**Latest TIBCO Version:**

TIBCO BW 6.4.2

TIBCO RV 8.4.5

TIBCO TRA 5.10.0

TIBCO EMS 8.4

TIBCO Admin 5.10.0

**Current Product TIBCO Version:**

TIBCO BW 6.3.3

TIBCO RV 8.4.4

TIBCO TRA 5.10.0

TIBCO EMS 8.1.0

TIBCO Admin 2.2.0 **(TEA)**

TIB\_activespaces\_2.2.1

TIB\_bwpluginAS\_6.3.0

**Old TIBCO Version:**

TIBCO BW 5.9.3

TIBCO RV 6.3.0

TIBCO TRA 5.7.3

TIBCO EMS 6.3.0

TIBCO Admin 5.7.0

**Tibco Active matrix** Bussiness Works :

### **Install in GUI Mode**

1. Run the TIBCO Universal Installer executable:
   1. TIBCOUniversalInstaller-mac.bin - Macintosh
   2. TIBCOUniversalInstaller-x86-64.bin - Unix
   3. TIBCOUniversalInstaller-x86-64 - Windows

### Install in Console Mode

1. Using a console window, navigate to the temporary directory that contains the universal installer and run the installer using of the following command lines:
   * TIBCOUniversalInstaller-mac.bin -console - Macintosh
   * TIBCOUniversalInstaller-x86-64.bin -console - Unix
   * TIBCOUniversalInstaller-x86-64 -console - Windows

### Install in Silent Mode

The following procedure explains how to install the software in silent mode. Any errors that occur during installation are listed in the installation log file (see the *User\_Home*/.TIBCO directory).

#### Configure the Installation Responses

A configuration file, TIBCOUniversalInstaller.silent, is provided with the universal installer. Copy the file to another name (but keep the .silent extension) and use it to configure your installation choices.

#### Install in Silent Mode

1. To run the installer execute the following command, replacing *myfile* with your response file name:

TIBCOUniversalInstaller -silent -V responseFile="myfile.silent"

If you are using the TIBCOUniversalInstaller.silent file, you need not supply the file name and can run TIBCOUniversalInstaller -silent.

**Java messaging service framework**

**Uses rendezvous messaging system**

**Models**

**Point to point (Queue )**

**Publish and subscribe ( Topic )**

**Multicast ( Topic )**

**Point to point**

**Producer (sender) consumer ( receiver)**

**If exclusive then first consumer will take that msg**

**JMS – Persistant(sender expects ack from server)**

**and non persistant**

**EMS- Reliable delivery**

**Client ack – ack all msgs received so far**

**Auto ack – automatically ack**

**Dups ok – lazily delay ack**

**No ack**

**Explicit client ack 🡪 rather than ack all msgs it ackneged individual msg**

**Explicit client dups ok ack**

**Create bridge source=type:queue1 target=type:topic1**

General Topic:

### Patterns**[**[**edit**](https://en.wikipedia.org/w/index.php?title=Enterprise_application_integration&action=edit&section=4)**]**

This section describes common design patterns for implementing EAI, including integration, access and lifetime patterns. These are abstract patterns and can be implemented in many different ways. There are many other patterns commonly used in the industry, ranging from high-level abstract design patterns to highly specific implementation patterns.[[4]](https://en.wikipedia.org/wiki/Enterprise_application_integration#cite_note-Enterprise_Integration_Patterns-4)

#### **Integration patterns[**[**edit**](https://en.wikipedia.org/w/index.php?title=Enterprise_application_integration&action=edit&section=5)**]**

There are two patterns that EAI systems implement:[[5]](https://en.wikipedia.org/wiki/Enterprise_application_integration#cite_note-5)

[**Mediation**](https://en.wikipedia.org/wiki/Data_mediation)**(intra-communication)**

Here, the EAI system acts as the go-between or broker between multiple applications. Whenever an interesting event occurs in an application (for instance, new information is created or a new transaction completed) an integration module in the EAI system is notified. The module then propagates the changes to other relevant applications.

[**Federation**](https://en.wikipedia.org/wiki/Federation_(information_technology))**(inter-communication)**

In this case, the EAI system acts as the overarching facade across multiple applications. All event calls from the 'outside world' to any of the applications are front-ended by the EAI system. The EAI system is configured to expose only the relevant information and interfaces of the underlying applications to the outside world, and performs all interactions with the underlying applications on behalf of the requester.

Both patterns are often used concurrently. The same EAI system could be keeping multiple applications in sync (mediation), while servicing requests from external users against these applications (federation).

#### **Access patterns[**[**edit**](https://en.wikipedia.org/w/index.php?title=Enterprise_application_integration&action=edit&section=6)**]**

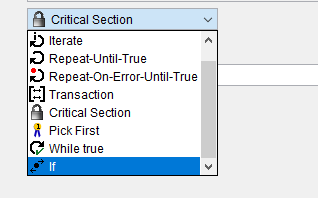
EAI supports both asynchronous (fire and forget) and synchronous access patterns, the former being typical in the mediation case and the latter in the federation case.[[*citation needed*](https://en.wikipedia.org/wiki/Wikipedia:Citation_needed)]

#### **Lifetime patterns[**[**edit**](https://en.wikipedia.org/w/index.php?title=Enterprise_application_integration&action=edit&section=7)**]**

An integration operation could be short-lived (e.g. keeping data in sync across two applications could be completed within a second) or long-lived (e.g. one of the steps could involve the EAI system interacting with a human [work flow](https://en.wikipedia.org/wiki/Work_flow) application for approval of a loan that takes hours or days to complete).[[*citation needed*](https://en.wikipedia.org/wiki/Wikipedia:Citation_needed)]

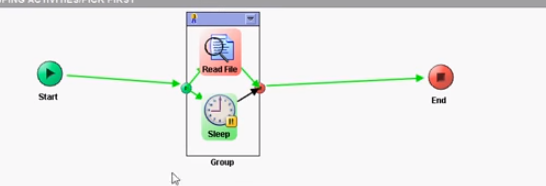
Groups: ( 8 + 1 “No Action Group” )

Index values always start from “1” in TIBCO . $Index always start from 1.



**Pick First** – In block if there three flow if first flow completes then that process itself will be completed . It wont wait for other flow in that block.

e.g Read file completed first. So it got completed.



**If-** when this block automatically numbers will be created.

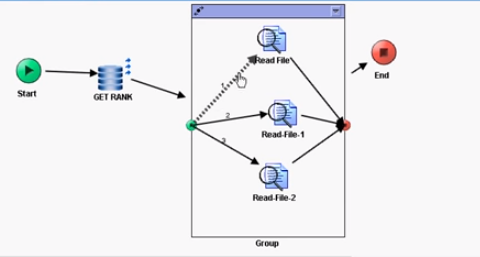
If(cond1)then //success with condition

{ stmt;} else

If(Cond2) then //success with condition

{ stmt;} else

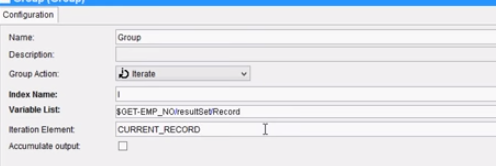
{ stmt;} // Success with no matching condition



Repeat UNtill TRUE & WHILE & REPEAT ON ERROR UNTILL TRUE

* But only Repeat On Error Until True group will not have accumulate output.

**Iterate:** will have variable list /iterate element / index . Based on variable list it will be repeating



**Repeat On Error Until True –** Repeat the loop until the condition evaluates to true. Executed at least only.

**Repeat Until True -** Until the condition satisfies it will iterate

**While –** loop will terminate only the condition fails

**JDBC Transaction** – If some activity fails in the group all other committed JDBC Transaction in that group will be roll backed.

**Critical Section –** Synchronize process instance. If we have read file in Critical Group. But we have that same kind of process in several process. We have use “Multiple Groups”.

We have to use “Single Group “ only three transition try to read same file.

**Single Group:** While this instance is running inside critical section and is in Sleep state; create another job (another instance) of the process. As you can see in below screenshot; second instance isn’t executing the critical section as It’s waiting for the first instance to complete its critical section

execution.

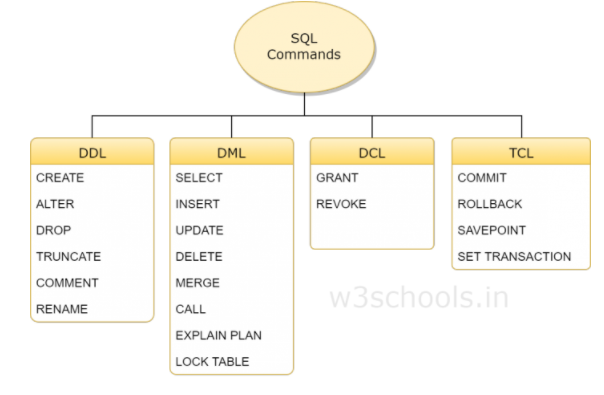
**JMS Local Transaction Group:** When u will place ur actvities in JMS Transaction than while sending message to EMS server one session will be mantained per connection and message will reside on EMS server in its memory and u won"t see any message on queue and client wont able to recieve any message and once transaction is completed all message will be available to client if failed messages will be deleted by ems server.

**What is the different feature of SQLDirect apart from other JDBC Activities?**

**SQLDirect –** Dynamic Queries and Only in SQLDirect we can execute DDL (CREATE, ALTER …ETC) Commands (DDL)

**JDBCQuery** – SELECT

**JDBCUpdate** – INSERT, UPDATE, DELETE



EMS:

TIBCO EMS (Enterprise Messaging Service) **is used for communication between processes and applications through a centralized EMS Server**. EMS simplifies as well as standardizes integration of complex applications in an enterprise environment.

The difference between JMS and TIBCO EMS is that **JMS provides two types** of delivery modes which are **Persistent** and **Non-Persistent** while **TIBCO EMS adds** another type of delivery mode which is **RELIABLE** delivery mode.

for restriction of messages acknowledgement, JMS provides**NO\_ACKNOWLEDGE** mode while **TIBCO EMS** extends the functionality of JMS by providing **EXPLICIT\_CLIENT\_ACKNOWLEDGE** mode and **EXPLICIT\_CLIENT\_DUPS\_OK\_ACKNOWLEDGE** mode.

 In **PERSISTENT delivery mode**, messages sent by the producer to the EMS Server are **persisted/stored in a disk** and a confirmation is sent to the producer for every message.

**Non-Persistent delivery mode**, messages sent by the producer **are not stored/persisted** on EMS Server

Ans: **Static EMS queues** **are created using EMS Administrator** or by configuring directly in the configuration **file (queues.conf)** and are permanent and can be used at enterprise level. Static queues have a permanent **TTL (Time To Live)** and they exist till the time they are deleted manually.

On the other hand, **dynamic queues are short lived queues which are created when needed and get expired once they are not in use.** Dynamic queues remain alive until at least one client is associated with them.

Purge queue queueName

bridges between destinations are configured in a configuration file named as **bridges.conf**

Bridges are used **to connect multiple destinations (queues or topics) so that** same message can be sent to multiple destinations.  When a bridge exists between two queues, the message is delivered to both queues.

All EMS Server properties are configured in the **file ( tibemsd.conf )**

6. What is structure of JMS Message?  
• Header(Required)  
• Properties(optional)  
• Body(optional)

EMS supports messages up to a maximum size of 512MB.

EMS extends the JMS delivery modes to include reliable delivery. Sending a RELIABLE\_DELIVERY message omits the server confirmation to improve performance regardless of the authorization setting.  
When using RELIABLE\_DELIVERY mode, the server never sends the producer a receipt confirmation or access denial and the producer does not wait for it. Reliable mode decreases the volume of message traffic, allowing higher message rates, which is useful for messages containing time-dependent data, such as stock price quotations.

Fault Tolerance:

Hearbeat failure:-Primary server sends a heartbeat message to backup server to indicate primary server is working .  
connection failure :-backup server detects the failure of tcp connection with primary server

Some times the producer may send messages faster than the consumers can receive them. So, the message capacity on the server will be exhausted. So we use flow control. Flow control can be specified on destinations.

**create topic :** Creates a topic with specified name and properties. Properties are listed in a  
comma-separated list, as described in topics.conf . You can set the properties directly in the topics.conf  
or by means of the setprop topic command in the EMS Administrator Tool.  
syntax : create topic <topic\_name> <[properties]>  
example: create topic t1

**Show Topic : Shows the details for the specified topic**  
syntax : show topic <topic-name>  
example: show topic t1

**purge topic :  Purge all messages for all subscribers on the named topic**  
syntax : purge topic <topic-name>  
example: purge topic t1

**delete topic : delete specefic topic**  
syntax : delete topic <topic-name >  
example: delete topic t1

**addprop topic : Adds properties to the topic. Property names are separated by commas**  
syntax : addprop topic <topic\_name> <properties,…>  
example: addprop topic t1 failsafe

**setprop topic : Set topic properties, overriding any existing properties**  
syntax : setprop topic <topic-name> <properties>  
example: setprop topic t1 secure,sender\_name

**create queue :** Creates a queue with the specified name and properties. The possible queue properties are described in  
Destination Properties. Properties are listed in a comma-separated list, as described in queues.conf. You can set the  
properties directly in the queues.conf or by means of the setprop queue command in the EMS Administrator Tool.  
syntax : create queue <queue\_name> <[properties]>  
example: create queue q1

**Show Queue : Shows the details for the specified queue**  
syntax : show queue <queue-name>  
example: show queue q1

**setprop queue : Set queue properties, overriding any existing properties. Any properties on a topic that are not explicitly specified by this command are removed**  
syntax : setprop queue <queue-name> <properties>  
example: setprop queue q1 secure,sender\_name

**addprop queue : Adds properties to the queue. Property names are separated by commas**  
syntax : addprop queue <topic\_name> <properties,…>  
example: addprop queue q1 failsafe

**purge queue : Purge all messages in the named queue**  
syntax : purge queue <queue-name>  
example: purge queue q1

**delete queue : delete specefic queue**  
syntax : delete queue <topic-name >  
example: delete queue t1

create bridge source=queue:bridgequeue target=topic:bridge

create bridge source=queue:bridgequeue target=topic:bridge  
topic

# TIBCO EMS – Properties of Queues and Topics (Where Tuning can be done)

[April 3, 2017](https://haritibcoblog.com/2017/04/03/tibco-ems-properties-of-queues-and-topics-where-tuning-can-be-done/)[Heuristic Researcher](https://haritibcoblog.com/author/hsivabc/)

You can set the properties directly in the topics.conf or queues.conf file or by means of the setprop topic or setprop queue command in the EMS Administrator Tool.

**1)   Failsafe**

The failsafe property determines whether the server writes persistent messages to disk synchronously or asynchronously.

Ø  When failsafe is not set, messages are written to the file on disk in asynchronous mode to obtain maximum performance. In this mode, the data may remain in system buffers for a short time before it is written to disk and it is possible that, in case of software or hardware failure, some data could be lost without the possibility of recovery

Ø  In failsafe mode, all data for that queue or topic are written into external storage in synchronous mode. In synchronous mode, a write operation is not complete until the data is physically recorded on the external device

The failsafe property ensures that no messages are ever lost in case of server failure

**2) Secure**

Ø  When the secure property is enabled for a destination, it instructs the server to check user permissions whenever a user attempts to perform an operation on that destination.

Ø  If the secure property is not set for a destination, the server does not check permissions for that destination and any authenticated user can perform any operation on that topic or queue.

**2)   Maxbytes**

Ø  Topics and queues can specify the maxbytes property in the form:

Maxbytes=value [KB|MB|GB]                   Ex: maxbytes=1000MB

Ø  For queues, maxbytes defines the maximum size (in bytes) that the queue can store, summed over all messages in the queue. Should this limit be exceeded, messages will be rejected by the server and the message producers send calls will return an error

Ø  If maxbytes is zero, or is not set, the server does not limit the memory allocation for the queue

Ø  For queues, maxbytes defines the maximum size (in bytes) that the queue can store, summed over all messages in the queue. Should this limit be exceeded, messages will be rejected by the server and the message producer sends calls will return an error

**4) maxmsgs**

Ø  Where value defines the maximum number of messages that can be waiting in a queue. When adding a message would exceed this limit, the server does not accept the message into storage, and the message producer’s send call returns an error.

Ø  If maxmsgs is zero, or is not set, the server does not limit the number of messages in the queue.

Ø  You can set both maxmsgs and maxbytes properties on the same queue. Exceeding either limit causes the server to reject new messages until consumers reduce the the queue size to below these limits.

**5) OverflowPolicy**

Topics and queues can specify the overflowPolicy property to change the effect of exceeding the message capacity established by either maxbytes or maxmsgs.

o   OverflowPolicy=default | discardOld | rejectIncoming

1. **Default**

Ø  For topics, default specifies that messages are sent to subscribers, regardless of maxbytes or maxmsgs setting.

Ø  For queues, default specifies that new messages are rejected by the server and an error is returned to the producer if the established maxbytes or maxmsgs value has been exceeded.

1. **DiscardOld**

Ø  For topics, discardOld specifies that, if any of the subscribers have an outstanding number of undelivered messages on the server that are over the message limit, the oldest messages are discarded before they are delivered to the subscriber.

Ø  The discardOld setting impacts subscribers individually. For example, you might have three subscribers to a topic, but only one subscriber exceeds the message limit. In this case, only the oldest messages for the one subscriber are discarded, while the other two subscribers continue to receive all of their messages.

Ø  For queues, discardOld specifies that, if messages on the queue have exceeded the maxbytes or maxmsgs value, the oldest messages are discarded from the queue and an error is returned to the message producer

**III.                rejectIncoming**

Ø  For topics, rejectIncoming specifies that, if any of the subscribers have an outstanding number of undelivered messages on the server that are over the message limit, all new messages are rejected and an error is returned to the producer.

Ø  For queues, rejectIncoming specifies that, if messages on the queue have exceeded the maxbytes or maxmsgs value, all new messages are rejected and an error is returned to the producer.

**6) global**

Ø  Messages destined for a topic or queue with the global property set are routed to the other servers that are participating in routing with this server.

You can set global using the form:   global

**7) sender\_name**

Ø  The sender\_ name property specifies that the server may include the sender’s username for messages sent to this destination.

You can set sender\_name using the form:    sender\_name

**8) sender\_name\_enforced**

Ø  The sender\_name\_enforced property specifies that messages sent to this destination must include the sender’s user name. The server retrieves the user name of the message producer using the same procedure described in the sender\_name property above. However, unlike, the sender\_name property, there is no way for message producers to override this property.

You can set sender\_name\_enforced using the form:    sender\_name\_enforced

Ø  If the sender\_name property is also set on the destination, this property overrides the sender\_name property.

**9) FlowControl**

Ø  The flowControl property specifies the target maximum size the server can use to store pending messages for the destination. Should the number of messages exceed the maximum; the server will slow down the producers to the rate required by the message consumers. This is useful when message producers send messages much more quickly than message consumers can consume them.

If you specify the flowControl property without a value, the target        maximum is set to 256KB.

Ø  The flow\_control parameter in tibemsd.conf file must be set to enable before the value in this property is enforced by the server. See Flow Control for more information about flow control.

**10) trace**

Ø  Specifies that tracing should be enabled for this destination.

o    You can set trace using the form:    trace [=body]

Ø  Specifying trace (without =body), generates trace messages that include only the message sequence and message ID. Specifying trace=body generates trace messages that include the message body

**11) import**

Ø  The import property allows messages published by an external system to be received by an EMS destination (a topic or a queue), as long as the transport to the external system is configured.

o    You can set import using the form:    import=”list”

**12) export**

Ø  The export property allows messages published by a client to a topic to be exported to the external systems with configured transports.

o    You can set import using the form:    export=”list”

Ø  It supports for only topics not queues.

**13) maxRedelivery**

Ø  The maxRedelivery property specifies the number of attempts the server should make to redeliver a message sent to a queue.

o    You can set maxRedelivery using the form:    maxRedelivery=count

Ø  Where count is an integer between 2 and 255 that specifies the maximum number of times a message can be delivered to receivers. A value of zero disables maxRedelivery, so there is no maximum.

Ø  Once the server has attempted to deliver the message the specified number of times, the message is either destroyed or, if the JMS\_TIBCO\_PRESERVE\_UNDELIVERED property on the message is set to true, the message is placed on the undelivered queue so it can be handled by a special consumer

**Undelivered Message Queue**

If a message expires or has exceeded the value specified by the maxRedelivery property on a queue, the server checks the message’s JMS\_TIBCO\_PRESERVE\_UNDELIVERED property. If  
JMS\_TIBCO\_PRESERVE\_UNDELIVERED is set to true, the server moves the message to the undelivered message queue, $sys.undelivered. This undelivered message queue is a system queue that is always present and cannot be deleted. If JMS\_TIBCO\_PRESERVE\_UNDELIVERED is set to false, the message will be deleted by the server.

**14) exclusive**

Ø  The exclusive property is available for queues only (not for topics).

Ø  When exclusive is set for a queue, the server sends all messages on that queue to one consumer. No other consumers can receive messages from the queue. Instead, these additional consumers act in a standby role; if the primary consumer fails, the server selects one of the s   tandby consumers as the new primary, and begins delivering messages to it.

Ø  By default, exclusive is not set for queues and the server distributes messages in a round-robin—one to each receiver that is ready. If any receivers are still ready to accept additional messages, the server distributes another round of messages—one to each receiver that is still ready. When none of the receivers are ready to receive more messages, the server waits until a queue receiver reports that it can accept a message.

**15) prefetch**

The message consumer portion of a client and the server cooperate to regulate fetching according to the prefetch property. The prefetch property applies to both topics and queues.

You can set  prefetch using the form:  prefetch=value

where value is one of the values in 2 0r more ,1,0,None.

**Value Description**

Ø  2 or more: The message consumer automatically fetches messages from the

server. The message consumer never fetches more than the number of messages specified by value.

Ø  1 :-The message consumer automatically fetches messages from the server initiating fetch only when it does not currently hold amessage.

Ø  None:-Disables automatic fetch. That is, the message consumer initiates fetch only when the client calls receive—either an explicit synchronous call, or an implicit call (in an asynchronous consumer).This value cannot be used with topics or global queues.

Ø  0:-The destination inherits the prefetch value from a parent

destination with a matching name. If it has no parent, or nodestination in the parent chain sets a value for prefetch, then the default value is 5 queues and 64 for topics.

Ø  When a destination does not set any value (i.e prefetch value is empty)for prefetch, then the default value is 0 (zero; that is, inherit the prefetch value).

**16) expiration**

Ø  If an expiration property is set for a destination, when the server delivers a message to that destination, the server overrides the JMSExpiration value set by the producer in the message header with the time specified by the expiration property.

o    You can set the expiration property for any queue and any topic using the form:

expiration=time [msec|sec|min|hour|day]

Ø  where time is the number of seconds. Zero is a special value that indicates messages to the destination never expire.

**TIBCO Admin:**

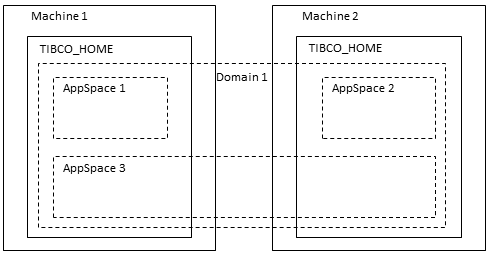
A domain is a logical group that provides an isolated environment for applications and their resources. Runtime entities such as AppSpaces and AppNodes are contained within a domain.

An AppSpace is a collection of one or more AppNodes.

A domain can contain one or more AppSpaces. AppSpaces can span multiple physical machines across networks. An AppSpace manifests on the physical machine as a predefined folder structure that contains information about the applications deployed in that domain.

Each AppSpace contains one or more execution runtimes called AppNodes which host the applications. When you deploy an application to an AppSpace, the application is deployed to all AppNodes that are part of the same AppSpace. An AppSpace is elastic, which allows AppNodes to be added dynamically to scale the load on an application, thereby providing load-balancing and fault-tolerance for applications. You can add and remove AppNodes to an AppSpace even after an application has been deployed. For more information, see Managing AppSpaces and AppNodes.

Following diagram shows AppSpace1 on Machine 1, AppSpace2 on Machine 2, and AppSpace3 spanning Machine 1 and Machine 2:



1.Ls – will list the domains or appspaces or appnodes depending upon the current location.

2.Cd --similar to linux command. Helps to navigate inside domain,appspace and appnode.

3.To see the existing domains,appspaces, appnodes or agent, use the show command.

show application

show domain

show appspace

show appnode

show agent.

4.To download existing profile use export command.

**Syntax:-**

Export  -d <<domain name>> -a <<appspace name>> -n <<appnode>>  application <<application name>> <<app version>>

For example to download order-preparation profile inside TIB\_PT\_BWDOMAIN1 domain the command would be.

export -d TIB\_PT\_BWDOMAIN1 -a PTDomain\_OrderPreparation\_AppSpaces -n PTDomain\_OrderPreparation\_AppNode1 application DMW-ORDER-PREPARATION 1.0

5.To upload a profile to an application use config command.

**Syntax:-**

Config -pf <<profile name with path>>  -d <<domain>> -a<<appspace>> -n<<appnode>>  application <<application name>> <<app version>>

For example to configure a profile to order-preparation in appnode PTDomain\_OrderPreparation\_AppNode2 under domain TIB\_PT\_BWDOMAIN1 the command would be

config -pf "/app/opt/tibco/bw/6.3/bin/DMW-ORDER-PREPARATION\_PTDomain\_OrderPreparation\_AppNode2\_default.substvar" -d TIB\_PT\_BWDOMAIN1 -a PTDomain\_OrderPreparation\_AppSpaces -n PTDomain\_OrderPreparation\_AppNode2  application DMW-ORDER-PREPARATION 1.0

6. We also have upload,deploy and undeploy commands. But I have not used them. Please find the reference for the same.

<https://docs.tibco.com/pub/activematrix_businessworks/6.2.2/doc/html/GUID-E8F42EBC-3A06-4CB4-B932-4122FA8B18E9.html>

<https://docs.tibco.com/pub/activematrix_businessworks/6.2.2/doc/html/GUID-EC830BFA-2C63-4B5D-835B-FD6DF0B94D72.html>

<https://docs.tibco.com/pub/activematrix_businessworks/6.2.2/doc/html/GUID-07DDBA24-FD1C-4807-8754-33559844589F.html>

7.If you need to restart the application you can use start and stop command.

**Syntax:-**

Start  -d <<domain>> -a<<appspace>> -n<<appnode>>  application <<application name>> <<app version>>

Stop  -d <<domain>> -a<<appspace>> -n<<appnode>>  application <<application name>> <<app version>>

In PT restarting the appnode will restart the application. To restart order preparation application you can use the following.

Stop -n PTDomain\_OrderPreparation\_AppNode1

Start -n PTDomain\_OrderPreparation\_AppNode1.

AS:

**TIBCO ActiveSpaces** is a TIBCO application that  provides a distributed in-memory data grid for an increase in performance by reducing reliance on costly transactional systems.

TBCO ActiveSpaces is a right choice to use when you have complex environment with a huge flux of data and you have to go for large number of database operations. With use of ActiveSpaces, performance can be increased greatly as in-memory data access is very fast as compared to a disk stored database access.

In this **step by step tutorial,** I will explain how we can use **TIBCO BW Plugin for ActiveSpaces** to store data to ActiveSpaces.  You must have ActiveSpaces software and **ActiveSpaces Plugin for BW**already installed in order to proceed with this tutorial.

### **What is the use of Check Point activity ?**

To detect duplicate messages, it is important to place the Checkpoint activity before any activities that we do not want to execute more than once.

We must specify a value for the duplicateKey element in theCheckpoint activity input schema.

This value should be some unique key contained in the event data that starts the process.

How it works ?

For example, the TransactionID value is unique for all new orders.

The following describes the procedure for duplicate detection by the process engine:

1. An incoming message is received and a process instance is created.

2. Activities in the process instance are executed until the first Checkpoint activity is reached. The Checkpoint activity has a value specified for the duplicateKey input element.

3. The process engine checks the current list of duplicateKeyvalues for a matching value.

a. If no process instance has stored the given duplicateKeyvalue, the process engine stores the value and completes theCheckpoint activity.

b. If another process instance has already stored the given duplicateKey value, then process engine terminates the process and throws a DuplicateException.

Process Engine Properties for Duplicate Detection

The following process engine properties need to set inbin/bwengine.xml to control duplicate key detection.

• bw.engine.dupKey.enabled —

specifies whether duplicate detection is performed.

true (the default) - indicates the process engine will check for identical duplicateKey values.

false -indicates duplicateKeys when specified are ignored.

• bw.engine.dupKey.timeout.minutes —

 specifies how long (in minutes) to keep stored duplicateKey values.

The default is 30 minutes.

Value -1 indicates the duplicateKey values are deleted when the job completes.

Value 0 indicates to store duplicateKey values indefinitely.

Any positive integer greater than 0 indicates the number of minutes to keep stored duplicateKeys.

• bw.engine.dupKey.pollPeriod.minutes —

specifies the number of minutes to wait before polling for expired duplicateKey values.

# TIBCO AppManage Utility EAR Deployment Step By Step Tutorial

By [Ajmal Abbasi](http://tutorialspedia.com/author/admin/) | February 4, 2015

[16 Comments](http://tutorialspedia.com/tibco-appmanage-utility-ear-deployment-step-by-step-tutorial/#comments)

In one of my previous tutorials, I explained the [**steps to deploy EAR in a TIBCO Domain using Administrator GUI**](http://tutorialspedia.com/how-to-deploy-ear-in-tibco-administrator/). Now in this **step by step tutorial,** I am going to explain how you can **deploy EAR in a domain using Appmanage utility through command line.**

## What is AppManage Utility?

Appmanage is a utility which is available in the bin folder of TRA and can be used to extract deployment configurations from EAR and to deploy EAR in administration domain through command line.

Appmanage utility can also be used to undeploy an application from domain, delete application from domain and to start and stop application services from Tibco administrative domain.

Now after explaining basics of Appmanage utility and its usage, let’s proceed with step by step tutorial of deploying EAR to administrator domain using Appmanage utility commands.

## Step 0: Develop a Simple Application in TIBCO Designer and Create EAR File

I am calling this step as Step 0 as Its not part of actual deployment, rather its kind of a pre-requisite as we need to first have a EAR available. For this example case, I just created a simple HTTP Application by designing a TIBCO process that receives a HTTP Request on a given port and then sends back a response using Send HTTP Response activity.  
I am not going to explain the steps of EAR creation in this tutorial as that has been explained in one of my previous tutorials already.

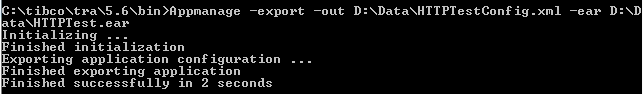
For this tutorial, we just proceed from the point where we have EAR file already created with the name HTTPTest.ear

## Step 1: Export Deployment Configuration from EAR using Appmanage Utility

As I mentioned earlier that AppManage utility can be used to export deployment configuration information from a EAR. For our case, we want to extract deployment configurations from the EAR HTTPTest.ear to a file (HTTPTestConfig.xml).

Below command will extract the configurations and save the XML in the given directory:

*AppManage -export -out D:\Data\HTTPTestConfig.xml -ear HTTPTest.ear*

[](http://tutorialspedia.com/wp-content/uploads/2015/02/appmanage-utility-export-configuration-from-ear.png)

Please note that in order to run the above command, you should first move to the bin folder of TRA where AppManage utility is present. (It is in the location TIBCO\_HOME/TRA/5.xx/bin/)

Once you h ave exported configurations in an XML file, you can make any changes in the file before proceeding to the deployment step. (e.g. you may need to make changes to some global variables like HTTP Port or Host name before deployment).

## Step 2: Deploy EAR using AppManage Utility

In order to deploy EAR to administration domain, we use below command:

*AppManage -deploy -ear D:\Data\HTTPTest.ear -deployConfig D:\Data\HTTPTestConfig.xml -app HTTPTestApp -user admin -pw admin -domain TIBCO\_AJMAL*

Explanation of different arguments of the above command is given below:

-ear : Specify name of the EAR to be deployed

-deployConfig : Specify deployment confguration file to be used for deployment of EAR

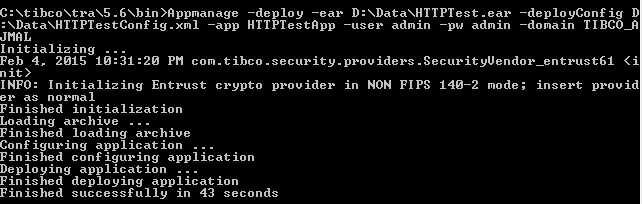
-app : Specify the name of the application once it will be deployed

-user :User name of TIBCO administrative domain

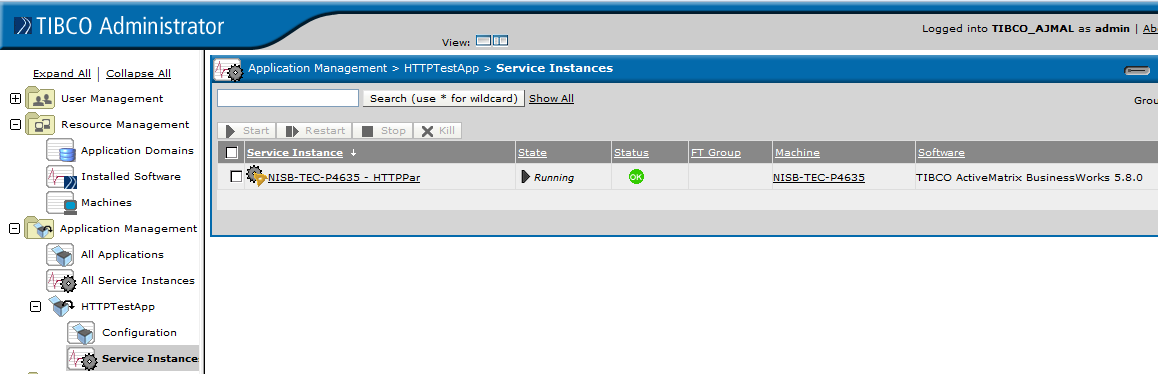
-pw : Password of domain

-domain : Name of domain in which application should be deployed

As you can see in below screenshot that command has run successfully and has deployed the application in the domain specified:

[](http://tutorialspedia.com/wp-content/uploads/2015/02/tibco-appmanage-deploy-ear-using-command-line.png)

You can verify the correct deployment of the application by going to TIBCO Administrator and check under Application

[](http://tutorialspedia.com/wp-content/uploads/2015/02/deployed-application-using-appmanage-in-tibco.png)

This completes tutorial on EAR deployment using Appmanage utility. I hope it will be useful for you. Feel free to comment/contact for any further help. Thanks

AppManage Commands

[TIBCO BusinessWorks™](https://community.tibco.com/products/tibco-activematrix-businessworks)

###### **By:**

[swapnil\_bankar\_4523](https://community.tibco.com/users/swapnilbankar4523)

Last updated:

2:22pm Nov 17, 2015

[Flag as Inappropriate](https://community.tibco.com/user/login?destination=node/52201)

[#AppManage](https://community.tibco.com/search/tags/appmanage-6056)

[Edit This PageCreate New Page](https://community.tibco.com/user/login/?destination=wiki/appmanage-commands)

Table of Contents

* [To Export Deployment conf file "Test\_Deploy.xml" from EAR](https://community.tibco.com/wiki/appmanage-commands" \l "toc-2)
* [To Export Deployment conf file "Test\_Deploy.xml" from Administrator](https://community.tibco.com/wiki/appmanage-commands#toc-3)
* [To Export Deployment conf file "Test\_Deploy.xml" from EAR & compare with previous deployment conf file "Test\_Deploy\_2.xml" by creating a log file "Test\_Deploy.xml.log"](https://community.tibco.com/wiki/appmanage-commands#toc-4)
* [Deploy the Ear file using updated deployment conf file](https://community.tibco.com/wiki/appmanage-commands#toc-5)

### **To Export Deployment conf file "Test\_Deploy.xml" from EAR**

AppManage -export

-out D:\Devel\IDE\AMXBW\_Workspace\Test\_Deploy.xml

-ear D:\Devel\IDE\AMXBW\_Workspace\Test\_Deploy.ear

-app SIT01/Test\_Deploy

-domain TIBCO\_ADMIN\_DOMAIN

-user admin

-pw admin

### **To Export Deployment conf file "Test\_Deploy.xml" from Administrator**

AppManage -export

-out D:\Devel\IDE\AMXBW\_Workspace\Test\_Deploy.xml

-app SIT01/Test\_Deploy

-domain TIBCO\_ADMIN\_DOMAIN

-user admin

-pw admin

### **To Export Deployment conf file "Test\_Deploy.xml" from EAR & compare with previous deployment conf file "Test\_Deploy\_2.xml" by creating a log file "Test\_Deploy.xml.log"**

AppManage -export

-ear D:\Devel\IDE\AMXBW\_Workspace\Test\_Deploy.ear

-deployconfig D:\Devel\IDE\AMXBW\_Workspace\Test\_Deploy\_2.xml

-out D:\Devel\IDE\AMXBW\_Workspace\Test\_Deploy.xml

-app SIT01/Test\_Deploy

-domain TIBCO\_ADMIN\_DOMAIN

-user admin

-pw admin

Resultant log file will have following content (with mention of difference) for example:

\*\*\*\*\*\*\* Processing GV section with name Global Variables \*\*\*\*\*\*\*

\*\*\*\*\*\*\* Processing service section \*\*\*\*\*\*\*

\*\*\*\*\*\*\* Processing service section with name Addition\_ProcessArchive.par \*\*\*\*\*\*\*

\*\*\*\*\*\*\* Processing GV section with name Adapter SDK Properties \*\*\*\*\*\*\*

\*\*\*\*\*\*\* Processing bw checkpoint section \*\*\*\*\*\*\*

\*\*\*\*\*\*\* Processing bw process section \*\*\*\*\*\*\*

ADDED service archive: Subtraction\_ProcessArchive.par

### **Deploy the Ear file using updated deployment conf file**

[The generated file includes substitution variables for the machine element value.

The variables use the syntax, ***%%-machine%%***,  (a combination of percent symbols, archive type and computer name).

During deployment you must configure an element that uses such a variable by replacing the substitution value with the actual value, without the percent symbols]

AppManage -deploy

-ear D:\Devel\IDE\AMXBW\_Workspace\Test\_Deploy.ear

-deployConfig D:\Devel\IDE\AMXBW\_Workspace\Test\_Deploy.xml

-app SIT01/Test\_Deploy

-domain TIBCO\_ADMIN\_DOMAIN

-user admin

-pw admin

# TIBCO APPMANAGE UTILITY WITH EXAMPLE

[MAY 22, 2013](https://sriksolutions.wordpress.com/2013/05/22/tibco-appmanage-utility-with-example/) [SRIK](https://sriksolutions.wordpress.com/author/sriksolutions/) [LEAVE A COMMENT](https://sriksolutions.wordpress.com/2013/05/22/tibco-appmanage-utility-with-example/#respond)

16 VOTES

The TIBCO appmanage utility is a powerful tool for a TIBCO adminstrator professional. With this you can get a deployment configuration file from ear file or from TIBCO administrator GUI without logging in to admin GUI. You can also upload ear file and configuration to domain administrator through back end and then deploy using the same utility. With this you can deploy your ear file into multiple domains and instances. It would be a seamless and smooth deployment if properly used. One another functionality of this utility is to stop/start applications which is great in defining a scheduled restart. Undeploy functionality is in place as well. All the said functionality ill be discussed on the rest of this tutorial.

## Export configuration file from ear archive

**Example:**AppManage -export -ear C:/temp/mySRIK.ear -out C:/temp/mySRIK.xml

**Output:** Configuration file is exported in C:/temp/mySRIK.xml. This XML contains the current GV define on the ear file

## Export configuration file from deployed application

AppManage -export -app Reporting/Adapter/AsiaReportModule -out C:/temp/AsiaReportModule.xml -domain AdminSRIK -cred cred\_teknoscope.tra

**Output:** Config file is exported in C:/temp/*AsiaReportModule.xml*

**Note:** cred file can be substituted with parameter -user <user> -pw <password>  
**Exporting all config files on a domain using tibco appmanage**

**Example:**AppManage -batchexport -domain AdminSRIK -cred cred\_srik.tra -dir C:/temp/export\_srik\_0410 -noear

**Output:** Config file and ear file is exported in C:/temp/export\_srik\_0410 with folder nam%s based on the application path as seen on the admin

**Deploy command using ear and config xml**…..  
AppManage -deploy -ear C:\temp\HOToStore.ear -deployconfig C:\temp\HOToStore.xml -domain TIBADMIN -user tibco -pw tibco  
**UnDeploy Command**

AppManage -undeploy -app HOToStore/HOToStore -domain TIBADMIN -user tibco -pw tibco  
**Deploy command for deploying an application that is in the undeployed state**

AppManage -deploy -app HOToStore/HOToStore -domain TIBADMIN -user tibco -pw tibco  
**Delete Command**

AppManage -delete -app HOToStore/HOToStore -user tibco -pw tibco -domain TIBADMIN -force

**Tibco AppManage Commands**

#### Extract all deployed EAR files in a tibco administrator domain

#### C:\Apps\tibco\tra\5.7\bin>AppManage -batchExport -user admin -pw admin  -domain Testdomain -dir c:/temp/TFS

#### Extract particular EAR from the Administrator domain

#### C:\Apps\tibco\tra\5.7\bin>AppManage -export -out C:\temp\OnyxAdmin.xml -app OnyxAdmin -user admin -pw admin -domain Testdomain

#### Extract information from above extracted  EAR in the form of xml

#### C:\Apps\tibco\tra\5.7\bin>Appmanage  -export  -ear "C:\temp\OnyAdmin\OnyxAdmin.ear" -out "C:\Test\EAR\OnyxAdmin.xml"

#### Start all BW services from the Administrator domain

#### C:\Apps\tibco\tra\5.7\bin>AppManage  -batchstart  -user  admin  -pw admin -domain Testdomain  -dir c:/temp/TFS

#### Stop all BW services from the Administrator domain

#### \    C:\Apps\tibco\tra\5.7\bin>AppManage  -batchstop  -user  admin  -pw  admin -domain Testdomain -dir c:/temp/TFS

#### Kill all BW services from the Administrator domain

#### C:\Apps\tibco\tra\5.7\bin>AppManage –batchkill  -user  admin  -pw Adm1n1strat0r -domain TFS -dir c:/temp/TFS

#### 

#### Deploy the EAR file from the Administrator domain

#### C:\Apps\tibco\tra\5.7\bin>AppManage -deploy -ear myTest.ear -deployConfig c:\test\myApp.xml -app myApp -user admin -pw admin-domain Testdomain

#### Build  an EAR file for a particular project

#### C:\Apps\tibco\tra\5.7\bin>buildear -s -ear -o "C:\Test\myportal.ear" -p "C:\Test\samplePORTAL"

In addition to the -export and -deploy options, the AppManage utility allows you to use:

* -upload to upload an application into an administration domain without configuring the application’s deployment options.
* -config to upload an application along with its deployment configuration file, which defines the application’s configuration options, but not deploy the application.
* -undeploy to undeploy an application.
* -delete to remove an application from an administration domain.
* -start or -stop to start or stop a service or process configured under an application.
* -override to use global variable values defined in the EAR file, instead of those defined in the original deployment configuration file when redeploying.
* -min to generate only XML tags for options you have changed.
* -max to export a template deployment configuration file with every possible setting included.
* -template to generate a deployment configuration file in template format.
* -password to prompt for a password that is used to encrypt or decrypt sensitive data in the deployment configuration file.
* -passwordFile to use a password file to encrypt or decrypt sensitive data in the deployment configuration file.
* -MoveAppData to change the transport setting for a given application.
* -truncate to truncate the application deployment revision.
* -desc to specify a description for the deployed application.
* -serialize to deploy service instances one at a time instead of in parallel.
* -exportDeployed to export the configuration for the last successful deployment rather than what is currently being modified and will be used for the next deployment.

-upload Option

You can use the AppManage utility to upload an EAR file into an administration domain. Specifying the -upload option is identical to importing an EAR in the TIBCO Administrator GUI. The application is loaded, but no deployment options are specified and the application is not deployed.

AppManage -upload -ear c:\ears\timer\_wait.ear -user admin -pw admin -domain tp002

TIBCO Runtime Agent Scripting Deployment User’s Guide

<bw name="BW Processes.par"> Processes.par".

-service

Other Options |23

-config Option

You can upload an EAR file and a deployment configuration file into an administration domain and not deploy the application. The -config option uploads the EAR file and the deployment configuration file, but does not deploy the application. You can omit the -ear option if the EAR file is already loaded in the domain.

AppManage -config -ear c:\ears\timer\_wait.ear -deployConfig c:\ears\deployments\timer\_wait.xml -app

timer\_wait -user admin -pw admin -domain tp002

-undeploy Option

You can undeploy an application using a command line similar to the following. The application will remain in the domain, but in an undeployed state.

AppManage -undeploy -app timer\_wait -domain tp001 -user admin -pw

admin

-delete Option

You can remove an application from an administration domain. If the application is deployed, you can undeploy and delete the application in one operation using the -force option. An error is returned if you attempt to delete a deployed application without specifying the -force option.

AppManage -delete -app timer\_wait -user admin -pw admin -domain tp001

-force

-start Option

You can use the

use the utility to

service contains

AppManage utility to start an application and all its associated processes, or

start just one service. The -service tag takes the name of a service. Each

a name. For example, if there is a TIBCO BusinessWorks service element

in a deployment configuration file, takes the value "BW

AppManage -start -app myApp -user a -pw a -domain test

AppManage -start -app myApp -service "BW Processes.par" -user a -pw a -domain test"

In the case where -binding is provided without -service, all services in the application are started.

AppManage -start -app myApp -binding

ActiveDatabaseAdapterConfiguration -user a -pw a -domain test

TIBCO Runtime Agent Scripting Deployment User’s Guide

-binding

24 | Chapter 2 Getting Started

-stop Option

You can use the AppManage utility to stop an application and all its associated processes, or use the utility to stop just one service. The -binding tag takes the name of a binding. Each binding contains a name. For example, if there is a binding element <binding name="BW

Processes"> in a deployment configuration file, takes the value "BW Processes".

AppManage -stop -app myApp -user a -pw a -domain test

AppManage -stop -app myApp -service

ActiveDatabaseAdapterConfiguration.aar -user a -pw a -domain test

AppManage -stop -app myApp -service "BW Processes.par" -binding

"BW Processes" -user a -pw a -domain test

–override Option

This option is only applicable when a deployment configuration file already exists. That is, you are redeploying with a changed archive file.

By default, a newly generated deployment configuration file preserves the value in the original deployment config file. Use this option with the -export option to create a deployment configuration file that uses the values defined for global variables in the archive file, rather than the values defined for global variables in the original deployment configuration file.

AppManage -export -ear c:\ears\deployment\filenotify.ear

-deployconfig c:\ears\deployments\filenotify.xml

-out c:\ears\deployments\filenotify-changed.xml -override

-min Option

Use this option with the -export and -ear options to create a small deployment configuration file that only includes XML tags for options you have changed in the EAR file. XML tags are not generated for default options that have values, which have not been changed from their defaults. Options for which XML tags are not generated will use default values.

-max Option

Use this option with the -export option to create a deployment configuration file that includes all possible XML tags. For example:

appmanage -export -app SendMsg -domain tp041 -user admin -pw admin

-out c:\temp\sendmsg.xml -max

TIBCO Runtime Agent Scripting Deployment User’s Guide

Other Options |25

-template Option

Use this option with the -export option to create a deployment configuration file that includes XML tags for all options. Certain options, such as machine tags will include values defined within percent (%%) characters that can be searched for and replaced. For example:

<bindings>

<binding name="">

<machine>%%demo2sub.aar-machine%%</machine>

<product>

<type>adb</type>

<version/>

<location/>

</product>

<description/>

<contact/>

</binding>

</bindings>

-password Option

Use this option with the -export option to encrypt sensitive data in the exported deployment configuration file. You will be prompted to enter an encryption password. For example:

AppManage -export -out c:\ears\deployments\timer\_wait.xml -app timer\_wait -user admin -pw admin -domain tp002 -password

Also use this option with the -deploy or -config option to upload a deployment configuration file whose sensitive data is encrypted with your custom password. You will be prompted to enter the encryption password that you specified when generating the deployment configuration file. For example:

AppManage -config -ear c:\ears\timer\_wait.ear -deployConfig c:\ears\deployments\timer\_wait.xml -app

timer\_wait -user admin -pw admin -domain tp002 -password

-passwordFile Option

Use this option with the -export option to encrypt sensitive data in the exported deployment configuration file using a properties file. The properties file contains the password encrypted using the obfuscate utility (Refer to TIBCO Runtime Agent Installation for more information about Obfuscate Utility). For example:

AppManage -export -out c:\ears\deployments\timer\_wait.xml -app timer\_wait -user admin -pw admin -domain

tp002 -passwordFile c:\my\_password.txt

Also use this option with the -deploy or -config option to upload a deployment configuration file whose sensitive data is encrypted with your custom password. To use this option, you must provide a properties file which contains the password encrypted using the obfuscate utility when generating the deployment configuration file. For example:

TIBCO Runtime Agent Scripting Deployment User’s Guide

26 | Chapter 2 Getting Started

AppManage -config -ear c:\ears\timer\_wait.ear -deployConfig c:\ears\deployments\timer\_wait.xml -app timer\_wait -user admin -pw admin -domain tp002 -passwordFile c:\my\_password.txt

-moveAppData Option

This option allows you to change the transport setting for a given application. It is similar to the -batchMoveAppData option, but operates against a single application. If the application is within a directory, use a forward slash (/) to separate it from the application name. For an overview of this option, see , Changing the Transport for Applications, on page 41.

The -deployconfig option can be used to configure the application with a given XML file.

The following example shows how to change the transport setting for an application from rv (Rendezvous) to local.

AppManage -moveAppData -app SendMsg -user admin -pw admin -domain tp041 -local

-truncate Option

With this option you can remove unwanted revisions of an application. Note that this option does not change the value of Max Deployment Revision for your application. The following example truncates the application’s revision history.

AppManage -truncate -app <app> -domain <domain> -user <user> -pw <password>] [-cred <cred>]

-serialize Option

If this option is used, then deployment will be done to one machine at a time. Without this, deployment is done simultaneously to all machines with service instances for the application being deployed.

When deployment is being done to multiple machines and there are contention issues, the use of this flag can alleviate them and greatly speedup the overall deployment process.

Contention issues can be identified by deployment taking many minutes, but not using significant amounts of CPU time on the admin server or target machines.

The following example shows how to use this option in a domain.

AppManage -deploy -app myApp -user user1 -pw user1 -domain test -serialize

TIBCO Runtime Agent Scripting Deployment User’s Guide

Other Options |27

-exportDeployed Option

This option is to be used with -export or -batchExport option. If -exportDeployed is specified, the active deployed configuration is exported. If not, then the current configuration changes that would be picked up by the next deployment are what is exported. If the application is in undeployed status, but was deployed earlier, the last deployed configuration is exported. If the application has never been deployed, the current configuration is exported just as if the flag was not specified.

When used with -export:

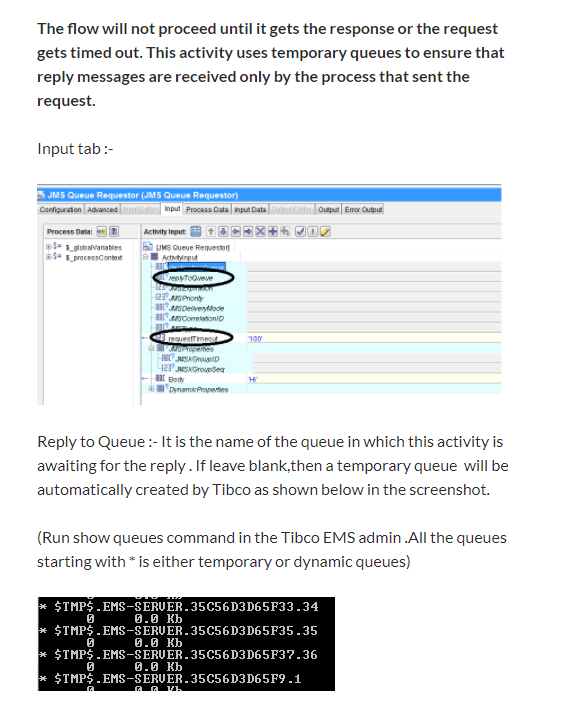
AppManage -export -app <app> -domain <domain> -out <uri> [-user <user> -pw <password>] [-cred <cred>] [-template] [-min] [-genEar] [-ear <archive>] [-exportDeployed]

When used with -batchExport:

AppManage -batchExport -domain <domain> -user <userName> -pw <password> [-cred <cred>] -dir <dir> [-template] [-min] [-noear] [-exportDeployed]



TEMPRORARY QUEUE:



# TIBCO Adapter Service Types

By [Ajmal Abbasi](http://tutorialspedia.com/author/admin/) | November 28, 2013

[3 Comments](http://tutorialspedia.com/tibco-adapter-service-types/#comments)

**TIBCO Adapters** are used as a bridge between other applications and TIBCO Integration environment for exchange of data between them.

There are four **types of adapter services** available which you can configure for your adapter application:

1. Publication Service
2. Subscription Service
3. Request-Response Service
4. Request-Response Invocation Service

**Tibco Adapter Publication Service:**

Adapter publication service is used to publish any event in a vendor application to **TIBCO Enterprise Service Bus (ESB)** to make it available to interested TIBCO applications or processes. For example, you can configure adapter publication service which will be used to publish a message to ESB every time there is an update in a database table. **TIBCO Adapter Publication Service** works in asynchronous mode. You can get a more insight into TIBCO Adapter Publication service by going through [**Tibco Database Adapter: Step By Step Tutorial**](http://tutorialspedia.com/tibco-database-adapter-step-by-step-tutorial/)

**Tibco Adapter Subscription Service:**

Adapter **Subscription service** works opposite to what publication service does. Subscription service is used to subscribe to some event or action in TIBCO environment and makes it available to some external application. For example, you can write a subscription service which will intimate an external application (e.g. Oracle database), every time there is a new HTTP request received in a TIBCO process.

Just like adapter publication service, subscription service also works in asynchronous mode.

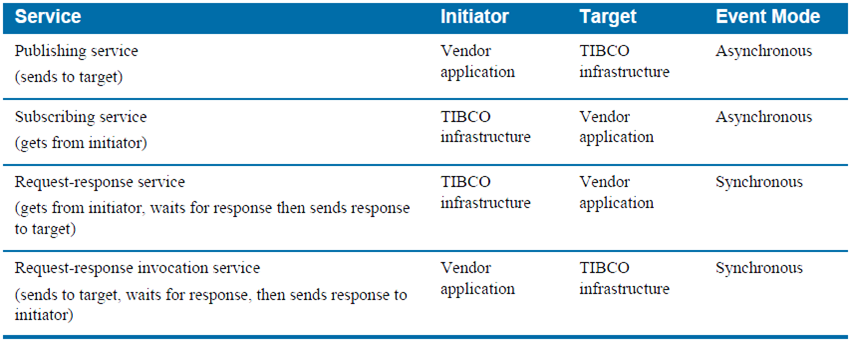
**Tibco Adapter Request-Response Service:**

Adapter Request-Response service has same functionality as a subscription service with the only difference that it works in synchronous mode. Requests are initiated from TIBCO environment and some other application acts as the message target. In this synchronous communication model, target application sends back a response to the initiator (TIBCO environment).

**Tibco Adapter Request-Response Invocation Service:**

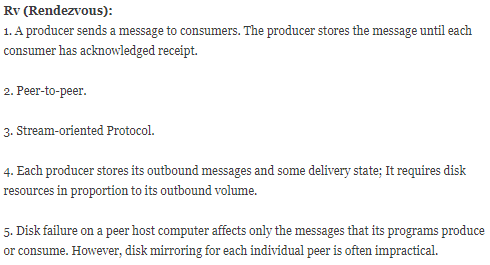
Adapter Request-Response Invocation service has same functionality as Request-Response service with the only difference that target and initiators are reversed. In this type of service, vendor application serves as the initiator and TIBCO environment behaves as target. This type of communication is synchronous as TIBCO environment sends back response synchronously to vendor application for every request.

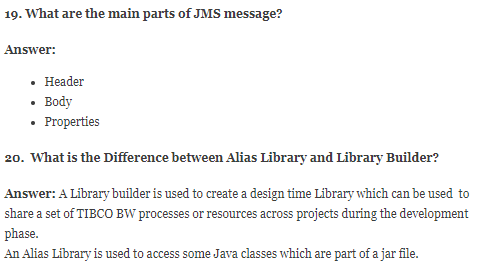
Below chart will help you to understand more about TIBCO Adapter service types and their working.

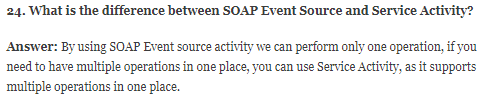


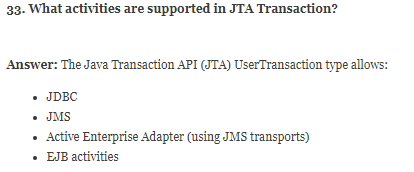
**Q. How can you make only one consumer receives message from the queue?**  
By making the queue exclusive. If the queue is exclusive, then all queue messages can only be retrieved by the first consumer specified for the queue. Exclusive queues are useful when you want only one application to receive messages for a specific queue. If the queue is not exclusive, any number of receivers can retrieve messages from the queue.

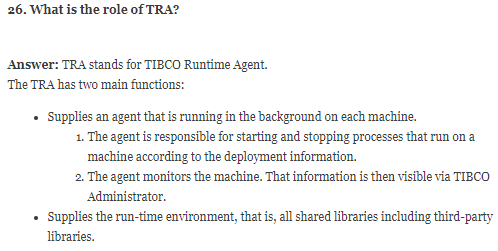
**Q. Why do we need Non-exclusive queues?**  
Non-exclusive queues are useful for balancing the load of incoming messages across multiple receivers. Regardless of whether the queue is exclusive or not, only one consumer can ever retrieve each message that is placed on the queue.











## [**How to perform Tibco bw performance tuning**](https://tibco4all.com/2016/01/25/how-to-perform-tibco-bw-performance-tuning/)

Posted on [January 25, 2016](https://tibco4all.com/2016/01/25/how-to-perform-tibco-bw-performance-tuning/)by [Lijo Ouseph](https://tibco4all.com/author/lijo1866/)

Performance tuning is a main thing while considering the total engine performance of the applications. Below are the best tuning exercise,

#### **Designer parameters**

**HEAP SIZE**: Following parameter used to set the heap size in bwengine.tra, designer.tra files for the engine:

Example: tibco.env.HEAP\_SIZE=256M

These values determine the amount of memory to be used by the engine.

#### **Admin parameters**

**Max Jobs** :Specifies the maximum number of process instances that can concurrently be loaded into memory.  
**Use Activation Limit**:Specifies that once a process instance is loaded, it must remain in memory until it completes.  
**Flow Limit**: Specifies the maximum number of currently running process instance to start before suspending the process starter.

**Include and Import in xsd:**

Import –It used to use or refer elements from other .xsd with different namespace

Include –It used to use or refer elements from other .xsd with same namespace



**Namespace and targetNamespace**

**TargetNameSpace-** it defines the .xsd identity whenever we create .xsd then we mention the targetNameSpace

**NameSpace-** This Namespace is like pakages in java in order to resolve conflict issues. Let us say both .xml has element <table> then we combine it will be conflict.

In order to that we will menitone as xmlns:<<prefixName>>

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"

xmlns:eucn=<http://www.tibco.com/schemas/LearningProject/JMS-Queue/Schema.xsd> // NameSpace

targetNamespace=<http://www.tibco.com/schemas/LearningProject/JMS-Queue/Schema.xsd> // TargetNamespace

elementFormDefault="qualified"

attributeFormDefault="unqualified">

<xs:element name="root" type="xs:string"/>

<xs:element name="cfcf" type="xs:anyType"/>

</xs:schema>

<h:table xmlns:h="http://www.w3.org/TR/html4/">  
  <h:tr>  
    <h:td>Apples</h:td>  
    <h:td>Bananas</h:td>  
  </h:tr>  
</h:table>

**What is Attribute and what is element?**

<book category="children">  
    <title>Harry Potter</title>  
    <author>J K. Rowling</author>  
    <year>2005</year>  
    <price>29.99</price>  
  </book>

Except “Category” all other are Elements. Category is Attribute . Only Attribute value will be quotted in double quotes.

**Main Usuage of Checkpoint:**

bw.engine.enableJobRecovery=TRUE save it in properties,cfg

bw.engine.ShutdownOnStarUp

No Checkpoint in Transaction Group

Backend files with XJOB Type is created

EngineCommand GetRecoverableProcess - Command to Execute

EngineCommand RestartRecoverableProcess - Command to Execute Loop it in Iterate ( Mapping the ProcessId from the Output of the 1st Output Engine Command

In Order to find duplication map the duplicate key input elements with the body of the JMS Receiver

So data will be saved with JObId:key

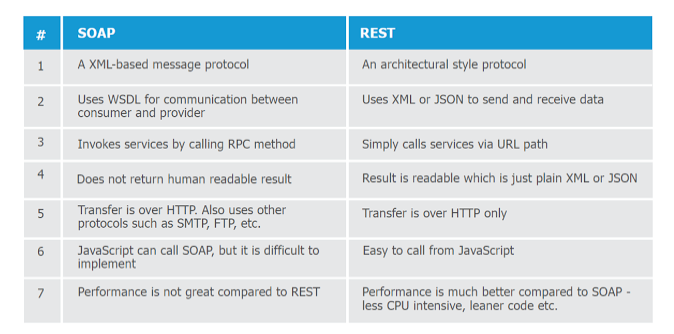
Once 101:ERA is executed and again if we are receiving 104:ERA. Then error will be thrown

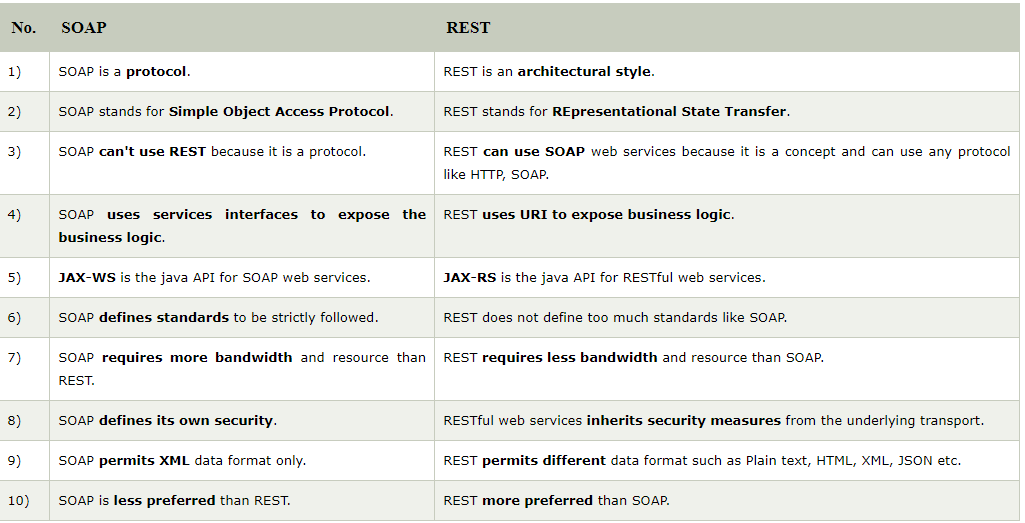
**SOAP over HTTP:**

Firewall friendly is that all services support this transport  
Supported on all platforms (easiest connectivity in b2b scenario)  
Clients can be simple and lightweight  
It is not always reliable. No guaranteed delivery.

**SOAP over JMS:**

High-volume distributed messaging  
Asynchronous messaging  
Where a transaction boundary is needed in the middleware  
Where the message consumers are slower than the producers  
Guaranteed deliver and/or only once delivery of messages  
Publish/subscribe  
provides guaranteed delivery using either persistent queues or durable topics.





## Why REST is better than SOAP?

Now that you know some differences between REST and SOAP web services, let's summarize our reasons of *why REST is better choice* for modern day web service requirement :  
  
1. REST can be consumed by any client  e.g. Java, C++, Python client and even a web browser with Ajax and JavaScript.  
  
2. REST is lightweight as compared to SOAP, it doesn't require CPU consuming XML parsing and it also consumes less bandwidth because unlike SOAP, REST doesn't require a SOAP header for every message.  
  
3. SOAP is an old technology, all modern technical giant are using REST e.g. Google, Twitter, and Flickr.  
  
4. REST is easy to learn, its just nouns and verbs. If you already know HTTP methods then its even easier.  
  
5. Java has excellent support for RESTFul web services, well it also has good support for SOAP web services but you have lots of choices here e.g. Jersey, RESTLet etc.  
  
  
Read more: <http://javarevisited.blogspot.com/2015/08/difference-between-soap-and-restfull-webservice-java.html#ixzz4wl8T6J5Z>

**Max Jobs:**

If the number of process instances in memory have reached the value of Max Jobs, then the process instances created are temporarily held on a disk. These process instances will be moved back into memory when sufficient memory is available.

**Flow Limit**:

When set, this property limits the number of process instances that can be created. If the number of process instances being created exceeds the value of FlowLimit, the engine suspends the creation of new process instances. However, it continues executing the process instances in memory. The engine resumes creating new process instances when process instances, approximately half the value specified for FlowLimit, have completed.

The number of process instances that can be created in memory is also limited by the memory available on the machine and the memory allocated to the JVM on which the process engine executes.

**ThreadCount:**

The process instances in memory are executed by the BusinessWorks engine. The number of process instances that can be executed concurrently by the engine is limited by the maximum number of threads, specified by property ThreadCount. Threads execute a finite number of tasks or activities uninterrupted and then yield to the next process instance that is ready.

**StepCount:**

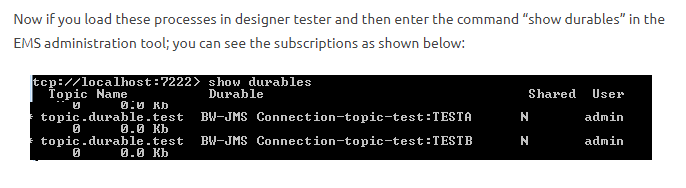
The engine property StepCount determines the number of tasks that are executed by a thread uninterruptedly. However, the exceptions to StepCount occur when the job is blocked or in a transaction. When a job is in a transaction, the thread will not be released even if the StepCount is exceeded. However, if a process instance is waiting or is in a blocked state, it can be paged out and the freed memory used to process another process instance.

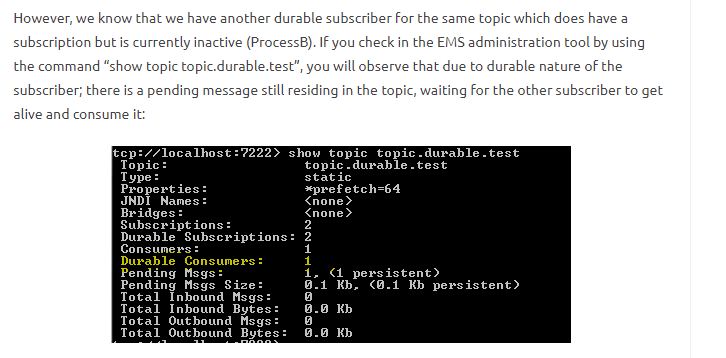
**Activation Limit:**

Activation Limit can be set if a process instance that is blocked should remain in memory till completion. Setting the ActivationLimit affects the engine performance substantially.

**What is Durable Subscribers in Topic?**

When you check the checkbox durable subscribers in Topic Subscriber. Then that receiver wil receive the msg if it is alive. If not alive EMS Server will save the msg and redeliver the msg once again to the durble subscriber once the receiver is avaibale to receive the msg.





**FlowLimit/ActivationLimit/MaxJobs:**

***Max Jobs:***

Maximum Number of jobs that can execute in memory at any time.  
Rest of the jobs will be paged i.e. stored to disk.

***Flow Limit:***

Is the total number of jobs that can be created i.e. Paged  
Jobs + Max Jobs. Once this limit is reached starter activities will be  
suspended.

***Activation Limit :***

If checked, it means that once a job is being  
executed it will not paged out. Normally when a job is being executed  
and come across a wait/sleep kind of activities it is paged out by the  
bwengine. When the event arrives, they are put back in memory and  
execution starts again. However if this is checked, it will ensure that  
job will be kept in memory from start to until it is completed.

if your Max Job is 10 then in ideal case your Flow Limit should be 20

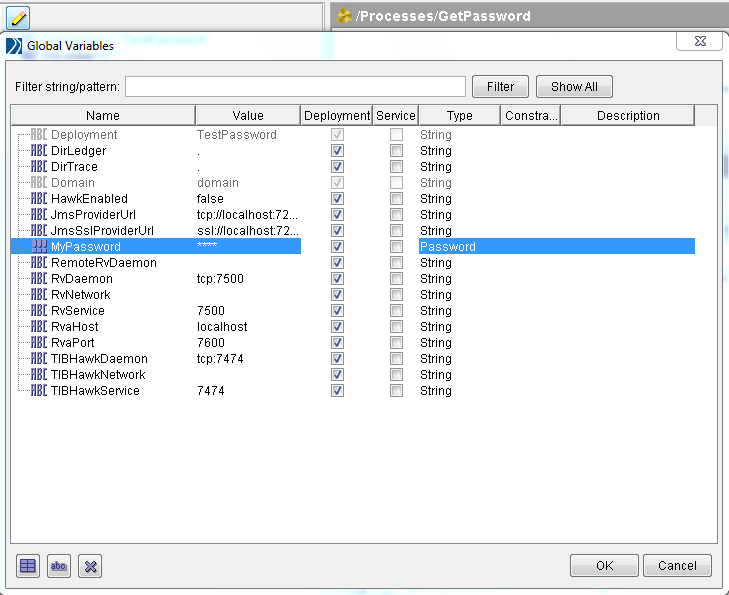
If MaxJob =100 FlowLimit=200 ActivationLimit=Enabled, Only 100 Jobs can be created and If we try to create more than 100 then rest of the jobs will be paged to the disk.

In Total we can have only 200 Jobs including paged jobs + jobs in memory.

If the paged Jobs + jobs in memory =FlowLimit then starter activity will be suspended. No jobs can be created. Then the starter activity will be resumed only after when paged Jobs + jobs in memory is half of the FlowLimit Value.

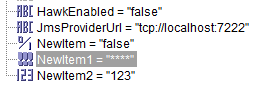
**Global Variable :-**

1. <http://tibco-tutorial-en.blogspot.in/2015/10/tibco-businessworks-512-global.html>
2. <http://tibcobusinessworksonlinetraining.blogspot.in/p/tibco-faqs.html> (Interview Questions)
3. Sequence Key <https://tibcoguru.org/2016/02/22/sequencing-key-in-tibco-bw/>



**Global Variable Attributes :-**

* **Name – Provide Name for the variable.**
* **Value – Provide Value for the variable.**
* **Deployment – Select the Deployment checkbox to make the variable visible when deploying Tibco Administrator. If that checkbox is clear, the variable will not visible in Tibco Adminstrator.**
* **Service – Select the Service level checkbox to make the variable as .par level.**
* **Type – In Tibco global variable supports 4 datatypes. These are String, Integer, Boolean and Password.**
* **Constraint – For String and Integer types, provide a range of allowed values like enumeration. (ex – for string – one, two, three) (ex – for Integer – 1 – 100).**
* **Description – Provide some description for that variable.**



**Service CheckBox:**

When you select the Service checkbox that means you can change the global variable value at service instance level. To make it more simple, suppose if you have an application which you wanted to deploy on two different machine and there is a global variable parameter which will be different for each machine. in that case if you have selected the service checkbox then you can change the value at service instance level otherwise cannot.

Please have a look at this structure of Deployment:

Application can have at least one Service  , Service can have One or more Service Instances

Application  -> EAR

|\_ Service+  -> Process Archive

|     |\_ Service Instance \*

|\_ Service+

     |\_ Service Instance \*

Global variables on EAR level are shared across all services and all service instances.

In Designer Global Variable editor , there are two checkboxes (Deployment , Service)

If you mark Deployment, this means that the Global Variable will be available only at Application level and its value is shared across all Services and Services Instances

If you mark both Deployment and Service checkboxes, then the GV will be available on both the Applicaiton and Service levels

The value you define at the Application level will be the default for all Services , but you can override it on Service  Instances level, and this is the difference.

So having the gloval variable on Service level gives you the ability to change it per Service Instance

an Example:

You have defined a global variable "Hostname" which will hold the hostname of the machine where the instance is running.

Deployment time, you need to run two instances of the same Service on two different machines. Then deployment will be as following:

Application  -> GV.Hostname

|\_ Service   ->  GV.Hostname

     |\_\_ Service Instance1  -> Machine A

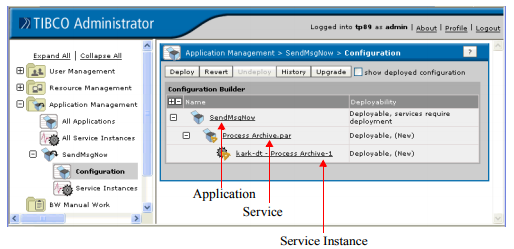
     |\_\_ Service Instance2 -> Machine B

you will over ride the value of Hostname for each Service Instance and the values will be as folliwing

ServiceInstance1.GV.Hostname = "MachineA"

ServiceInstance2.GV.Hostname = "MachineB"

Please let me know if you need any more illustration



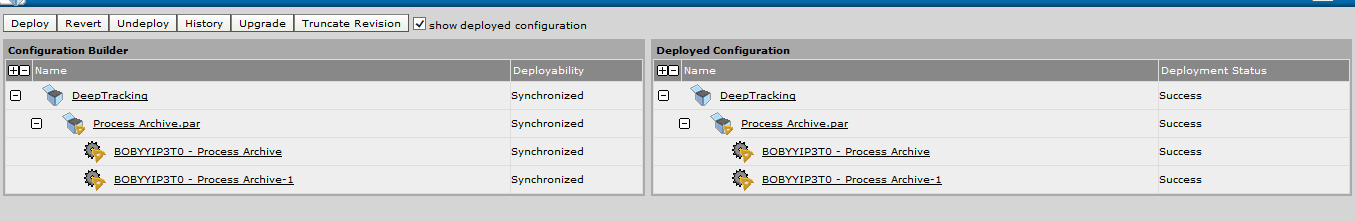
**Use of Service Check Instance level GV configuration can be done. Only the GV service checkbox is checked will be able to configure at Service and Service Instance Level.**

**Application Level: It will hold all the GV’s both deployment and service checkbox enabled.**

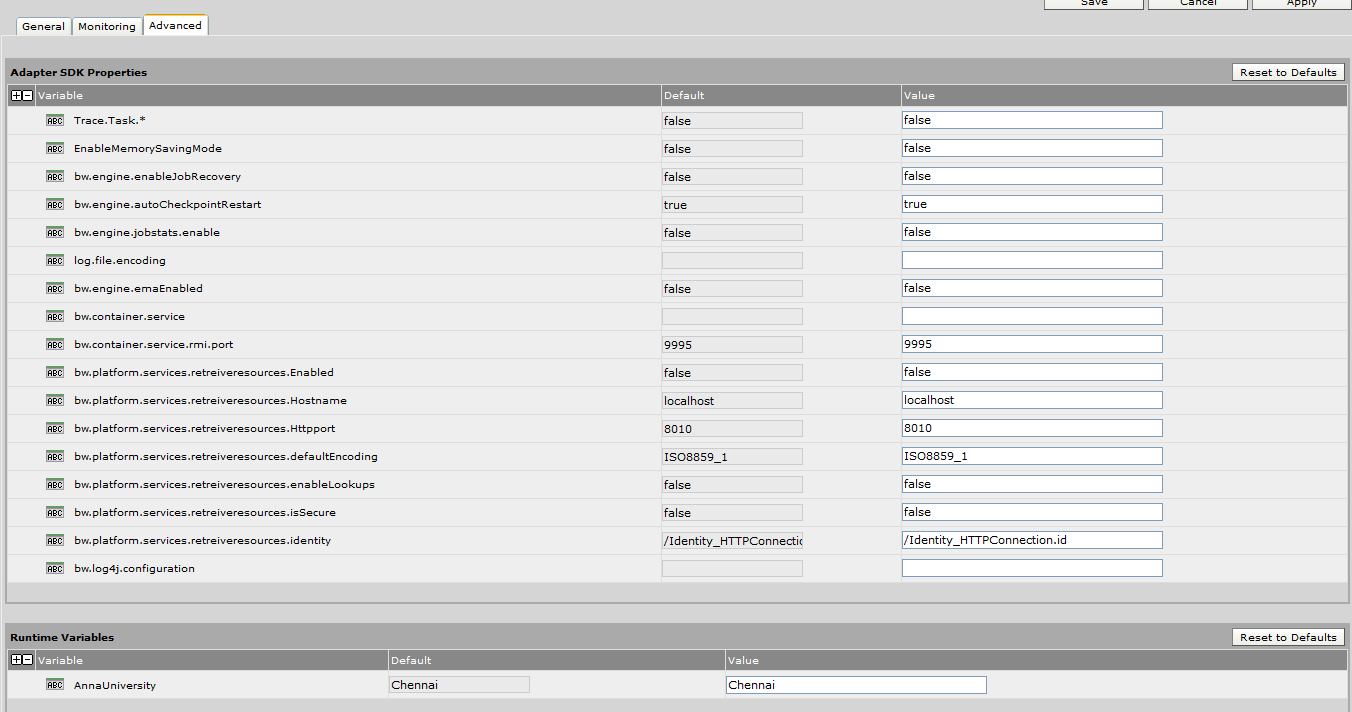
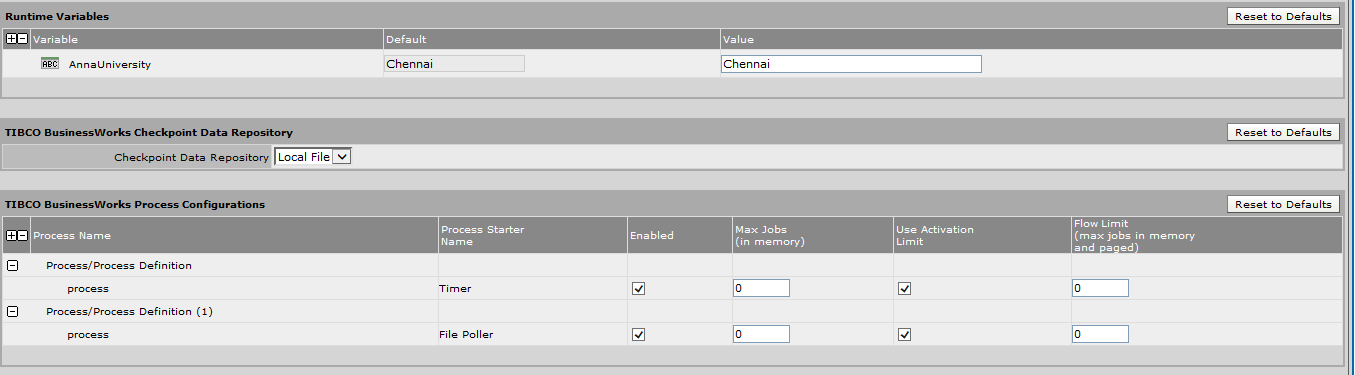
**ServiceInstance level & Service Level: Only Service Enabled GV’s**

**Service Instance Level:**

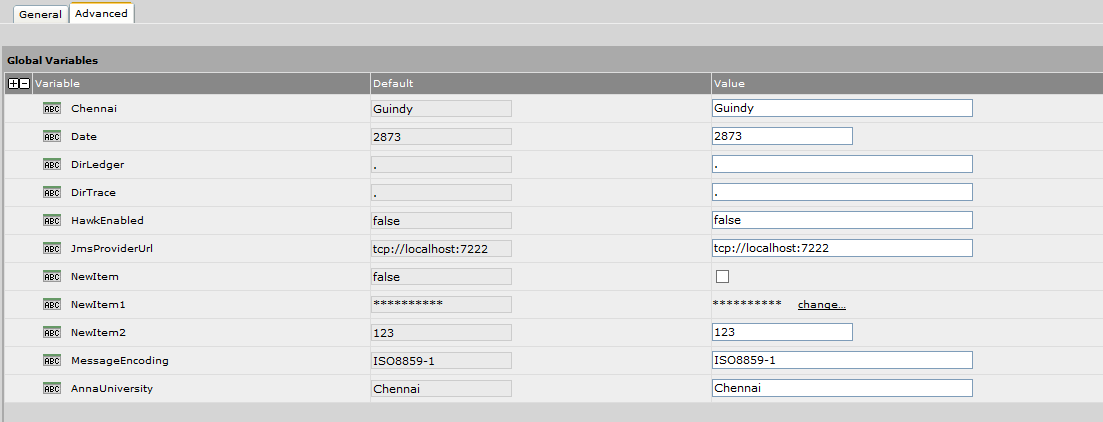




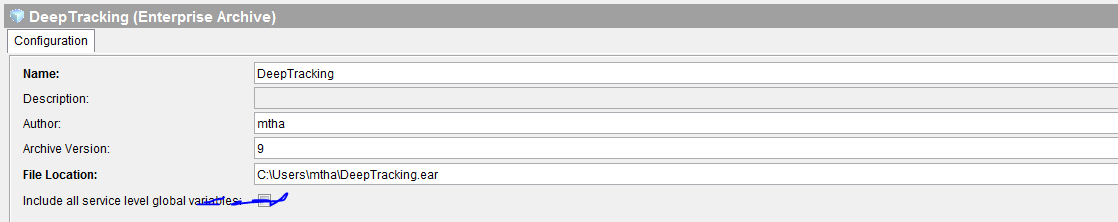
**Service Level:**

Application level:



|  |  |
| --- | --- |
| Include all service level global variables | Includes all global variables for which you clicked the Service check box.  N**ote:** This checkbox is provided for some adapters that do not properly report all their properties. TIBCO Designer cannot tell if that adapter is using a service level global variables or not. This check box explicitly forces inclusion of all service-level variables. |



How to correlate EMS messages in a request response scenario

[TIBCO BusinessWorks™](https://community.tibco.com/products/tibco-activematrix-businessworks)[TIBCO Enterprise Message Service™](https://community.tibco.com/products/tibco-enterprise-message-service)

###### **By:**

[mepatel\_38653](https://community.tibco.com/users/mepatel38653)

Last updated:

2:04pm Dec 08, 2016

[Flag as Inappropriate](https://community.tibco.com/user/login?destination=node/52656)

[Edit This PageCreate New Page](https://community.tibco.com/user/login/?destination=wiki/how-correlate-ems-messages-request-response-scenario)

Table of Contents

* [Case 1](https://community.tibco.com/wiki/how-correlate-ems-messages-request-response-scenario#toc-2)
* [Case 2](https://community.tibco.com/wiki/how-correlate-ems-messages-request-response-scenario#toc-3)
* [Case 3](https://community.tibco.com/wiki/how-correlate-ems-messages-request-response-scenario#toc-4)
* [Document References](https://community.tibco.com/wiki/how-correlate-ems-messages-request-response-scenario#toc-5)
* [Troubleshooting](https://community.tibco.com/wiki/how-correlate-ems-messages-request-response-scenario#toc-6)
* [Information to be sent to TIBCO Support](https://community.tibco.com/wiki/how-correlate-ems-messages-request-response-scenario#toc-7)

### **Case 1**

Consider the scenario where you are using a JMS Queue Requestor which sends a request and waits for a reply. Additionally, you have a corresponding process (say a JMSQueue Receiver) which receives these requests and sends back replies (Reply To JMS Message).

The JMS request/reply activity uses temporary destinations to ensure that reply messages are received only by the process instance that sent the request. While sending each request the JMS Queue Requestor creates a temporary queue for the reply. It then sends the temporary reply queue name along with the request message. The temporary queue name is unique for each process instance.

If the replyToQueue queue (static) is specified then all replies will be sent to the same queue and there will be no guarantee that the correct reply will be received by the process instance that sent the request.

You can use an expression for the replyToQueue to create different replyToDestinations for each request.

### **Case 2**

In Case1, if you need to use constant destinations for all replies and you do not want to use temporary destinations, then instead of using JMSQueueRequestor you need to do the following procedure:

1. use a pair of "JMSQueueSender" and "Wait for JMSQueueMessage" activities
2. map the messageID of the JMSSender as the event key of the "Wait for JMS" activity
3. use the JMSCorrelationID header of the input message as the Candidate Event Key

### **Case 3**

In a multi-engine environment, where you have multiple "Wait for JMS Message" activities listening on the same queue for reply messages, you should consider using GetJMSQueue Message.

In a multi-engine environment, with multiple “Wait For” activities listening on the same queue, it is likely that the first requestor will be waiting for a reply it will never receive as the second requestor has already consumed the reply message. Since the candidate event key does not match the incoming message’s event key the message is discarded. In this case the first requestor which sent out the request will never receive the reply.

This is the default behaviour of “Wait For” activities. When using “Wait For” JMS message activities, a listener consumes all messages from the queue at engine startup and stores them in process memory. In the case of multiple “Wait For” activities listening on the same queue, if one listener has already consumed the message, the other listener on the same queue will never receive the message.

The correct design would be to use the “Get JMS Message” activity instead of the “Wait For JMS” activity. You can set the "selector" property of the "Get JMS Queue Message" activity to use the following XPath to correlate the request message with its reply message.

concat("JMSCorrelationID = '"

,$JMS-Queue-ender/pfx:aEmptyOutputClass/pfx:MessageID,"'")

When using a message selector, the EMS server does the filtering of the message based on the selector and determines if the message can be delivered to the particular "GetJMSQueue Message" activity.

Whereas with the “Wait for JMS" activity, the message is sent to the queue as soon as it arrives on the queue and the filtering is done at the job level where the Candidate Event key is matched with the incoming message’s event key.

### **Document References**

For details, please refer to the following TIBCO ActiveMatrix BusinessWorks™ documentation:

[Palette Reference](https://docs.tibco.com/pub/activematrix_businessworks/6.2.2/doc/html/GUID-83119708-D92E-4F52-A97A-D2B264D3CE11.html) --> [Chapter 9 JMS Palette](https://docs.tibco.com/pub/activematrix_businessworks/6.2.2/doc/html/GUID-E4C7F5B7-F4D1-405C-AB35-1673E8552E17.html)

### **Troubleshooting**

* If the correct replies are not received, review the process design.

You can connect to **tibemsadmin** tool and check for the number of receivers on the queue by using

Show queue

You can enable tracing for message IDs and correlation IDs in the tibemsd.conf

track\_message\_ids = enabled

track\_correlation\_ids = enabled

* Additionally, you can turn on detailed tracing for both EMS server and client as follows
  1. set server log\_trace=DEFAULT,+PRODCONS,+MSG
  2. set server client\_trace=enabled
  3. Addprop queue trace=body (\* For both the request and reply queues.)

Then check the messages that are sent by the server and received by the client.

### **Information to be sent to TIBCO Support**

1. Confirm the Admin/TRA/BW/EMS versions with hotfixes, if any.
2. Please send the multi-file project and the deployed .ear file.
3. EMS configuration files.
4. Other output of EMS admin commands as and when requested by TIBCO Support.

|  |  |  |  |
| --- | --- | --- | --- |
| *Table 4**Effects of various configuration settings* | | | |
| **Max Jobs** | **Activation Limit** | **Max Jobs (Flow Limit)** | **Description** |
| 0 | Cleared or selected | 0 | An unlimited number of process instances can be created and concurrently loaded into memory.  Activation Limit is ignored when Max Jobs is set to 0. |
| 0 | Cleared or selected | N | No paging of process instances. Allows up to N process instances before placing process starter in flow controlled stated.  Activation Limit is ignored when Max Jobs is set to 0. |
| 1 | Selected | N | One process instance is loaded into memory at a time and kept there until it completes its execution. **This guarantees incoming events are processed in the order in which they occur**. Up to N process instances are paged to disk, and then the process starter is placed into flow controlled state.  **Note:** If your goal is to sequentially process incoming events, use the Sequencing Key field on the Misc tab of the process starter. Using Max Jobs and Activation Limit incurs overhead as process instances are paged to disk and retrieved from disk. |
| 1 | Selected | 0 | Once process instance is loaded into memory at a time and kept there until it completes its execution**. This guarantees incoming events are processed in the order in which they occur.** There is no limit on the number of process instances that can be created and paged to disk.  **Note:** If your goal is to sequentially process incoming events, use the Sequencing Key field on the Misc tab of the process starter. Using Max Jobs and Activation Limit incurs overhead as process instances are paged to disk and retrieved from disk. |
| 1 | Cleared | N | One process instance is loaded into memory at a time, but up to N process instances are created. Incoming events can be processed in any order because process instances are not kept in memory until they complete execution. |
| M | Selected | 0 | An unlimited number of process instances can be created, but only M are loaded into memory and processed concurrently.  This setting ensures a limited amount of concurrent processing. This situation is useful if you have limited resources, such as database connections. You can set Max Jobs to a relatively small number and the Activation Limit option keeps each service in memory until the service completes. Each loaded process uses a machine resource until the service completes. Once a service releases the resource, a new process can be loaded into memory and the corresponding service can use the resource. |
| N | Same as above, except only N process instances are created before the process engine is placed in the flow controlled state. |
| M | Cleared | 0 | An unlimited number of process instances can be created, but only M are loaded into memory and processed concurrently. After M process instances are created, new process instances are paged to disk. There is no guarantee of the order in which process instances are executed. |
| N | Same as above, except only N process instances are created before the process engine is placed in the flow controlled state. |

**When we should go for XA Transaction?**

1.  All parties to a JDBC transaction must be JDBC activities.  Thus a JDBC transaction cannot include a JMS activity, however an XA transaction can.

2.  All parties to a JDBC transaction must use the same JDBC connection.  Thus activities taking place against different databases cannot be a part of the same JDBC transaction, although they could if they were using an XA transaction.

3.  XA transactions require a separate XA transaction manager to coordinate the transaction.  With JDBC transactions as long as you have BW and a database you're good.

There is a nice summary of transactions in the BW Process Design Guide.

JDBC Transaction- All JDBC Palettes should use same connection.If different connection each connection will be separate transaction

XA Transaction- Allow JMS Activity. JDBC Acitvity using different connection are also considered as single transaction

**JDBC**

The JDBC transaction allows multiple JDBC activities that access the same database connection to participate in a transaction. Only JDBC activities that use the same JDBC Connection participate in this transaction type, but other activities can be part of the transaction group. If the transaction commits, all JDBC activities using the same JDBC connection in the transaction group commit. If the transaction rolls back, all JDBC activities using the same JDBC connection in the transaction group roll back.

The transaction group commits automatically if all activities in the group complete and a non-error transition is taken out of the transaction group. If any errors occur while processing the activities in the group, even errors in non-JDBC activities, the transaction is rolled back and an error is returned (you should have an error transition out of the group to handle this situation).

Individual JDBC activities can override the default transaction behavior and commit separately. See the description of the JDBC palette in *TIBCO ActiveMatrix BusinessWorks Palette Reference* for more information about using JDBC activities.

**Multiple JDBC Connections In Transaction Groups**

All activities that use the same JDBC Connection shared configuration resource are part of the same transaction. It is possible to use more than one JDBC Connection in the same transaction group. However, only activities that use the same JDBC Connection are guaranteed to commit or rollback together when the transaction completes.

If you have more than one JDBC Connection in the transaction group, each set of activities that uses a JDBC Connection is considered a separate transaction. For example, you have three JDBC Updates in a transaction group, A, B, and C. A and B use JDBC Connection X, but C uses JDBC Connection Y. In this case, the updates for activities A and B are part of one transaction and the update for activity C is part of a different transaction.

To create a distributed transaction across multiple databases, use the XA transaction type.

**Configuring JDBC Transactions**

To configure a JDBC transaction, select JDBC Transaction as the transaction type of the group. Also, in the JDBC Connection resource(s) used by JDBC and Checkpoint activities in the group, select JDBC in the Connection Type field.

**How to enable SOAP Trace?**

Add the following properties in ***designer.tra***

java.property.com.tibco.plugin.soap.trace.inbound=true

java.property.com.tibco.plugin.soap.trace.outbound=true

java.property.com.tibco.plugin.soap.trace.filename=c:/temp/soap.txt

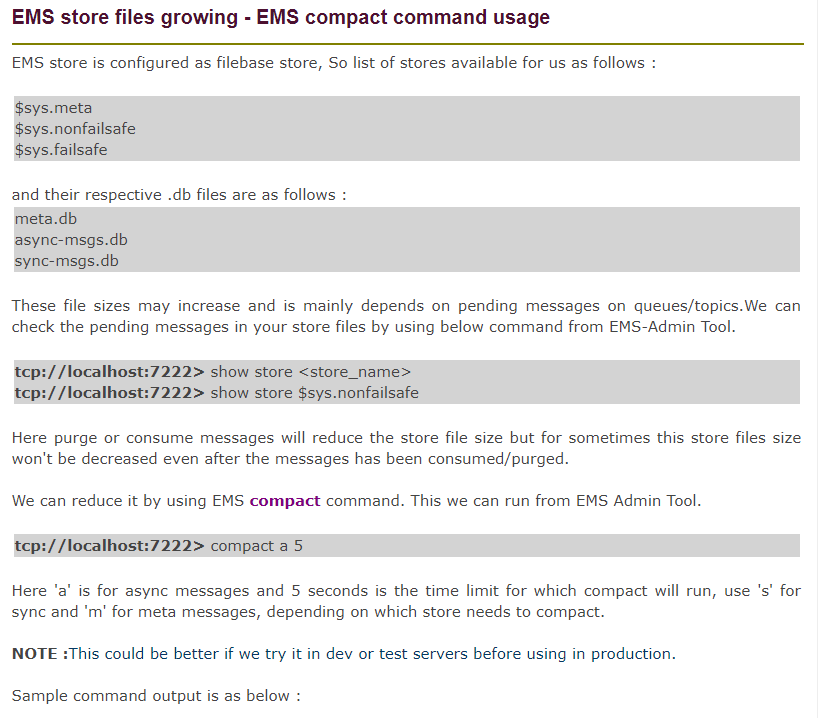
java.property.com.tibco.plugin.soap.trace.pretty=true

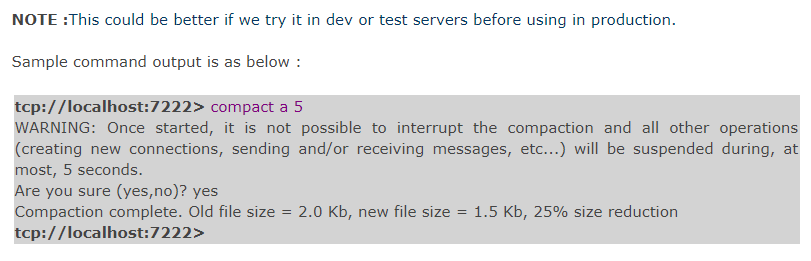
For "by passing" the parsing , Plaes add :

java.property.com.tibco.plugin.soap.SOAPSendReceiveActivity.NoOutputValidation=true

java.property.com.tibco.plugin.soap.SOAPSendReceiveActivity.NoInputValidation

EMS Compact Command Use:





# HOW TO REDUCE THE SIZE OF EMS DATA STORE FILES?

Posted on [March 20, 2017](http://www.techieswiki.com/how-to-reduce-the-size-of-ems-data-store-files.html) by [techieswiki](http://www.techieswiki.com/author/techieswiki)

EMS store is configured as file base store, so it will have below file based Databases corresponding to each store.

|  |  |
| --- | --- |
| DB | Store |
| Meta.db | $sys.meta |
| Async-msgs.db | $sys.nonfailsafe |
| Sync-msgs.db | $sys.failsafe |

Reason of store file size increase is mainly depends upon pending messages on queues/topics. There is a bug in EMS where once the store size increases beyond a certain limit due to pending messages it is never able to reduce its size even after the messages have been consumed or purged.  
Use the below command in EMS Server tool to check pending messages in store files.

syntax : show store [store\_name]

show store $sys.failsafe.

This would give you message count stored in $sys.failsafe

Use COMPACT command to reduce the size of store files.

Syntax: compact [store type] [max time]

**Store types**:

asynchronous or async or a for asynchronous storage  
synchronous or sync or s for synchronous storage  
meta or m for meta storage

**max time**:  
upper limit in seconds on compaction time(0 for specifying no upper limit)

compact sync 5 /\*this command will run for 5 sec\*/

**Output:** Compaction complete. Old file size = 23 GB, new file size = 1.57 GB, 98% size reduction  
NOTE: Once started, it is not possible to interrupt the compaction and all other operations (creating new connections, sending and/or receiving messages, etc…) will be suspended during, at most, 5 seconds.

NOTE:Run this command in EMS server when less amount of messages are there in queues/topics.

**Questions:**

### **What Is The Critical Section In Tibco Business Works And What Is It Used For?**

**Answer :**

Critical section is used to sync process instances so that only once process instance executes the grouped activities at any given time. Any concurrently running process instances that contain a corresponding critical section group wait until the process instance that is currently executing the critical section group completes. Particularly used for controlling access to shared variables. Critical section is one of the group actions. Others are iterating, repeat until true, repeat on error until true, while true, pick first and transaction. If we select critical section as the group, then we get two options. Single group and multiple groups.

**Single Group**

**If you wish to synchronize process instances for a single process definition in a single process engine (only one ear(instance) is deployed), perform the following:**

* 1. Create a group around the activities you wish to synchronize.
  2. Specify Critical Section for the Group Action field.
  3. Specify Single Group for the Scope field.
  4. Only one process instance at any given time will execute the activities contained in the Critical Section group.

**Multiple Groups:**

**If you wish to synchronize process instances for multiple process definitions, or if you wish to synchronize process instances across multiple process engines, perform the following:**

* 1. Create a Lock shared configuration resource and specify a name for the resource.
  2. To perform the synchronization across multiple process engines, check the Multi-Engine field of the Lock resource. When the process instances are executed by the same process engine, locking is performed in memory. When the process instances are executed across multiple engines, the process engines must be configured to use a database for storage, and a database transaction is performed to ensure that only one process instance is executing the critical section group at any given time.
  3. Create a group around the activities you wish to synchronize.
  4. Specify Critical Section for the Group Action Field.
  5. Specify Multiple Groups for the Scope field.
  6. Use the Browse button in the Lock Object field to locate the Lock shared configuration resource you created in Step 1.
  7. Perform steps 3 to 6 for any process definitions you wish to synchronize. Make sure you specify the same Lock shared configuration object for all Critical Section groups.

### **If There Are Three Activities In The Critical Section, Jdbc Update Then A Write File And Then Again Jdbc Update. If The Second Jdbc Update Fails, Then What Will Happen To The File Written By Write File Activity?**

**Answer :**

The contents of the file will persist.

### **What Are Different Types Of Transaction Groups Supported In Tibco Bw?**

**Answer :**

Iterate, repeat until true, repeat on error until true, critical section, transaction, pick first, while true.

### **If The Message Has Hit The Queue, And There Are No Receivers For It, And The Receiver Comes Alive After 6 Hours, What Happens To The Message?**

**Answer :**

Message stays on the queue.

### **If There Are Five Messages And Prefetch Property Is Set To 4 Then What Happens?**

**Answer :**

Fetches 4 messages

### **What All Palettes You Have Worked In Tibco Bw?**

**Answer :**

File palette, generall activities palette, jdbc palette, http palette, jms palette, parse palette, service palette, soap palette, wsdl palette.

### **What Are The Different Types Of Variables Available In Tibco?**

**Answer :**

**There are three types of variables available in TIBCO. They are:**

**1.Shared variable**

**2.Job Shared Variable**

**3.Local Variable**

**4.Process Variable**

**5.Error Variable**

**6.Global variables**

**Global Variables:**Global variables are used for assigning constants a value which is used over a project.

**Process variables:** Process variables are again of four types.

**They are:**

* 1. Activity output
  2. Predefined process variables
  3. Error variables
  4. User defined process variables.
  5. Predefined process variables include $\_GlobalVarialbes and $\_ProcessContext
  6. shared variables – User defined process variables could be defined at a process definition level and assigned a value by using an assign activity.

### **What Are The Different Acknowledgment Modes For A Message?**

**Answer :**

**There are 6 modes of acknowledgement. They are:**

* 1. **Auto:**message is acknowledged automatically
  2. **Client:** use confirm activity for acknowledgement
  3. **TIBCO EMS Explicit:** use confirm activity for acknowledgement, only available for TIBCO EMS
  4. TIBCO EMS no ack
  5. **Dups ok:**message is acknowledged automatically on its receipt
  6. **Transactional:** when a transaction is included in a process definition. The message is ack when the transaction commits.

### **What Is Prefetch And Failsafe?**

**Answer :**

Prefetch is fetching the messages from the server before receiver calls. Failsafe property enables to write persistence messages to the file with synchronous i/o calls.

### **How Do You Send A Response From A Web Service?**

**Answer :**

We write to output to the queue from which we received the request.

### **If You Want To Count The Number Of Times A Job Is Called, How Will You Go About Doing It?**

**Answer :**

Use shared variable in a critical section.

### **If There Are Hundred Messages Lying On The Queue, What Will You Do And Which Activity Will You Use To Retrieve One Message At A Time?**

**Answer :**

JMS Queue receiver with confirm activity. Sequencing key in a misc tab of the configuration could also be used.

### **What Is The Sequencing Key In The Misc Tab Used For?**

**Answer :**

It is used to call process instances in the order they are created.

### **What Happens If I Put A Static Value Say ‘nitin’ In The Sequencing Key?**

**Answer :**

All the processes which have this key in their sequencing key field will be executed in the order they were created.

### **What Is Sql Direct? What Is It Used For?**

**Answer :**

SQL direct is used to execute command dynamically using output of other activities. This activity allows you to execute commands which other activities in the JDBC don’t allow, like DDL command create table.

### **Can You Change A Value Of A Global Variable At Runtime?**

**Answer :**

You can change the value of a global variable when you deploy your project in TIBCO Administrator.

See the section on modifying runtime variables in TIBCO BusinessWorks Administration for more information on using TIBCO Administrator.

You can also specify values for global variables when starting a process engine on the command line. To do this, specify the following as a command line argument when starting the process engine:

**-tibco.clientVar.<variablePathAndName> <value>**

where variablePathAndName is the name of the variable you wish to set, including the path to the variable if it is contained in a folder. Value is the value you wish to set the variable to.

**For example:** if you have a global variable named item1 contained in a folder named myGroup and you wish to set its value to 500, add the following argument to the command line when starting the process engine:

**-tibco.clientVar.myGroup/item1 500**

### **What Are The Modes Of Tibco Bw Installations?**

**Answer :**

* 1. GUI mode
  2. Console mode
  3. Silent mode

### **If You Have Installed A Particular Version Of Tibco Software E.g. Tibco Bw X.y.z, What Are X, Y And Z Number Stands For?**

**Answer :**

**Integration can be at different application layers:**

* 1. X:Patch
  2. Y:Major
  3. Z:Minor

### **What Is The Role Of Tra?**

**Answer :**

TRA stands for TIBCO Runtime Agent.

**The TRA has two main functions:**

* 1. Supplies an agent that is running in the background on each machine.
  2. The agent is responsible for starting and stopping processes that run on a machine according to the deployment information.
  3. The agent monitors the machine. That information is then visible via TIBCO Administrator.
  4. Supplies the run-time environment, that is, all shared libraries including third-party libraries.

### **What Are The Resources That Gets Included In The Ear File, Created By The Tibco Designer?**

**Answer :**

An EAR file can contain local project resources, Library Builder resources, and files as specified in Alias Library resources. In addition, the TIBCO Designer class path may include references to other files that are included in the EAR file.

### **What Are The Different Modes Of Service Invocation?**

**Answer :**

Services can be invoked in several ways.

A one-way operation is executed once and does not wait for a response.

A request-response operation is executed once and waits for one response. In a request-response service, communication flows in both directions. The complete interaction consists of two point-to-point messages—a request and a response. The interaction is only considered complete after the response has arrived.

* 1. Publication (notification) means an operation sends information on an as-needed basis, potentially multiple times.
  2. Subscription means incoming information is processed on an as-needed basis, potentially multiple times.

### **What Are The Tibco Bw Activities That Can Participate In Transactions?**

**Answer :**

Not all TIBCO BusinessWorks activities can participate in a transaction.

**Only the following types of activities have transactional capabilities:**

* 1. JDBC activities
  2. JMS activities
  3. ActiveEnterprise Adapter activities that use JMS transports
  4. EJB activities
  5. TIBCO iProcess BusinessWorks Connector activities

### **What Are The Different Types Of Transactions Tibco Provides?**

**Answer :**

TIBCO BusinessWorks offers a variety of types of transactions that can be used in different situations. You can use the type of transaction that suits the needs of your integration project. When you create a transaction group, you must specify the type of transaction.

**TIBCO BusinessWorks supports the following types of transactions:**

* 1. JDBC
  2. Java Transaction API (JTA) UserTransaction
  3. XA Transaction

### **What Activities Are Supported In Jta Transaction?**

**Answer :**

**The Java Transaction API (JTA) UserTransaction type allows:**

* 1. JDBC
  2. JMS
  3. ActiveEnterprise Adapter (using JMS transports)
  4. EJB activities

to participate in transactions.

### **What Are The Options For Configuring Storage For Process Engine’s Checkpoint Repository?**

**Answer :**

**The options for configuring storage for process engine’s checkpoint repository are:**

* 1. Local File
  2. Database. Fault tolerant engines can recover from a checkpoint only when database is used.

### **Process Engines In A Fault Tolerant Group Can Be Configured As Peers Or Master Secondary.how Do These Differ?**

**Answer :**

**The options for configuring storage for process engine’s checkpoint repository are:**

* 1. Peer means all of them have the same weight. In this case when one engine fails another one takes over and continues processing till it fails.
  2. In master secondary configuration weights are unequal, the secondary starts processing when master fails. But when master recovers, secondary stops and master continues processing.

### **What Are The Three Scenarios Where Bw Engine Must Be Configured With Database Persistence Instead Of Local File?**

**Answer :**

**The three scenarios are:**

* 1. Shared Variables across BW engines.
  2. Locking across groups in multiple BW engines.
  3. Wait Notify across BW engines.

### **If You Want A Group To Be Executed If There Is Some Un Handled Error But Subject To Some Max Number Of Iterations Which Group Do You Use?**

**Answer :**

We can use Repeat on Error until true.

### **When Is A ‘generate Error’ Activity Useful?**

**Answer :**

When you handle an error inside a called sub-process or group and want to re-throw the error to the caller (happens by default if you don’t handle the error in the called process)

**What is TIBCO CLE?**

Intelligent Common Logging Error framework (iCLE) addresses these issues by providing a common solution framework for logging and exception handling as enterprise services with a common standard structure format.

**18)** If you want to check the SSL connectivity from designer,  
  
       a) Add the below entries in properties.cfg file and place it in TIBCO\_HOME directory   
           Trace.Task.\*=true                Trace.Startup=true                Trace.JC.\*=true            Trace.Engine=true               Trace.Debug.\*=true            bw.plugin.http.server.debug:true  
      b) In the Designer, click on Tester tab, click on Advanced button and specify following statement at "Test Engine User Args" and click on OK.   
-p TIBCO\_HOME/properties.cfg  
      c) Check the console for logs.  
  
Trace.Task.\*=true

Trace.Startup=true // ENABLING SL TRACE

Trace.JC.\*=true

Trace.Engine=true

Trace.Debug.\*=true

bw.plugin.http.server.debug: true

java.property.TIBCO\_SECURITY\_VENDOR=j2se

java.property.javax.net.debug=ssl,handshake

java.property.https.protocols=SSLv3,TLSv1-TLSv1.1,TLSv1.2

**ssl\_debug(2): Starting handshake (iSaSiLk 3.03)**

**ssl\_debug(2): Received v3 client\_hello handshake message.**

**ssl\_debug(2): Client requested SSL version 3.1, selecting version 3.1.**

**ssl\_debug(2): Client is trying to resume session 79:C9:37:1D:B9:A7:4E:4F...**

**ssl\_debug(2): Resuming previous session...**

**ssl\_debug(2): Sending server\_hello handshake message.**

**ssl\_debug(2): Selecting CipherSuite: SSL\_RSA\_WITH\_3DES\_EDE\_CBC\_SHA**

**ssl\_debug(2): Selecting CompressionMethod: NULL**

**ssl\_debug(2): Sending change\_cipher\_spec message...**

**ssl\_debug(2): Sending finished message...**

**ssl\_debug(2): Received change\_cipher\_spec message.**

**ssl\_debug(2): Received finished message.**

**ssl\_debug(2): Session added to session cache.**

**ssl\_debug(2): Handshake completed, statistics:**

**ssl\_debug(2): Read 131 bytes in 3 records, wrote 114 bytes in 3 records.**

> console.log 2>&1

Trace.Task.\*=true

Trace.Startup=true

Trace.JC.\*=true

Trace.Engine=true

Trace.Debug.\*=true

bw.plugin.http.server.debug: true

java.property.TIBCO\_SECURITY\_VENDOR=j2se

java.property.javax.net.debug=ssl,handshake

java.property.https.protocols=SSLv3,TLSv1-TLSv1.1,TLSv1.2

> console.log 2>&1

keytool -exportcert -alias demoidentity -keystore C:\Users\mtha\Desktop\ROUTER-ESB-Process\_Archive-1\CDIINT\DemoTrust.jks -storepass DemoTrustKeyStorePassPhrase -file PAIRTESTING.cer

**12)** Adding machine to DOMAIN from Command Line  
  
**Step 1:** Go to TIBCO\_HOME/tra/5.7/template/domainutility/cmdline  
**Step 2:** Create a new file by copying the AddMachine.xml to another file name.  
Ex: cp AddMachine.xml AddMachine\_XXX.xml  
**Step 3:** Edit AddMachine\_XXX.xml, enter domain related specific entries.  
**Step 4:** Go to /TIBCO\_HOME/tra/5.7/bin/, execute the following command.  
./domainutilitycmd -cmdFile TIBCO\_HOME/tra/5.7/template/domainutility/cmdline/AddMachine\_XXX.xml –verbose  
**Step 5:** FINISH.  
  
**11)** To change the Global Variables value at runtime, add a property like below in the application tra file and restart the application.  
*tibco.clientVar.Shared/JMSConnection/MaxSessions=16*  
  
**10)** To see the input/output of any acitivity in a perticular BW process at run time, add the following properties in the application tra file and restart the application.  
*Trace.Task.A/B/C/ProcessDefinition.process.Group/ActivityName=true****bw.engine.showInput=true  
bw.engine.showOutput=true***  
Here A/B/C is the directory structure,ActivityName is the acutal activity. If no group is there, remove .Group  
**Note:** It affects the performance of bwengine.  
  
**9)** While using *Notify Configuration* activity, check Local Only option in the configuration tab to allow an in-memory notification when the Wait and Notify activities are performed on the same machine.  
  
**8)** While dealing with DB Connections, use the folllowing property to close the Idle DB connections  
*java.property.bw.engine.dbConnection.idleTimeout=10* (value should be in Minutes)

**How to use terms in interview?**

Architectuural Design Principle -Architecture Style

SOAP

---------

Object Access Protocol

how to access Object if I need to access object

XML Based WebService - used for transferring the message.

->Functionality hosted on the web <<System extending its capability to another application>>

Request and Response we use the

<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope" xmlns:bas="http://www.vodafone.de/ESB/Schema/Common/BaseTypes-ESB-001" xmlns:mfs="http://www.vodafone.de/ESB/Service/EAI/Product/MFSubscriber/MFSubscriber-004">

<soap:Header>

<bas:controllObject>

<!--Optional:-->

<bas:retryObject>

<!--Optional:-->

<bas:timeOut\_s>?</bas:timeOut\_s>

</bas:retryObject>

</bas:controllObject>

<bas:appMonDetailsStrict>

<bas:bpId>?</bas:bpId>

<bas:bpName>?</bas:bpName>

<bas:boId>?</bas:boId>

<bas:callingApp>?</bas:callingApp>

<bas:initiator>?</bas:initiator>

</bas:appMonDetailsStrict>

</soap:Header>

<soap:Body>

<mfs:GetSubscriberInfoRequest>

<mfs:marketCode>?</mfs:marketCode>

<mfs:serialNumber>?</mfs:serialNumber>

</mfs:GetSubscriberInfoRequest>

</soap:Body>

</soap:Envelope>

Network Traffic Issue

Clear way to represent a Object.

Web/Mobile

Validation/Authorization/Authentication

Business

Validation

Tech

ps -erf | grep "hawk"

Stop Admin

Stop Hawk

Start Hawk

Start Admin