DUMPSBOSS.

TIBCO ActiveMatrix BusinessWorks 5

Tibco TB0-123

Version Demo

Total Demo Questions: 13

Total Premium Questions: 138

Buy Premium PDF

https://dumpsboss.co

support@dumpsboss.co

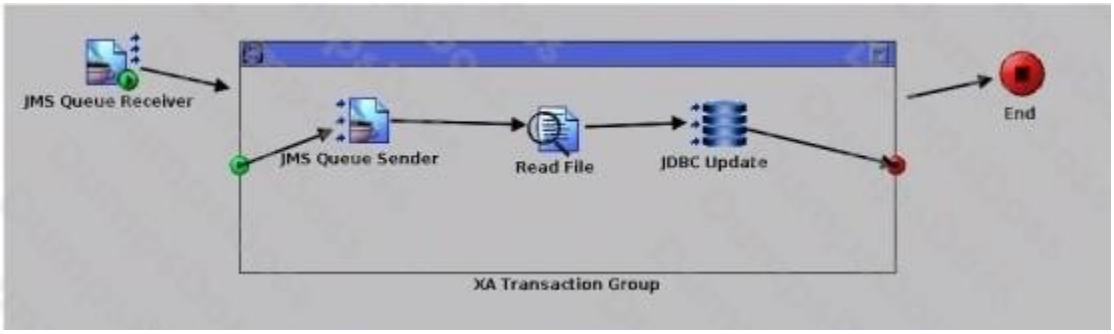
support@dumpsboss.co
dumpsboss.co

Topic Break Down

Topic	No. of Questions
Topic 1, Volume A	63
Topic 2, Volume B	55
Total	138

QUESTION NO: 1

Click the Exhibit button.



The JMS Queue Receiver is part of the transaction group. Which two events occur if the

Read File activity fails? (Choose two.)

A. The transaction rolls back.

B. The message is redelivered.

C. The message is confirmed.

D. The transaction group commits successfully.

E. The JDBC activity rolls back.

ANSWER: A B

Explanation:

The transaction rolls back and the message is redelivered. In ActiveMatrix BusinessWorks 5, a JMS Queue Receiver that participates in a transaction group receives and acknowledges the JMS message within that transaction boundary. Successful completion of the group is required before the JMS receive can be committed and the consumed message can be permanently acknowledged.

If the Read File activity fails, the process instance cannot complete the transaction group successfully. BusinessWorks therefore rolls back the active transaction. Rolling back reverses the transactional work performed by the group and prevents final confirmation of the inbound JMS message. The JMS provider then makes the uncommitted message available for delivery again, subject to the provider's configured redelivery and destination policies. This behavior provides the reliable, once-or-more delivery processing model expected for a transactional JMS consumer: a message is retained until all work in its BusinessWorks transaction has completed successfully.

TIBCO's ActiveMatrix BusinessWorks documentation describes transaction groups and their use with transactional activities, including JMS messaging: TIBCO ActiveMatrix BusinessWorks 5.15 documentation. The JMS transactional model and rollback/redelivery semantics are also defined in the Jakarta Messaging specification.

QUESTION NO: 2

Multiple applications and a TIBCO EMS server are in the same Administrator domain. When using a secondary domain server, which three functions are available? (Choose three.)

A. delete an application

B. create queue or topic

C. delete a queue or topic

D. purge a queue or topic

E. deploy an application

F. undeploy an application

ANSWER: B C D

Explanation:

A secondary TIBCO Administrator domain server provides limited administration capabilities for resources associated with the domain, including administration of a registered TIBCO Enterprise Message Service server. The available messaging-destination operations include **create queue or topic**, **delete a queue or topic**, and **purge a queue or topic**. These functions let an administrator manage the lifecycle of JMS destinations and clear accumulated destination data when operationally necessary, without requiring the primary domain server for those EMS-oriented actions. Creating destinations is used to make queues or topics available to connected applications; deleting destinations removes them when they are no longer required; and purging is an administrative cleanup operation for destination contents. This division of capabilities supports continuity of basic EMS administration when a secondary server is being used, while preserving domain-level control and deployment responsibilities in the appropriate Administrator role. TIBCO Administrator documentation describes domain-server roles and administrative access, and the EMS documentation covers destination administration and maintenance operations. See the TIBCO Administrator documentation and the TIBCO Enterprise Message Service documentation.

QUESTION NO: 4

The installation registry consists of which two files? (Choose two.)

A. vpd.properties

B. history.xmi

C. install.properties

D. machine.xmi

E. vpd.properties.tibco.

ANSWER: A E

Explanation:

The installation registry is represented by **vpd.properties** together with the host-specific **vpd.properties.tibco.<systemname>** file. TIBCO installation tooling uses this registry information to identify installed products, their installation locations, versions, and related environment-specific details. The general **vpd.properties** file provides the core registry data, while the TIBCO system-name variant holds registry information associated with the particular machine or system context. Maintaining both files is important when administering, upgrading, repairing, or uninstalling TIBCO software, because installation utilities rely on the registry to locate and correctly manage existing installations. In particular, moving, deleting, or manually editing registry artifacts without following TIBCO procedures can cause an installer or uninstaller to no longer recognize installed components. This registry convention is documented in TIBCO installation guidance for the ActiveMatrix BusinessWorks and Runtime Agent product families. See the TIBCO ActiveMatrix BusinessWorks documentation and the TIBCO Runtime Agent documentation for installation and environment-administration references.

QUESTION NO: 5

When a transition is created, which condition type is used by default?

A. Success with Condition

B. Error

C. Otherwise

D. Success if No Matching Condition

E. Success

ANSWER: E

Explanation:

Success is the default condition type assigned when a transition is created in TIBCO ActiveMatrix BusinessWorks 5. A transition controls the path that a process follows after an activity completes. With the default Success condition, the process proceeds along that transition when the preceding activity completes successfully. This makes Success the natural default because the most common process-flow requirement is to continue normal execution after an activity finishes without an error.

In BusinessWorks Designer, the transition's condition can subsequently be changed in the transition properties when the process requires more specialized routing behavior, such as evaluating an XPath expression, handling an error outcome, or defining fallback routing. However, those are explicit design choices made after the transition has been created; they are not the initial condition applied by the Designer.

This behavior is described in the BusinessWorks process-design documentation's discussion of transition conditions and their configuration. See the TIBCO documentation catalog for the ActiveMatrix BusinessWorks 5 documentation set at TIBCO ActiveMatrix BusinessWorks documentation and the product information page at TIBCO ActiveMatrix BusinessWorks.

QUESTION NO: 6

ActiveMatrix BusinessWorks uses which HTTP server by default?

A. JBoss

B. Pramati

C. Tomcat

D. Geronimo

ANSWER: C

Explanation:

Tomcat is correct. In the ActiveMatrix BusinessWorks 5 environment, Apache Tomcat is the default HTTP server technology used to provide the servlet-container and HTTP-serving capabilities required by the platform. This default is significant when deploying and administering BusinessWorks applications that expose HTTP-based resources, because Tomcat supplies the standard web-container behavior for handling HTTP requests, routing them to deployed components, and supporting the associated web application infrastructure.

Tomcat is a mature Java servlet container and HTTP server implementation. Its use aligns with the Java-based architecture of BusinessWorks 5 and gives the runtime a well-established implementation of the Java web and servlet model. Administrators should therefore recognize Tomcat as the default server when considering HTTP configuration, ports, connectors, deployment behavior, and web-application integration in this BusinessWorks generation.

TIBCO's documentation portal provides the documentation set for ActiveMatrix BusinessWorks releases, including version-specific administration and deployment material: TIBCO ActiveMatrix BusinessWorks documentation. Background on Tomcat's role as an HTTP server and Java servlet container is available from the Apache Tomcat project.

QUESTION NO: 7

A BusinessWorks process must be deployed that receives input from an EMS topic and publishes data to an EMS queue. The topic and queue names must contain a token used for migrating between testing and production environments. The system administrator must be able to purge the topic and queue from within the TIBCO Administrator.

Which two tools should you use to create a TIBCO Administrator domain that meets these requirements? (Choose two.)

A. Domain Utility

B. Designer

C. BusinessWorks Global Variable

D. AppManage Utility

ANSWER: A C

Explanation:

Domain Utility is required to create and configure a TIBCO Administrator domain. During domain setup, it can be used to associate the domain with the required infrastructure, including an EMS server connection. Once EMS is correctly configured for administration in the domain, an administrator can use the TIBCO Administrator EMS management interface to inspect destinations and perform administrative operations such as purging queues and topics. This establishes the domain-level administration capability requested in the scenario.

BusinessWorks Global Variable is the appropriate mechanism for making EMS destination names environment-specific without changing process design. A process can define the topic and queue names using global-variable expressions containing a token, then supply different values through deployment configuration for testing and production. This keeps the same BusinessWorks archive portable across environments while resolving the actual destination names at deployment time. TIBCO documentation describes BusinessWorks global variables as deployment-time configuration values and documents Administrator/domain administration separately; see the TIBCO ActiveMatrix BusinessWorks documentation and the TIBCO Administrator documentation.

QUESTION NO: 8

Given the following JMS Local Transaction Group:

- two JMS Queue Sender activities configured with a JMS Shared Connection1
- one JMS Topic Publisher activity configured with JMS Shared Connection2
- one Wait For JMS Queue Message activity

How many transactions are created for a single job for this transaction group?

A. 1

B. 2

C. 3

D. 4

E. 5

ANSWER: C

Explanation:

3 is correct. In a BusinessWorks JMS Local Transaction Group, JMS send and publish activities that use the same JMS Shared Connection participate in one local JMS transaction. Therefore, the two JMS Queue Sender activities configured with JMS Shared Connection1 are handled by one transaction. The JMS Topic Publisher configured with the separate JMS Shared Connection2 requires another local JMS transaction because local JMS transactions are scoped to the JMS connection/session used for the work.

The Wait For JMS Queue Message activity contributes a separate transaction for its queue-receive operation. Consequently, one job creates a transaction for the senders on the first shared connection, a transaction for the publisher on the second shared connection, and a separate transaction associated with waiting for and receiving the queue message: three

transactions in total. This behavior follows the JMS model in which a transacted session defines the scope for produced and consumed messages, while distinct connection/session contexts cannot be combined into one local transaction. See the TIBCO ActiveMatrix BusinessWorks documentation portal at TIBCO ActiveMatrix BusinessWorks 5.16 documentation and the JMS Session API description of local transaction boundaries at Oracle JMS Session documentation.

QUESTION NO: 9

Which two statements are true when checking out from a version control system with

ActiveMatrix BusinessWorks? (Choose two.)

A. A project is the largest unit of work.

B. A folder is the largest unit of work.

C. An activity file is the smallest unit of work.

D. A process file is the smallest unit of work.

ANSWER: A D

Explanation:

“A project is the largest unit of work.” is correct because ActiveMatrix BusinessWorks organizes design-time artifacts in projects. A project provides the top-level scope for related resources, including process definitions, shared resources, schemas, and folders. When source-control operations are performed from BusinessWorks Designer, the project is therefore the broadest logical collection that can be managed as a single work unit.

“A process file is the smallest unit of work.” is also correct. A BusinessWorks process definition is persisted as an individual process resource file and can be independently versioned, checked out, modified, and checked in. This granularity supports concurrent development: one developer can reserve a particular process while related processes and resources remain available for work by other team members. The Designer source-control integration maps repository operations to these BusinessWorks resource boundaries, allowing both project-level and individual-process-level control according to the scope of the intended change.

TIBCO’s product documentation catalog provides the ActiveMatrix BusinessWorks documentation sets, including Designer and source-control guidance: TIBCO ActiveMatrix BusinessWorks documentation. Product lifecycle and documentation availability information is also maintained at TIBCO Documentation.

QUESTION NO: 10

You wish to be able to configure a Global Variable on a per-service basis from TIBCO

Administrator. Which Global Variable option should you check?

A. Value

B. Service

C. Constraint

D. Type

ANSWER: B

Explanation:

The correct setting is **Service**. In TIBCO ActiveMatrix BusinessWorks 5, global variables normally provide deployment-time configuration values that can be substituted throughout a process or resource configuration. Selecting the Service option identifies the global variable as one that can be configured at the individual service level in TIBCO Administrator. This allows separate deployed services to use different values for the same global-variable definition, without requiring a redesign of the

process archive or a shared value across every service. This is especially useful for service-specific endpoints, credentials, timeouts, or other environment-dependent settings where each service instance requires its own configuration. The setting is made when defining the global variable in BusinessWorks Designer, then exposed through the Administrator deployment and service configuration interface. TIBCO's ActiveMatrix BusinessWorks documentation describes global variables as deployment configuration properties and documents the Administrator facilities used to manage service deployments and their properties. See the TIBCO ActiveMatrix BusinessWorks 5.16 documentation portal and the TIBCO ActiveMatrix BusinessWorks product documentation area for the applicable BusinessWorks and Administrator guides.

QUESTION NO: 11

What is the purpose of the tokenize function?

A. to concatenate two strings

B. extract fields from a string

C. to map undefined data to a defined data type

D. to replace a specific character in a string

ANSWER: B

Explanation:

The correct answer is **extract fields from a string**. In TIBCO ActiveMatrix BusinessWorks, `tokenize` is used in XPath expressions and mappings to split character data into separate tokens using a delimiter or a regular-expression pattern. This is useful when an incoming value contains multiple logical fields in one string—for example, a comma-separated list, a pipe-delimited identifier, or text whose components are separated by whitespace. The result is a sequence of individual string values, which can then be mapped, iterated over, or otherwise processed in a BusinessWorks process. For example, tokenizing `North|South|West` with a pipe delimiter produces distinct values for `North`, `South`, and `West`. This behavior supports extracting structured components from an unstructured or delimited input value, rather than merely altering the original string. The function's semantics align with the XPath/XSLT Functions and Operators definition of `fn:tokenize`, which returns strings obtained by breaking an input string at matches of a supplied pattern. See the W3C tokenize function specification and the TIBCO ActiveMatrix BusinessWorks documentation portal.

QUESTION NO: 12

An HTTP Receiver is overloaded and you with to throttle the number of incoming HTTP requests. What are two ways to accomplish this task? (Choose two.)

A. use the bw.plugin.http.server.minProcessors property

B. set FlowLimit on the entire engine

C. use the bw.plugin.http.server.maxProcessors property

D. set FlowLimit for the particular process

ANSWER: C D

Explanation:

The setting `bw.plugin.http.server.maxProcessors` limits the maximum number of processor threads available to the BusinessWorks HTTP server. Restricting this upper bound constrains the number of HTTP requests that can be actively handled concurrently, causing excess work to wait rather than allowing an unrestricted number of request-processing threads to consume engine resources. This is an HTTP-server-level throttle and is useful when an HTTP Receiver is receiving more concurrent traffic than the engine can safely process.

Setting FlowLimit for the particular process provides complementary control at the process level. A process FlowLimit caps the number of concurrent process instances that BusinessWorks permits for that process definition. Because each request

received by the HTTP Receiver initiates or drives a process instance, this limit prevents that specific service process from expanding without bound under load. Together, the HTTP server processor maximum and the process-specific FlowLimit control concurrency at the listener/transport and application-process layers, respectively. TIBCO ActiveMatrix BusinessWorks documentation is available through the TIBCO ActiveMatrix BusinessWorks documentation portal; the product's configuration and administration material is also indexed in the TIBCO Documentation site.

QUESTION NO: 13

You wish to deploy a pair of engines that share states between jobs via Shared Variables.

Which statement is true about this configuration?

A. States cannot be shared across engines, only across jobs within one engine.

B. Sharing states is achieved via TCP communication.

C. A database is required.

D. A shared location in the filesystem is sufficient.

ANSWER: C

Explanation:

A database is required. In TIBCO ActiveMatrix BusinessWorks 5, Shared Variables provide process-level state that can be accessed by multiple process instances and, when configured appropriately, by multiple engines. For two engines to share the same variable state reliably, BusinessWorks uses a database-backed shared-variable store. The database is the common, durable coordination point through which engines read and update shared values. This arrangement also supports consistent behavior across engine restarts and enables the locking/concurrency mechanisms needed when more than one engine can access the same state. Consequently, the engines must be configured to use the same supported database and the required BusinessWorks database schema must be created and available to both engines. A filesystem location alone does not provide the transactional shared-state repository required for this multi-engine configuration. See the TIBCO documentation portal for the ActiveMatrix BusinessWorks 5 administration and process-design documentation: TIBCO ActiveMatrix BusinessWorks 5.13 documentation. TIBCO's product documentation index is also available at docs.tibco.com.