transport of EDI data over the internet in a securemanner, between EDI Trading partners.

The means of transferring over the internet are AS1 (Applicability Statement 1) – SMTP (asynchronous), AS2 – HTTP/HTTPS (synchronous – peer to peer) and AS3 – FTP (synchronous – client/server)

It is an alternative data transport to value added network (VAN) based data communications.

**What is the difference between EDI and EDIINT?**

EDIINT is EDI transfer through Internet and EDI is a standard used to exchange documents

**What is VAN?**

A VAN is an intermediary third party network - Value Added Network. A VAN is responsible for the secure and reliable communication of EDI documents. VAN is a like an electronic post office that interconnects with another VAN. The VANs make sure that EDI transactions are sent and received. The VAN is responsible for routing, storing and delivering EDI messages. They also provide delivery reports and other services.

**Explain VAN-Bridge functionality**

**What are the different EDI standards?**

EDI – Standards made by ANSI American National Standards Institute

EDI ASCX12 – EDI standards made by The Accredited Standards Committee (ASC) X12

EDI FACT - United Nations/Electronic Data Interchange For Administration, Commerce, andTransport (UN/EDIFACT). It is widely used in Europe.

EIDX (Electronics Industry Data Exchange Association)

EDIFICE (EDI for Companies Interested in Computing and Electronics). It is European EIDX

XML-EDI – is collection of 5 Core technologies namely XML – EDI – Templates – Agents -Repository

**What are the different types of EDI documents?**

Business documents like invoices, purchase orders, and advanced ship notices can all be exchanged between companies through EDI.

850 purchase order

856 shipment notice

810 invoices

There are EDI standards for the chemical industry, pharmaceutical industry, consumer products industry, retail industry and many more.

**What are the segments in an EDI document?**

6 segments - ISA, IEA, GS, GE, ST, SE

**What is the EDI / EDI X12 Structure?**

EDI documents themselves made up of sections. Not necessary that there has

data segments made up of Heading, Detail & Summary to be all the 3 sections.

HEADER SECTION (One Occurrence)

SEGMENTS relate to ENTIRE MESSAGE

Codes, Addresses, Contracts, References

DETAIL SECTION (Multiple Occurrences)

SEGMENTS relate to DETAILS where it is written Each Line Item, Schedules, Quantities

SUMMARY SECTION (Control Totals)

SEGMENTS with TOTALS or CONTROL data

Interchange Segment Header (ISA) – SENDERID RECEIVER ID & other information Messages – made of Data Segments – made of Data Elements

Interchange Segment Trailer (IEA)

ISA : 16 elements(1,2 - Authentication, 3&4 - Security, 5&7 - Qualifier ID, 6 & 8 - Sender andReceiver ID, 9&10 - date and time, 11 - usage indicator, 12 - version, 13 - control number, 14 - acknowledgement, 15 - replicator, 16 – termination)

IEA :01 - number of function groups, 02 - control number[=ISA 13].

GS :8 elements (01 -transaction set name [ex: PO], 06 control number)

GE :2 elements (01 - number of transaction sets, 02 - control number[=GS06]) ST : 2 elements (01 - transaction set code, 02 - control number)

SE :2 elements (01 - number of elements in ST and SE. 02 control number[=ST02])

**How do you recognize an EDI document?**

Segments, Elements, Segment Terminators (~) and Element Terminators (\*)

Functional Groups are a group of related (department wise) Transaction Sets.

Functional Groups – enclosed between GS GE segments. Indicated by PO (Purchase Order), IN (Invoice) etc

GS is start of functional group header segment. GS – followed by PO/IN etc & control number of the GS segment 00001, 00002 etc & other information

GE is start of functional group - trailer segment. GE – followed by count number of the transaction sets enclosed in the groups and the Control number of the GS segment 00001, 00002 etc

Transaction sets are basically the whole EDI documents themselves which belong to a particular Function group 850 PO or 810 IN

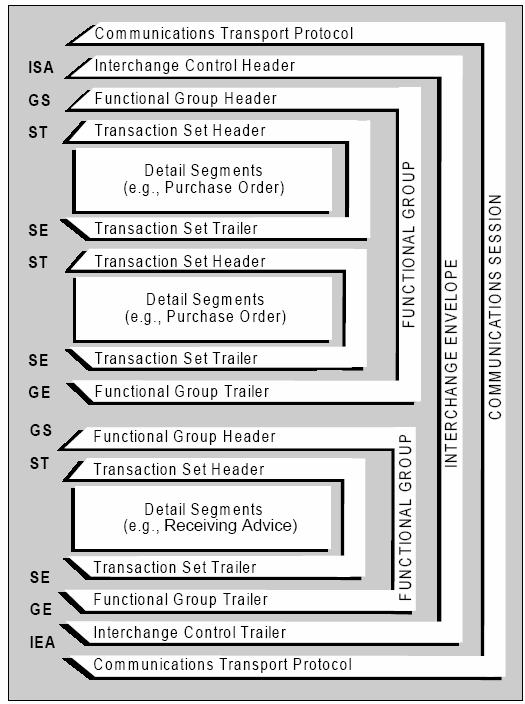
Transaction Sets – are enclosed between ST header and SE trailer segments between a GS and GEFunctional Group.

Segment common elements can be helpful in recognizing the EDIFACT:

NAD – Name – Name and address

LIN – Item Id – Line item number, part number RFF – Reference – Quote number, release number DTM – Date and Time

QTY – Quantity



**Difference between EDI ASCX12 Vs EDIFACT**

Different Envelope nomenclature

ISA

GS

ST

SE

GE

IEA

UNB

UNG

UNH

UNT

UNE

UNZ

**What are different transaction sets in EDI?**

850 – Purchase order, 856 – Advanced Shipment Notice, 810 – Invoice, 997 – Functional Acknowledgement

**A most common EDI cycle?**

* Customer transmits EDI 850 (purchase order)
* Supplier transmits EDI 997 (functional acknowledgement)
* Supplier transmits EDI 856 (advance ship notice)
* Customer transmits EDI 997 (functional acknowledgement)
* Supplier transmits EDI 810 (electronic invoice)
* Customer transmits EDI 997 (functional acknowledgement)
* Customer transmits EFT (Electronic Funds Transfer) Payment

**What is the last element in ISA segment?**

ISA segment has 16 elements. The last is ISA16 - Sub element Separator

**What are EDI loops?**

When Data segments have to be read together to obtain meaningful information – we use EDI Loops. EDI Loops group certain dependent segments using the Segment ID.

**How are you going to identify an 850 document by seeing only the content?**

Transaction Set Code 01 - ST 01

The Beginning segment is BEG (PO Header Data)

**Explain about 850 transaction segments?**

P01 – Base item Data

CTP – Pricing information

PID – Product item description

SAC – Service promotion allowance or charge information

**What is 854 EDI transaction?**

854 Shipment Delivery Discrepancy Information

**Which segments will be used in 997?**

AK1, AK2, AK3, AK4, AK5, AK9

FA can be generated at Group level, Transaction Set level, Segment level and Element level

**What is an EDI adapter?**

It is nothing but the EDI Module which on installed creates 4 packages on the IS namely – wmEDI, wMEDIforTN, wMEDIINT, wMEDIsample

It provides services and transformation management tools to do the below functions in real time or in batch.

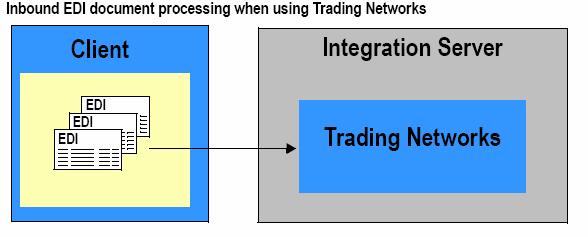
Process most EDI standards - Parse, convert, format, and validate EDI documents - Process EDI documents containing multiple interchanges/groups/transactions with multiple versions - Generate functional acknowledgements (FAs) - Create envelopes for EDI documents - Transport EDI documents using the FTP, HTTP, and HTTPS protocols

Maintains information of EDI TP – Send and Receive EDI – Connect to VAN for delivery pickup – Process through processing rules – Batch sending EDI – Generate FA for inbound EDI - reconciliation

**How an EDI document will be processed in Trading Network? Explain the entire scenario of handling when an EDI 850 comes into your TN? If multiple Transactions are there in EDI how do you handle it?**

* Validates the interchange envelope(s)
* Parses the document into its constituent elements: the interchange envelope(s), the group envelope(s), and the individual transaction set(s)
* Resubmits the individual elements to Trading Networks (wm.tn:receive)
* Checks the Trading Networks configuration files to determine whether and how to use the Trading Networks large document handling feature.
* Assigns document, group, and conversation IDs to each EDI element
* Performs business document exchange on the EDI element (interchange envelope, group envelope, or transaction set), including:
  1. Determining the processing rule to use to process each element
  2. Performing the pre-processing actions defined in the processing rule
  3. Performing the processing actions defined in the processing rule
* Performs conversation management on the EDI element, if applicable

For inbound processing, you create clients that send EDI documents to Trading Networks and you set up information in Trading Networks to process the document.

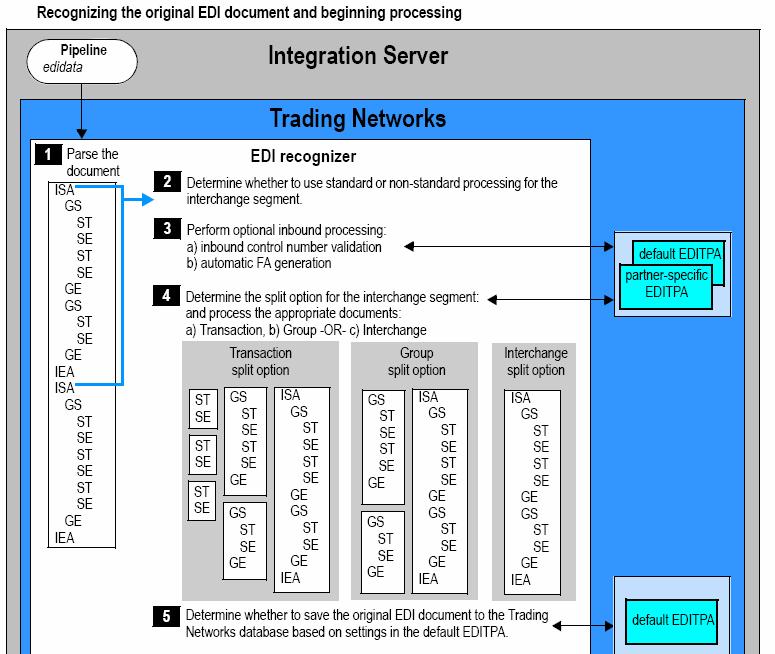


Clients send documents to the Integration Server to be processed through Trading Networks using FTP, HTTP, HTTPS, File Polling, EDIINT AS1, EDIINT AS2, ENIINT AS3

In the IS a EDI Content Handler takes over the processing.

Initial EDI Processing in the TN

WmEDIforTN package has an EDI recognizer



Standard processing - EDI Module uses all standard Trading Networks objects for processing.EDI Recognizer obtains profiles of sender and receiver from the ISA header and it retrieves the respective EDITPA.

All documents (Transaction, Group and Interchange) are processed using the settings defined for the Sender and Receiver in Interchange Header.

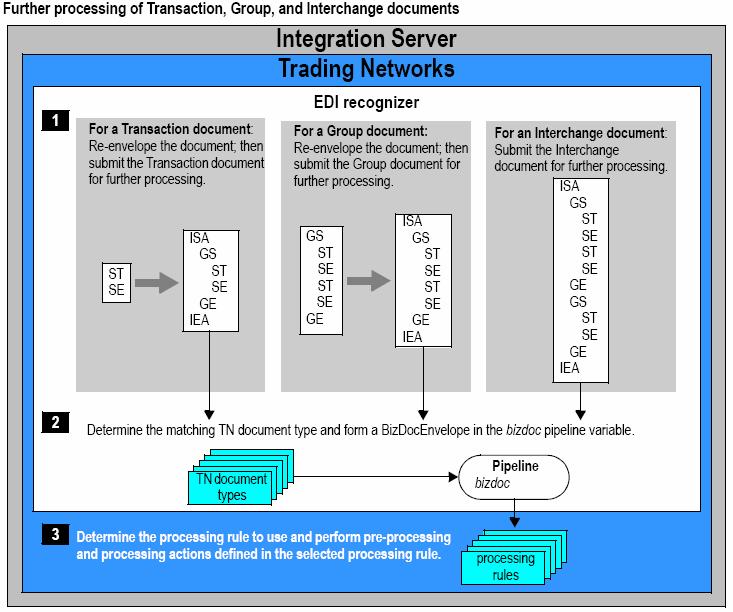
Non Standard processing - Custom interchange sender/receiver pair information is defined inWmEDIforTN package home page. Based on this existence i.e. if the sender/receiver

pair information exists for the sender/receiver of the interchange segment, the EDI Module performs non-standard processing.

EDI Recognizer uses the interchange sender/receiver pair information to determine the EDI ID qualifiers to use with the sender/receiver identified in the group header and it retrieves the respective EDITPA.

Transaction and Group documents are processed using the settings you defined for the sender/receiver specified in the group header.

If the original EDI document contains multiple interchanges, after splitting documents and sending them for processing, the EDI recognizer can save the entire original EDI document to the Trading Networks database based on a setting that you define in the default EDITPA.



Pre-Processing Actions defined in TN Document Types

* Validate Structure: against flat file schema. Default – no validation
* Check for Duplicate Document: uniqueness check. Default – no validation
* Save Document to Database: Default – yes

Processing Actions

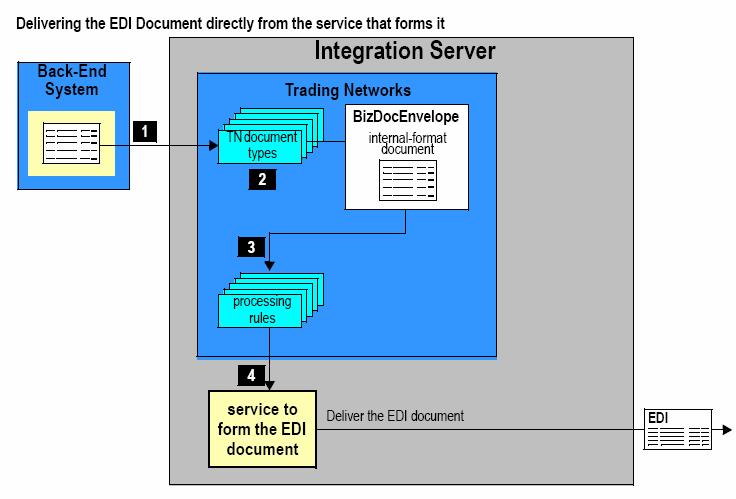
Typically the Execute a Service action to process the inbound document

EDI Trading Partner Agreements (EDITPAs)

EDITPA is a set of variables that you specify to tailor how the EDI Module processes documents that are exchanged between two trading partners.

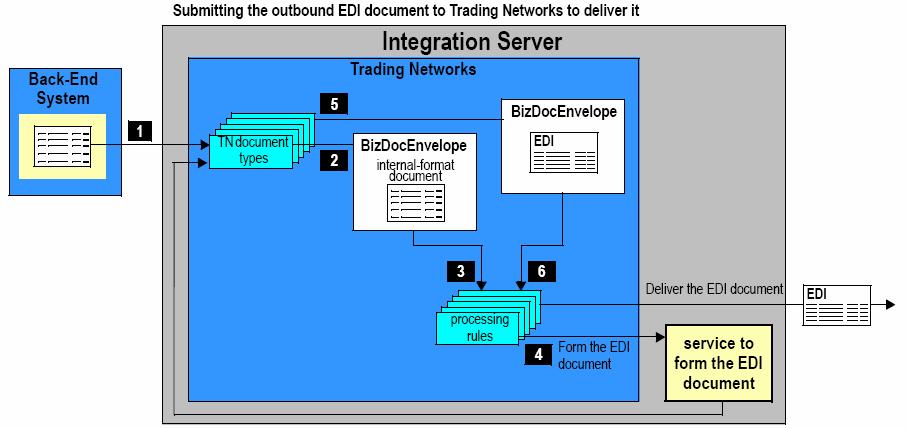
* partner-specific EDITPA
* default EDITPA

Outbound TN Processing



1>Back-end system or client sends an internal-format document to the Integration Serverinvoking the wm.tn:receive service to send the document to Trading Networks.

3> Processing Rule - Execute a Service processing action



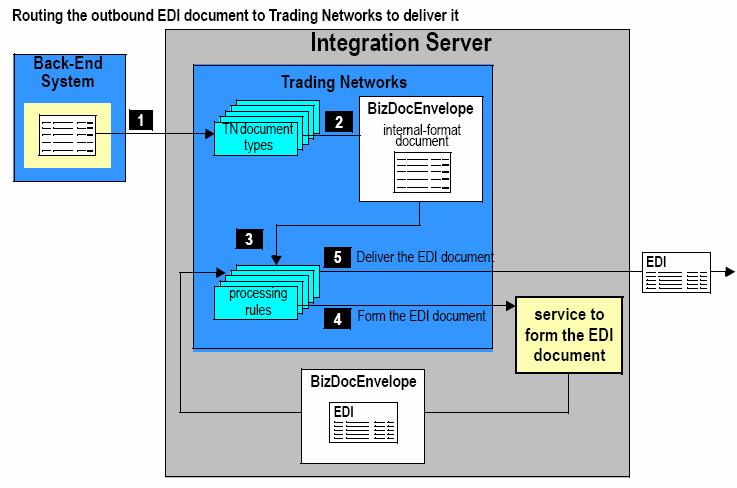
4>After forming the EDI document, service invokes thewm.tn.doc.xml:routeXmlservice to

submit the outbound EDI document back into Trading Networks document recognition.

5>EDI recognizer obtains EDITPA information and splits the document based on the split option(Group or Interchange) in the EDITPA.

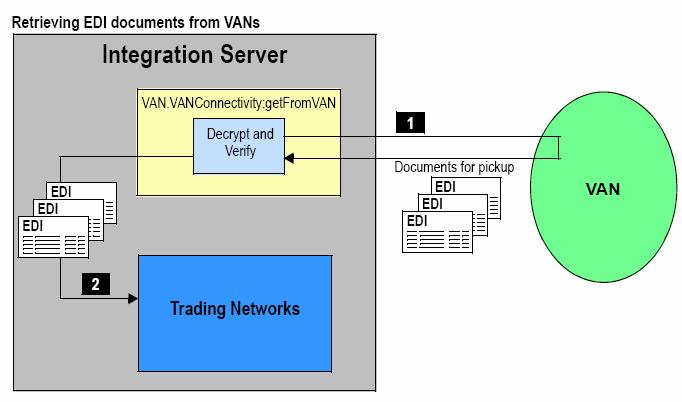
6>Processing rule - Execute a Service (Custom Service for EDI Delivery) or Deliver Document

By (Send document to VAN or Batch processing of EDI)



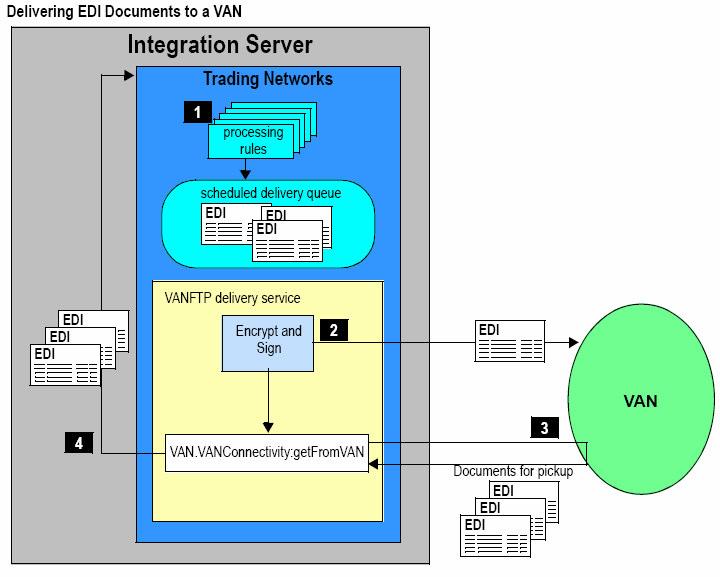
VAN Pickup & Delivery

Inbound Processing - EDI Module provides the VAN.VANConnectivity:getFromVAN service to retrieve documents waiting on a VAN



Outbound Processing - VAN.VANConnectivity:putToVAN service to deliver outbound EDI documents

to a VAN



1>processing rule uses the Deliver Document By processing action to deliver a document to ascheduled delivery queue associated with the VANFTP delivery service.

2>When the schedule associated with the delivery queues arrives – EDI documents are deliveredto the VAN by the VANFTP delivery service

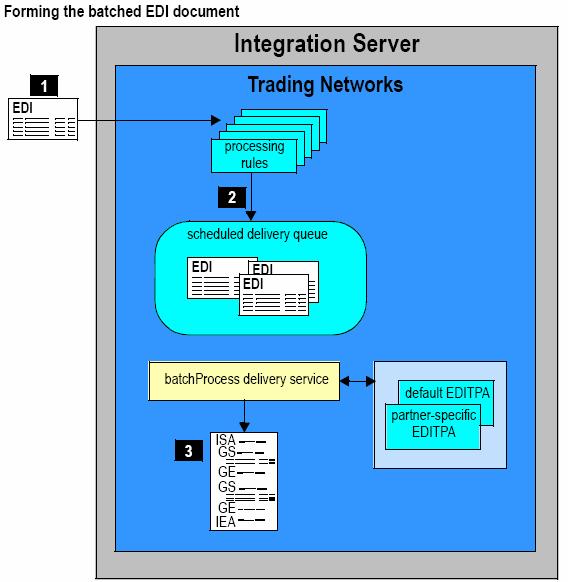
3>Optionally, the VANFTP delivery service can invoke the built-in serviceVAN.VANConnectivity:getFromVAN service, which the EDI Module provides, to retrieve any documents that might be waiting on the VAN for pick up.

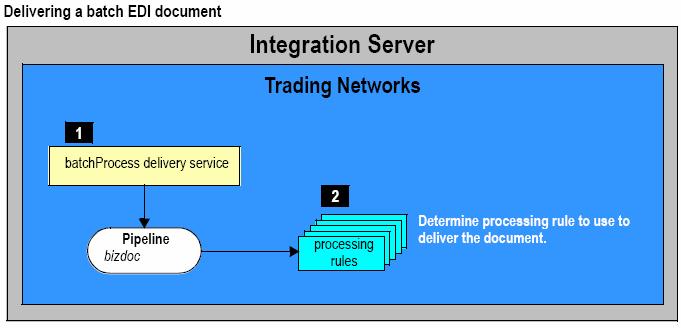
Batching of EDI documents can be done. Batching combines EDI documents into a single document and adds group level and interchange level headers and trailers

Batching is more efficient as

* Documents can be grouped and sent at scheduled times
* Increases system performance eliminates frequent communication connections and individual envelope authentication
* Adapts well to legacy systems as they are batch oriented too.

For Batching - wm.b2b.editn.batch:batchProcess present in the EDI Module. To use this service public scheduled delivery queues should be defined in TN and a schedule specified for thebatchProcessservice to act on the documents in the queue.





**If ten trading partners are sending EDI-850 to your enterprise how would you differentiate among them?**

EDI Recognizer obtains profiles of sender and receiver from the ISA header and it retrieves

the respective EDITPA

**Where can we set the properties of large EDI document? What is the default size of the Large EDI document handling? What are the parameters to configure the large EDI document handling?**

We can define a threshold size at which the EDI Module considers a document large and these large documents are handled/processed differently (stored in local hard disk space – tspace instead of memory)

wm.b2b.edi:convertToValues– by using the iterator variable convert documents to IData objectsiteratively to process the large document a segment/section at a time.

http://IntegrationServer:port/WmEDI – Configuration - Configure Properties EDIBigDocThreshold= n

n – is n is a negative whole number EDI Module processes all documents in traditional way Default value -1 (No document is considered large)

0 - EDI Module consider all documents large

n – positive whole number. EDI Module considers all documents over n bytes to be large EDIBigDocThreshold=1000000

Trading Networks large document settings:

From the Server Administrator, select Settings > Extended from the menu

watt.server.tspace.location

watt.server.tspace.max

Administration > Integration > B2B Settings > Server Settings

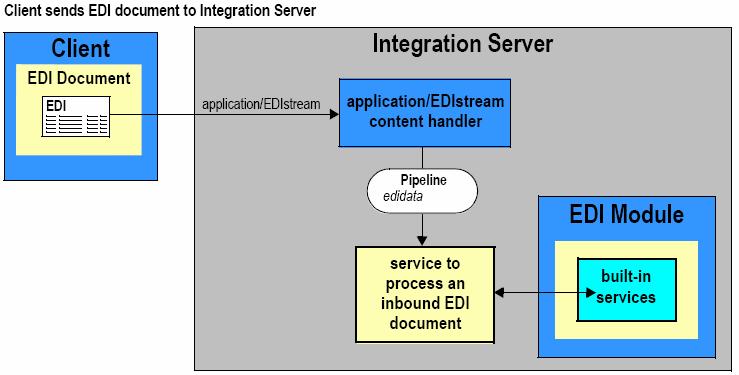
tn.BigDocThreshold (-n +n 0 -1)

tn.xml.xqlThreshold - The number of bytes that Trading Networks reads into memory to performthe XQL queries in a TN XML document type (+ whole number, default is 1000 bytes)

**How do you process an inbound and outbound EDI document? What are the services to process Inbound & Outbound EDI documents? What is the process of parsing an EDI document in developer?**

Inbound – Trading Partner to a System

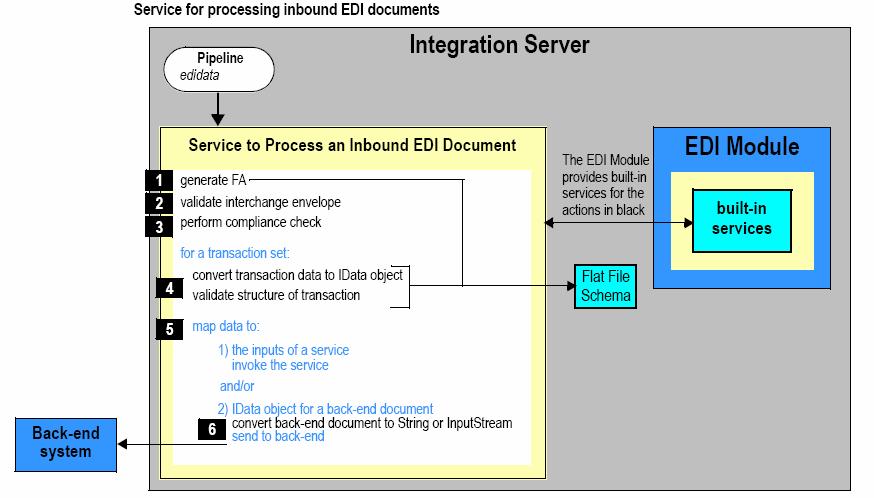
Clients send documents to the IS using HTTP/HTTPS, FTP or file polling.



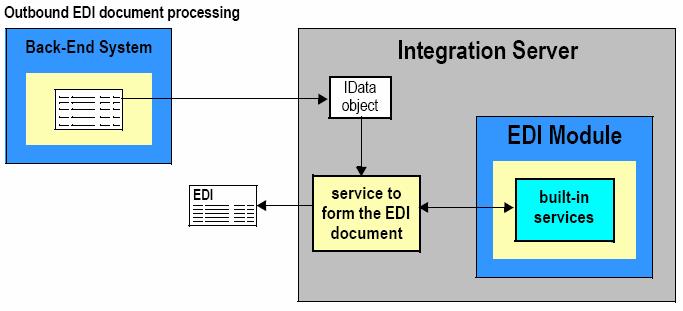
EDI content handler creates the variable edidata in the pipeline.

Inbound EDI processing services Map Data

* FROM Standard Segments & Elements
* TO Internal Proprietary Records & Data Fields

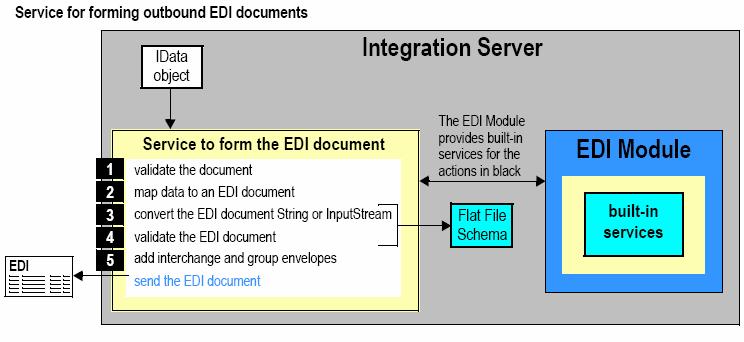


|  |  |  |
| --- | --- | --- |
| Majority of the | | effort will most likely be in processing the individual transaction sets |
| within the | EDI document. | |
| Outbound – | System | to a Trading Partner |
| The EDI documents | | created should be sent to the Trading partners |



Outbound EDI processing services Map Data

* FROM Internal Proprietary Records & Data Fields
* TO Standard Segments & Elements



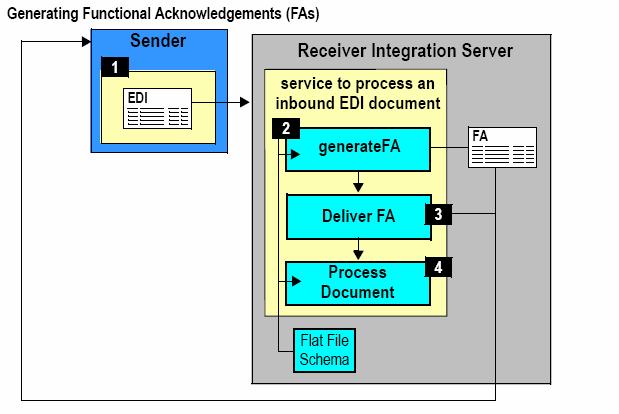
**What are the various services for processing an EDI file?**

Some of them are –

wm.tn:receivewm.tn.doc.xml:routeXmlVAN.VANConnectivity:getFromVANVAN.VANConnectivity:putToVAN wm.b2b.editn.batch:batchProcess wm.b2b.edi:convertToValues pub.util.getEDIString

**Can we create/send functional acknowledgments automatically? How?**

FA is a type of EDI transaction set (997) that acknowledges the receipt, as well as the structural and syntactical validity of an EDI document.



The built in service used is wm.b2b.edi.util:generateFA

**Where is the Envelope process service located?**

An inbound EDI is validated using this process. wm.b2b.edi.envelopeProcessin WmEDI package

**What is Bizdoc and how do you get the content from the Bizdoc?**

The name of the variable in the pipeline that contains the BizDocEnvelope. A BizDocEnvelope represents a routable Trading Networks transaction. It contains a document that Trading Networks is processing and includes additional information that Trading Networks requires for routing and processing the document. It confirms to conforms to the IS document type wm.tn.rec:BizDocEnvelope.

**What is a Flat File Schema? What is the purpose of Flat File Schema?**

A flat file schema is the blueprint that contains the instructions for parsing or creating a flat file. It is a model against which you validate a flat file. There are two kinds:

* Schemas that define the structure of EDI documents (for conversion and validation of EDI – edidata string to IData object)
* Schemas that define the structure of your internal-format documents (IData object to String which is sent to backend)

**What are the services for Flat File Schema?**

EDI Module has built-in services to create FF Schema of EDI documents using:

* SEF (Standard Exchange Format) which contains all the metadata of a EDI
* IDOC (for SAP users)

WmFlatFile package pub.flatFile.generate:createDocumentTypepub.flatFile.generate:createFFDictionary

**Explain convertToStrings andconvertToValues?**

wm.b2b.edi:convertToStrings- Converts an IS document (IData object) to a String, based on aflat file schema that you specify.

wm.b2b.edi:convertToValues- Converts an InputStream or String (e.g., an EDI transaction setdocument) to an IS document (IData object) based on the input flat file schema.

**What is a BAPI & IDOC?**

Business application programming interface is a SAP mechanism for synchronous transactions.

SAP supports EDI through Intermediate documents (IDOCS). IDOC (intermediate document) is a standard data structure for electronic data interchange (EDI) between application programs written for the popular SAP business system or between an SAP application and an external program.

IDOC serve as the vehicle for data transfer in SAP Application Link Enabling (ALE) system. Each IDOC is assigned a unique number for tracking and future reference.

IDOC are used for asynchronous transactions: Each IDOC generated exists as a self-contained text file that can then be transmitted to the requesting workstation without connecting to the central database.

**Types of IDOC**

Material Master - MATMAS

Vendor Master - CREMAS

Customer Master - DEBMAS

Bill Of Material - BOMMAT

**Structure of IDOC**

It Consists of several segments, and segments contain several fields.

It contains the following three type of records –

1.One Control Record. 2.One or many Data Record

3.One or many Status record.

**What is the basic difference between RosettaNet and ebXML standards?**

RosettaNet is a widely adopted standard that provides message content, choreography and transport specifications for high-tech manufacturing companies. It is slightly more mature than ebXML, and its standards development process is targeted at the high-tech industry.

RosettaNet uses the B2B conceptual model by the Business Internet Consortium. ebXML

(Electronic Business using eXtensible Markup Language) is a set of standards developed toenable enterprises to conduct business over the Internet.

RosettaNet is a vertical business process standards in various business domain areas. Their PIPs (Partner Interface Process) are very complete, and well thought out.

ebXML is a horizontal standard of B2B interaction. It is not targeted towards any particular industry.

**What are the different XML parsers?**

JDOM, SAX

SAX Simple API for XML uses an event callback mechanism requiring you to code methods tohandle events thrown by the parser as it encounters different entities within the XML document.

DOM Document Object Model parses an XML document and returns an instance oforg.w3c.dom.Document. This document object's tree must then be "walked" in order to process the different elements.

JDOM is an open source API, designed specifically for Java programmers, that represents an XMLtree as Elements and Attributes. JDOM can interact with SAX or DOM.

With JDOM, you construct an instance of a builder (org.jdom.input.SAXBuilder or org.jdom.input.DOMBuilder) and then invoke the build() method on the builder to construct a Document object from the input source (a File, an InputStream, a URL, etc.)

**What is Modeler?**

WebMethods Modeler is a graphical tool for building a business process model and executing the model. After you create the business process model, you can elect to generate the run time components needed to execute the model in the underlying webMethods integration platform.

You can build process models that bring together data from diverse sources in your Enterprise such as:

Internal systems and applications

Trading Partners

Web services

Mainframes

Workflows (human interactions)

**What is the purpose of monitor?**

webMethods Monitor is an administrative tool that you use to examine instances of business processes, services, integrations, and documents that the integration platform is processing or has finished processing. Besides viewing status information about your processes, services, integrations, and documents, you can also use Monitor to perform control tasks such as suspending or resuming processes or editing and resubmitting documents, services, or the step pipeline. webMethods Monitor retrieves information about processes, services, integrations, and documents by querying the audit logs and the Trading Networks database. The Monitor is hosted by an integration server. You access webMethods Monitor through a web browser.