

TIBCO BusinessWorks 6 Certification Exam

Tibco TCP-BW6

Version Demo

Total Demo Questions: 25

Total Premium Questions: 256

Buy Premium PDF

<https://passqueen.com>

support@passqueen.com

QUESTION NO: 1

Which two deployment options are available from TIBCO Business Studio? (Choose two.)

- A. DAA file loaded through ActiveMatrix Administrator
- B. Zip file
- C. XML file
- D. Scripted Deployment
- E. Deployment Wizard

ANSWER: D E

Explanation:

TIBCO Business Studio supports interactive deployment through the Deployment Wizard and automated deployment through Scripted Deployment. The Deployment Wizard provides the guided Studio interface for selecting the BusinessWorks application, configuring deployment-specific settings such as an application profile and target AppSpace, creating the deployable archive, and deploying it to the selected runtime environment. It is appropriate when an administrator or developer is performing a controlled deployment interactively.

Scripted Deployment supports repeatable, noninteractive deployment operations. Studio can generate deployment scripts that use BusinessWorks administration commands, allowing the same deployment procedure to be incorporated into build automation, release pipelines, or standardized promotion processes across environments. This preserves deployment consistency while avoiding manual entry of deployment settings for each release. These two approaches represent the interactive and automated deployment mechanisms provided through the Business Studio workflow. TIBCO's BusinessWorks documentation describes deployment administration and the command-line tooling used for automated operations in its product documentation: [TIBCO ActiveMatrix BusinessWorks documentation](#) and [BusinessWorks 6.9 documentation](#).

QUESTION NO: 2

The following is the declaration of secure daemon in Rendezvous C API: `tibrv_status tibrvSecureDaemon_SetUserCertWithKey(const char* userCertWithKey, const char* password);` Which two statements are correct about this function? (Choose two.)

- A. The password parameter is used for the password for this user.
- B. The password parameter is used to decrypt the private key.
- C. The userCertWithKey parameter is used to register user certificate with a private key. The text of this certificate may be in PEM or PK12 encoding.
- D. The userCertWithKey parameter is used to register user certificate with a private key. The text of this certificate must be in PEM encoding.

ANSWER: B D

Explanation:

The password parameter is used to decrypt the private key. This function supplies the secure daemon with identity material that includes both a user certificate and its associated private key. Because private keys are commonly stored in encrypted form, the password argument is the passphrase used to decrypt that private-key material before the secure daemon uses it. It is not an application user's login password or a general-purpose user credential.

The userCertWithKey parameter is used to register user certificate with a private key. The text of this certificate must be in PEM encoding. The API accepts the certificate and private key together as PEM text, allowing the Rendezvous secure

daemon to establish the client identity needed for secure Rendezvous operation. The certificate identifies the user or client, while the matching private key proves possession of that identity during cryptographic authentication. Supplying the material in the documented PEM representation ensures that the native C API can parse and register it correctly. See the official [TIBCO Rendezvous documentation](#) and the [TIBCO Rendezvous product page](#) for product documentation and security context.

QUESTION NO: 3

An event was defined with Time to Live equal to 10 minutes. An instance of that event arrived from a JMS destination configured with EXPLICIT_CLIENT_ACKNOWLEDGE mode and was consumed in the preprocessor. When will the JMS message corresponding to this event be acknowledged?

- A. immediately after the Event.consumeEvent call
- B. after 10 minutes
- C. during the post RTC phase
- D. after the preprocessor finishes

ANSWER: A

Explanation:

immediately after the Event.consumeEvent call is correct. With an event destination configured for EXPLICIT_CLIENT_ACKNOWLEDGE, BusinessEvents does not automatically acknowledge the inbound JMS message merely because it has been received or because preprocessing has completed. Instead, the application explicitly controls completion of the JMS message by invoking `Event.consumeEvent`. When that call is made for the event in the preprocessor, BusinessEvents acknowledges the JMS message at that point, marking it as successfully consumed by the client.

The event's Time to Live of 10 minutes governs the lifetime of the BusinessEvents event instance. It is not an acknowledgement delay or timer for the underlying JMS message. Explicit acknowledgement is useful where preprocessing logic must decide when processing has reached a point at which the source message can safely be removed from the destination. This behavior follows the JMS client-acknowledgement model, in which acknowledgement occurs as an explicit client action rather than as an automatic consequence of listener return. See the [TIBCO BusinessEvents documentation](#) and the [Jakarta Messaging specification](#).

QUESTION NO: 4

You want a process definition that will call one of three available subprocesses (SP1, SP2 and SP3), depending on the data being received by its starter. Which three actions must be taken to achieve this? (Choose three.)

- A. have SP1, SP2, and SP3 use the same input, output, and error schemas
- B. define a template and use it to implement SP1, SP2, and SP3
- C. use a Call Dynamic Process activity and select SP1, SP2, and SP3 as the alternatives
- D. set an XPATH expression in the Call Process activity to determine at run-time the name of the subprocess to call
- E. set one of the three subprocesses as the process name in the Call Process activity

ANSWER: A D E

Explanation:

The required configuration uses a Call Process activity whose target process name is resolved at run time. The subprocesses must use the same input, output, and error schemas because the calling process has one fixed activity interface at design time. BusinessWorks therefore needs every possible target to accept data in the same structure, return

compatible output, and expose compatible error definitions. This lets the caller map its starter data into the Call Process input and handle the returned data consistently regardless of which subprocess is selected.

A concrete subprocess must be selected as the process name in the Call Process activity during configuration. That selected process supplies the design-time interface and schema information used by the activity. The process name is then made dynamic by setting an XPath expression that evaluates the target subprocess name from the incoming starter data at run time. The expression can consequently select SP1, SP2, or SP3 while retaining the single compatible interface established at design time. This is the BusinessWorks dynamic subprocess-selection pattern using Call Process configuration, rather than a separate activity that enumerates alternatives. See the [TIBCO BusinessWorks documentation library](#) and the [TIBCO Documentation portal](#) for BusinessWorks activity configuration references.

QUESTION NO: 5

What can be modified from the TIBCO Designer Startup Panel Administration tab?

- A. domain membership
- B. user directory
- C. deployment target machine
- D. transition line mode

ANSWER: B

Explanation:

The correct answer is **user directory**. In TIBCO Designer, the Startup Panel's Administration tab includes the User Directory setting, which specifies the filesystem location Designer uses for the current user's local Designer data and configuration. This location is important because it separates user-specific working information from the shared TIBCO installation and project resources. An administrator or developer can change the directory when a standard location is required by an organization, when profiles must be stored on a managed network location, or when multiple local Designer environments need to use different user-specific settings. The setting is made before opening or working on projects so that Designer initializes its user environment from the selected directory. TIBCO Designer documentation describes the Startup Panel as the place to establish startup and environment settings, including the user directory used by the Designer client. See the TIBCO Designer product documentation at [TIBCO Designer documentation](#) and the related TIBCO Runtime Agent documentation at [TIBCO Runtime Agent documentation](#).

QUESTION NO: 6

An Enterprise Message Service server is configured with the following trace options:

`log_trace=DEFAULT,-LIMITs`

What will the logs contain?

- A. messages regarding ACL, route, connection errors, and no messages regarding destination limits
- B. messages regarding route, ACL, destination overflow errors, and no messages regarding connection limits
- C. messages regarding connection errors, destination overflow, route, and no messages regarding ACL violations
- D. messages regarding destination overflow, ACL, connection errors, and no messages regarding routing limits

ANSWER: A

Explanation:

messages regarding ACL, route, connection errors, and no messages regarding destination limits is correct because the EMS `log_trace` setting accepts predefined trace groups and allows an individual group to be removed by prefixing it with a minus sign. `DEFAULT` enables the standard set of operational trace categories, including

trace output for access-control-list activity, routing activity, and connection-related errors. The additional `-LIMITs` directive explicitly subtracts the `LIMITS` trace category from that default set. Consequently, the server continues to write the applicable default ACL, route, and connection diagnostic messages, but suppresses trace entries generated for destination-limit conditions. In EMS terminology, the `LIMITS` category is used for events associated with configured limits, such as destination overflow or other destination capacity-limit notifications. This syntax is useful when administrators want normal diagnostics while avoiding potentially noisy limit-related log entries. The EMS documentation describes server logging and trace configuration, including trace categories and additive or subtractive trace specifications, in its configuration reference. See the [TIBCO Enterprise Message Service documentation library](#) and the [EMS 8.6 documentation](#).

QUESTION NO: 7

How can you implement a consistent form experience for a common subset of data across multiple tasks and forms, while minimizing the number of places where changes must be made?

- A. Use embedded forms.
- B. Use a pageflow.
- C. Copy and paste a form section into other forms.
- D. Make a form pane read-only.

ANSWER: A

Explanation:

Use embedded forms. Embedded forms provide a reusable form component for a shared subset of data and presentation logic. A designer can create the common fields, layout, validation behavior, and bindings once, then embed that form in each task or parent form that requires the same user experience. This establishes a single maintenance point: when the shared content must change, the embedded form is updated rather than editing every consuming form individually. The parent form can still supply its own surrounding layout and task-specific fields, while the embedded component keeps the common portion uniform across all of its uses. This approach is particularly useful where several workflow tasks collect or display the same business-object attributes, such as customer details, contact information, or an approval summary. Reuse also reduces the risk that separately maintained copies gradually diverge in field configuration or appearance. TIBCO documentation resources for the relevant BPM and form-design capabilities are available through the [TIBCO ActiveMatrix BPM documentation portal](#) and the [TIBCO Product Documentation portal](#).

QUESTION NO: 8

`tibrvlisten` is started with the following settings: `-network ";224.2.2.2;224.6.6.6" "mytest"` `tibrvsend` is started with the following settings: `-network ";224.6.6.6;224.2.2.2" "mytest"`.

"message #1" What is the expected result?

- A. The listener receives "message #1" only when run on a different host than the sender.
- B. The listener will receive the message only if both the listener and sender connect remotely to the same daemon.
- C. The listener does not receive "message #1" when run on a different host than the sender.

ANSWER: A

Explanation:

The listener receives "message #1" only when run on a different host than the sender. In a Rendezvous network specification containing multiple multicast groups, the ordering of the groups is significant for the locally associated transport/network identity. Here, the sender and listener specify the same two multicast groups but in opposite order, so they do not have the same local network specification. When both programs use the same local Rendezvous daemon, the daemon's local-delivery behavior does not treat these differently ordered specifications as the same transport network for direct local delivery.

When the programs run on separate hosts, the sender's multicast traffic is placed on the configured multicast groups, and the listener is configured to listen on those groups. Since the listener includes the sender's multicast group in its network specification, it can receive the publication for the subject `mytest` after it arrives through the network. This behavior illustrates why multicast-group ordering must be kept consistent among local applications when local delivery is required, even where the configurations contain the same group addresses.

For Rendezvous network and transport configuration concepts, see the [TIBCO Rendezvous documentation](#) and the [TIBCO Rendezvous product documentation resources](#).

QUESTION NO: 9

A called sub-process has a Generate Error activity. There are no error transitions in any of the involved processes. What happens when the Generate Error activity is executed?

- A. The process instance terminates.
- B. The uncaught exception causes the engine to exit.
- C. The job interface transitions into an error state.
- D. The sub-process restarts processing with its start activity.

ANSWER: A

Explanation:

The process instance terminates. The Generate Error activity explicitly raises a BusinessWorks process error. BW6 first looks for an applicable error transition in the current scope of the called sub-process. When no such transition exists, the error is propagated out of the called process to its caller, where BW6 again searches for an applicable error handler or error transition. In this scenario, neither the called process nor any involved calling process provides an error transition, so the error remains unhandled. An unhandled process error ends execution of the affected process-instance invocation rather than resuming the sub-process or continuing normal processing. This behavior is important when designing reusable sub-processes: an error transition in a caller can handle an error propagated from a called process, perform compensation or logging, and decide how processing should proceed. Without that handling path, the instance has no recovery route and is terminated. See the TIBCO BusinessWorks documentation portal at [TIBCO ActiveMatrix BusinessWorks 6.10 documentation](#) and TIBCO's product documentation index at [docs.tibco.com](#).

QUESTION NO: 10

A process receives an HTTP request and must return a response to the same client connection. Which activity completes the request-response interaction?

- A. HTTP Response
- B. Send HTTP Request
- C. HTTP Receiver
- D. Wait

ANSWER: A

Explanation:

HTTP Response is correct. An HTTP Receiver starts the process when it accepts an inbound HTTP request, and the HTTP Response activity sends the configured status, headers, and body back to the client that initiated that request.

The response activity is associated with the inbound request context, so it is used to complete a synchronous HTTP interaction. A Send HTTP Request activity is used when a process acts as an HTTP client, not when it must respond to a request accepted by an HTTP Receiver. See the [TIBCO ActiveMatrix BusinessWorks documentation](#).

QUESTION NO: 11

You plan to test a network to ensure that it meets the following requirements: 100,000 point-to-point messages to be published by a single source in batch mode with ten messages per batch with an interval gap of 10 milliseconds. What is the correct `rvperf` command to start a test run?

- A. `rvperf -service 7500 -daemon 7500 -subject PTP -messages 100000 -interval 10 -batch 10`
- B. `rvperf -service 7501 -daemon 7500 -inbox -messages 100000 -interval 10 -batch 10`
- C. `rvperf -service 7501 -daemon 7500 -subDect _PTP -messages 100000 -interval 0.01 -batch 10`
- D. `rvperf -service 7500 -daemon 7500 -inbox -messages 100000 -interval 0.01 -batch 10`
- E. `rvperf -service 7501 -daemon 7500 -subDect _INBOX -messages 100000 -interval 10 -batch 10`

ANSWER: D

Explanation:

`rvperf -service 7500 -daemon 7500 -inbox -messages 100000 -interval 0.01 -batch 10` correctly translates every stated test parameter into the `rvperf` invocation. The `-inbox` switch selects inbox-based, point-to-point communication, which is required rather than ordinary subject-based multicast publication. The command configures both the Rendezvous service and the local Rendezvous daemon to use port 7500, providing a consistent local transport configuration for the test. `-messages 100000` sets the total number of messages produced by the source, while `-batch 10` groups transmission into bursts of ten messages. The interval value is expressed in seconds; therefore, 10 milliseconds must be specified as 0.01 seconds using `-interval 0.01`. This produces the requested ten-message batch followed by a 10-millisecond gap, repeated until 100,000 messages have been sent. The command therefore measures the intended point-to-point workload rather than a differently timed or differently addressed traffic pattern. Refer to the official [TIBCO Rendezvous documentation](#) and the [Rendezvous command-line documentation set](#) for `rvperf` parameter details.

QUESTION NO: 12

Which resource is used to define a reusable connection configuration for JDBC activities in a BusinessWorks application?

- A. JDBC Connection
- B. JDBC Query
- C. Database Palette
- D. SQL Mapper

ANSWER: A

Explanation:

A **JDBC Connection** shared resource is used to define reusable database connectivity for JDBC activities. The shared resource contains connection information such as the JDBC driver, database URL, user credentials, and connection-pool settings. JDBC Query, JDBC Update, and other JDBC activities can then reference that resource.

Using a shared JDBC connection resource centralizes database configuration and allows deployment-specific values to be externalized through application configuration. It also avoids repeating connection details in every process activity that accesses the same database. See the [TIBCO ActiveMatrix BusinessWorks documentation](#).

QUESTION NO: 13

Which of the following parameters are required to retrieve a deleted record using a web service query?

- A. `RECORD_STATE` and `ACTIVE`

- B. RECORD_ID and ACTIVE
- C. RECORD_ID and RECORD_STATE
- D. No parameters settings are required
- E. RECORD_VERSION and RECORD_ID
- F. RECORD_VERSION and ACTIVE

ANSWER: A

Explanation:

RECORD_STATE and ACTIVE is correct. A deleted record is not returned by a standard query that considers only currently active records. The web-service query must therefore include the record-state criterion so that the query can address records in the deleted lifecycle state, together with the active-status parameter that controls whether inactive/deleted records are included in the result set. Used together, these parameters make the request explicitly target the desired record status rather than relying on the service's normal active-record behavior.

In TIBCO MDM, record state and active status are distinct pieces of record metadata used when locating and managing repository records. Supplying both is important for a deleted-record retrieval because the record can still exist in the repository while no longer being treated as an active record. The request can consequently identify the deleted state and permit that state to be queried. This is consistent with TIBCO MDM's documented web-service and record-management model, in which record lifecycle information is exposed through service operations and repository metadata. See the [TIBCO MDM documentation library](#) and TIBCO's [MDM product documentation overview](#) for the applicable service and record-management documentation.

QUESTION NO: 14

What are two accurate descriptions of a record that has conflict? (Choose two.)

- A. A record has three versions and the same attribute in all versions has the same value in only two of the versions.
- B. A record has three versions and the same attribute in all versions has different values.
- C. A record has two versions and the same attribute in both versions has different values.
- D. A record has two versions and the same attribute in one version has a null/empty value and the attribute in the other version has a value.

ANSWER: B C

Explanation:

A record has a conflict when concurrent versions contain competing values for the same attribute and the system cannot identify a single value as the unambiguous result. "A record has three versions and the same attribute in all versions has different values" is therefore a conflict: each version supplies a distinct value for one attribute, requiring conflict resolution before a definitive merged value can be established. "A record has two versions and the same attribute in both versions has different values" is also a conflict because the two versions directly compete for the attribute's resulting value.

This reflects the essential conflict-detection principle used in versioned and replicated data: a conflict is based on divergent updates to the same data item, rather than simply the existence of multiple record versions. A resolver or an explicitly configured reconciliation policy is needed to select, merge, or otherwise determine the final attribute value. In BusinessWorks-based integration solutions, such resolution is important when coordinating state across systems and ensuring that downstream processing receives a consistent result. See the TIBCO BusinessWorks documentation landing page at [TIBCO ActiveMatrix BusinessWorks documentation](#) and TIBCO's product documentation portal at [docs.tibco.com](#).

QUESTION NO: 15

Where can a fault handler be attached in a process? (Choose two.)

- A. Scope Level
- B. Process Level
- C. Service Binding Level
- D. Application Module Level
- E. Reference Binding Level

ANSWER: A B

Explanation:

In TIBCO BusinessWorks 6, fault handling can be defined at the process level and at the scope level. A process-level fault handler provides a process-wide mechanism for handling faults that are not handled by a nested scope. It is therefore appropriate for applying common recovery, logging, notification, or compensation behavior for failures that propagate to the overall process boundary.

A scope-level fault handler is attached to a Scope activity and applies to the activities enclosed by that scope. This lets a process isolate a logical unit of work and provide specialized handling for faults generated within that unit. For example, a scope can catch a fault from an external invocation, perform local cleanup or recovery, and then either complete normally or rethrow a fault for higher-level handling. This nested model allows fault handling to be designed at the level where the relevant business context is available, while still retaining process-wide handling for unhandled failures.

TIBCO documents process and scope fault handling as part of the BusinessWorks process design and error-handling model. See the [TIBCO ActiveMatrix BusinessWorks documentation](#) and the [BusinessWorks product documentation portal](#).

QUESTION NO: 16

Which two transport types are supported by the ActiveEnterprise Adapter palette? (Choose two.)

- A. JMS Route transport
- B. Tibrv Reliable transport
- C. Tibrv Network transport
- D. JMS Topic transport
- E. HTTP transport

ANSWER: B D

Explanation:

The ActiveEnterprise Adapter palette supports adapter communication through both TIBCO Rendezvous and TIBCO Enterprise Message Service (JMS) transport configurations. **Tibrv Reliable transport** is therefore supported: it uses TIBCO Rendezvous reliable-message delivery semantics for communication with an adapter service. This transport is appropriate where the adapter endpoint is configured to use Rendezvous messaging and reliable delivery behavior.

JMS Topic transport is also supported. JMS topics implement publish-and-subscribe messaging, allowing a publisher to distribute a message to subscribers registered on the topic. In an ActiveEnterprise Adapter integration, this enables BusinessWorks processes and adapter services to exchange messages using the EMS/JMS topic model when that is how the adapter service has been deployed and configured.

These selections reflect the palette's support for the principal messaging infrastructures used by ActiveEnterprise Adapter services: Rendezvous and EMS/JMS. The specific transport selected in a process must correspond to the communication settings exposed by the target adapter service. TIBCO documentation for BusinessWorks and its adapter integrations is available through the [TIBCO Documentation Portal](#); product documentation is organized in the [TIBCO BusinessWorks documentation catalog](#).

QUESTION NO: 17

Which TIBCO CIM utility is used to apply a hot fix?

- A. hotfixUtil
- B. tibcoUtil
- C. tibcoCIMUtil
- D. migrateUtil
- E. customUtil

ANSWER: C

Explanation:

tibcoCIMUtil is the TIBCO Customer Information Management utility used for CIM maintenance operations, including applying a hotfix to an existing CIM installation. A CIM hotfix is not merely a replacement of an individual file; it can contain coordinated updates that must be applied with the product-aware utility so that the installed environment remains consistent. Using **tibcoCIMUtil** ensures that the hotfix procedure is performed in the supported CIM maintenance context and that the relevant installation metadata and components are handled as required by the release's hotfix instructions.

Administrators should always use the syntax, prerequisites, backup requirements, and post-installation steps specified in the documentation delivered with the particular hotfix, because those details can vary by CIM version and fix package. The TIBCO documentation portal provides the CIM documentation set and release-specific installation material at [TIBCO Customer Information Management documentation](#). Product downloads, hotfix availability, and support guidance are available through the [TIBCO Support portal](#).

QUESTION NO: 18

Which two statements are true about Subflow and SpawnWorkflow activities? (Choose two.)

- A. Subflows can be nested, but SpawnWorkflows can not.
- B. Both execution modes are supported by both activities.
- C. SpawnWorkflow is ASYNCHR and Subflow is SYNCHR.
- D. Subflow is ASYNCHR and SpawnWorkflow is SYNCHR.
- E. Both Subflows and SpawnWorkflow can be nested.

ANSWER: B E

Explanation:

Both execution modes are supported by both activities. This means that a designer can configure Subflow and SpawnWorkflow usage according to the required process interaction pattern: synchronous execution is appropriate when the calling workflow must wait for the invoked workflow to complete and use its result, while asynchronous execution is appropriate when work should continue independently. Supporting these modes gives process designers flexibility to balance response time, coupling, and background processing requirements.

Both Subflows and SpawnWorkflow can be nested. Nesting lets a workflow invoke another workflow that can itself invoke additional workflows, enabling a modular design in which reusable business logic is organized into manageable units. Such composition is particularly useful for separating common processing steps, coordinating longer-running work, and building larger solutions without placing all logic in one workflow definition. The practical design consideration is to define clear input, output, error-handling, and lifecycle boundaries at every workflow level. TIBCO BusinessWorks documentation describes process composition and activity configuration as core capabilities of BusinessWorks applications; see the [TIBCO ActiveMatrix BusinessWorks documentation library](#) and the [TIBCO BusinessWorks product documentation and resources](#).

QUESTION NO: 19

What are two valid advanced properties on the process level? (Choose two.)

- A. Ignore Case Suspend
- B. Ignore Blank Addressees
- C. Use Deadline For Task Duration
- D. Priority Value Expression
- E. Enable Case Prediction

ANSWER: B E

Explanation:

Ignore Blank Addressees and **Enable Case Prediction** are valid process-level advanced properties. Ignore Blank Addressees controls how the process handles an addressee list when one or more resolved addressee values are empty. Enabling it lets the process omit blank recipients rather than treating those empty values as recipients during address resolution. This is useful where recipient information is built dynamically from process data and some potential recipients are optional.

Enable Case Prediction is an advanced process configuration used for case-aware processing. It permits the process to participate in the platform's case-prediction capability, allowing prediction-related behavior to be enabled at the process definition level rather than being configured independently on each individual task. Both settings affect runtime behavior for the entire process and therefore belong in the process-level advanced-properties area. Consult the applicable product-version documentation when configuring these settings, because available properties and their labels can vary between releases. The TIBCO documentation portal is available at [TIBCO Documentation](#), and the BusinessWorks product documentation catalog is available at [TIBCO BusinessWorks documentation](#).

QUESTION NO: 20

Which two statements are true about the Block Data Visibility Data Pattern? (Choose two.)

- A. It is able to define data elements, which are accessible by each of the components of the corresponding sub- process.
- B. It can be defined by tasks, which are accessible only within the context of individual execution instances of u that task.
- C. It is specific to a process instance or case. They can be accessed by all components of the process during the execution of the case.
- D. It exists in the external operating environment, able to be accessed by components of processes during ^ execution.

ANSWER: A C

Explanation:

Block-oriented data visibility concerns data whose availability is controlled by the execution boundary in which it is defined. Data defined for a subprocess or block is available to the components that execute within that corresponding subprocess, allowing the activities in that block to share the same contextual information without exposing it outside the intended scope. This is represented by "It is able to define data elements, which are accessible by each of the components of the corresponding sub- process." In addition, process-instance data represents the broader enclosing execution context: it is created for a particular process instance (or case) and is available to components participating in that process during the case's execution. Thus, "It is specific to a process instance or case. They can be accessed by all components of the process during the execution of the case." correctly identifies the process-level visibility context in which block-scoped execution occurs. These visibility boundaries support encapsulation while enabling controlled sharing among related activities. The workflow data-pattern definitions describe block data as data accessible by all components in a corresponding block, and case data as data associated with and accessible throughout a case. See the [Workflow Data Patterns catalog](#) and [TIBCO ActiveMatrix BusinessWorks documentation](#).

QUESTION NO: 21

What are two accurate descriptions of a record that has conflict? (Choose two.)

- A. A record has three versions and the same attribute in all versions has the same value in only two of the versions.
- B. A record has three versions and the same attribute in all versions has different values.
- C. A record has two versions and the same attribute in both versions has different values.

ANSWER: B C

Explanation:

A record conflict is identified by comparing occurrences (versions) of the same record and determining whether a common attribute has competing populated values. "A record has three versions and the same attribute in all versions has different values" is a conflict because no shared value can be selected for that attribute across the three versions; each occurrence supplies a distinct competing value. "A record has two versions and the same attribute in both versions has different values" is likewise a conflict because the two occurrences directly disagree on the attribute value. These are the standard cases in which reconciliation requires a user or defined resolution policy to decide which value should be retained. The distinction is important in multi-version data management: a conflict concerns genuinely competing attribute values, rather than merely the presence of multiple record occurrences. TIBCO documentation and product documentation resources describe the relevant BusinessWorks platform and its data-management capabilities at [TIBCO ActiveMatrix BusinessWorks documentation](#) and [TIBCO BusinessWorks product information](#).

QUESTION NO: 22

Server R defines a global queue named Q1. R is also the owner of Q1. Servers P and S define routed queues Q1@R. The servers P, R, and S are connected by routes. Which two statements are true? (Choose two.)

- A. Messages are available to local and remote consumers even before the message reaches the home queue.
- B. When a message is sent from server P to Queue Q1, P forwards the message to the home queue on server R.
- C. When a message is sent from server S to Q1 and the route is not connected, the message is not delivered, and the client application receives the exception `DESTINATION_NOT_AVAILABLE`
- D. If Server P or the route connection from P to R fails, a consumer application of server P will not be able to receive messages from Q1 until the servers resume communication.

ANSWER: B D

Explanation:

In an EMS global queue, the queue owned by the designated server is the home queue and is the authoritative location for the queue's messages and delivery state. A routed queue such as Q1@R provides an access point on another server, but a produced message is routed toward the home queue. Therefore, when a producer connected to server P sends to Q1, server P forwards the message to the home queue owned by server R. This central ownership lets EMS maintain consistent queue behavior across all routed instances.

Consumption through a routed queue also depends on communication with the home server. A consumer application connected to server P uses the routed queue to access messages managed by the home queue on server R. If server P fails, or if its route to server R is unavailable, that consumer cannot obtain messages from Q1 through that routed instance until connectivity and inter-server communication are restored. These behaviors follow the EMS global-queue architecture, in which routed queues extend access to a home queue rather than independently owning its messages. See the [TIBCO Enterprise Message Service documentation](#) and the [EMS product documentation library](#).

QUESTION NO: 23

Which two input parameters are mandatory for the BuildEAR Utility? (Choose two.)

- A. the location of the TIBCO Designer project
- B. the project path of the Enterprise Archive Resource
- C. the location of the Global Variables file
- D. the output directory for the EAR file

ANSWER: A B

Explanation:

The BuildEAR utility needs both the location of the TIBCO Designer project and the project path of the Enterprise Archive Resource to build an Enterprise Archive file. The Designer project location identifies the source project repository that the utility must load. It contains the process definitions, resources, and project metadata from which the deployable archive is assembled. The Enterprise Archive Resource path identifies the archive resource within that project and supplies the archive-specific configuration, including the project components to package.

Together, these inputs tell BuildEAR both *where* the project is stored and *which* Enterprise Archive definition in that project is to be built. This separation is significant because a single Designer project may contain resources for different deployment archives. BuildEAR reads the selected Enterprise Archive Resource and resolves its references against the specified Designer project before producing the EAR package. TIBCO documentation for the applicable BusinessWorks and Designer release should be used to confirm the precise command syntax and supported switches for the installed version. See the [TIBCO ActiveMatrix BusinessWorks documentation](#) and the [TIBCO Designer documentation](#).

QUESTION NO: 24

Which processes are recovered when an ActiveMatrix BusinessWorks engine is gracefully stopped and restarted?

- A. all processes that were in progress when the shutdown was initiated
- B. all processes that have passed a Checkpoint activity but did not complete
- C. all processes that were paged by the engine during their execution
- D. all processes that are marked "persistent" during deployment

ANSWER: B

Explanation:

All processes that have passed a Checkpoint activity but did not complete are recovered because a Checkpoint activity creates a durable recovery point for an executing BusinessWorks process instance. When the engine is stopped gracefully, BusinessWorks can preserve the state associated with eligible checkpointed instances. After the engine restarts, it identifies incomplete instances with persisted checkpoint state and resumes each one from its most recently completed checkpoint rather than starting the process again from the beginning.

This mechanism is intended for long-running or stateful process flows where continuity through an engine restart is required. The checkpoint captures the information necessary for the engine to re-establish execution at that defined point, including the process context required to continue the remaining work. For recovery to be meaningful, the process must have reached the Checkpoint activity before it became incomplete; an instance that has not yet reached such a recovery boundary has no checkpoint state from which to resume.

TIBCO documents checkpointing and process-instance persistence as part of BusinessWorks process execution and recovery behavior. See the [TIBCO ActiveMatrix BusinessWorks documentation](#) and the [TIBCO Documentation portal](#) for the version-specific Process Design and Runtime Administration guides.

QUESTION NO: 25

tibrvlisten is started with the following settings: -network ";224.2.2.2;224.6.6.6" "mytest" tibrvsend is started with the following settings: -network ";224.6.6.6;224.2.2.2" "mytest". "message #1" What is the expected result?

- A. The listener receives "message #1" only when run on a different host than the sender.
- B. The listener will receive the message only if both the listener and sender run on the same host.
- C. The listener will receive the message only if both the listener and sender connect remotely to the same daemon.
- D. The listener does not receive "message #1" when run on a different host than the sender.

ANSWER: A

Explanation:

The listener receives "message #1" only when run on a different host than the sender. In this configuration, the network parameters contain the same two multicast addresses but in reverse order. Rendezvous treats the ordered multicast-group specification as significant for local delivery behavior. When the two programs use the same Rendezvous daemon on one host, the local optimization does not provide a matching local path between these reversed network specifications. Consequently, the local listener does not receive the publication through that path.

When the programs run on separate hosts, the remote Rendezvous communication mechanism uses multicast reception on the configured groups. Each side is configured with both multicast addresses, so the receiving host can join and receive traffic associated with the sender's multicast configuration. The subject `mytest` matches the listener's subscription, allowing the remote listener to receive the published payload, `message #1`. This behavior illustrates that network-string ordering can matter when assessing whether two Rendezvous transports communicate locally versus through the network.

See the TIBCO Rendezvous product documentation at [TIBCO Documentation: Rendezvous](#) and the [TIBCO Rendezvous 8.8 documentation set](#).