

IBM Cloud Pak for Business Automation v24.0.0 Solution Architect

IBM C1000-194

Version Demo

Total Demo Questions: 7

Total Premium Questions: 71

Buy Premium PDF

<https://passqueen.com>

support@passqueen.com

Topic Break Down

Topic	No. of Questions
Topic 1, Cloud Pak for Business Automation architecture	7
Topic 2, Cloud Pak for Business Automation installation	3
Topic 3, Cloud Pak for Business Automation administration	3
Topic 4, Cloud Pak for Business Automation capabilities	58
Total	71

QUESTION NO: 1

An organization is designing an Operational Decision Manager decision service for product pricing. Product-specific pricing rules must remain independent, while eligibility rules must be reused by several decision services and maintained in one location.

Which two rule-project design practices support these requirements? (Choose two.)

- A. Place product-specific pricing rules in a main rule project.
- B. Place reusable eligibility rules in a standard rule project referenced by the main rule projects.
- C. Copy the eligibility rules into every main rule project.
- D. Embed the eligibility rules directly in the calling Business Automation Workflow process.
- E. Place all product and eligibility rules in one unstructured rule project.

ANSWER: A B

Explanation:

Placing product-specific rules in a main rule project keeps the logic that belongs to a particular decision service under the ownership and lifecycle of that service. **Placing reusable eligibility rules in a standard rule project referenced by the main rule projects** supports reuse while allowing the shared logic to be changed and governed centrally.

Copying eligibility rules into each main rule project creates duplicate logic and increases the risk of inconsistent policy changes. Embedding the rules in Business Automation Workflow removes them from ODM governance, and placing all logic in one unstructured project does not preserve the required separation.

QUESTION NO: 2

Which statement describes Business Automation Navigator?

- A. A web-based image viewer that allows users to view and annotate PDFs.
- B. A browser-based interface that provides users with a unified work experience.
- C. A thick client that enables users to complete data processing for all document types.
- D. A hybrid client-based application which allows users to work offline.

ANSWER: B

Explanation:

Business Automation Navigator is correctly described as “A browser-based interface that provides users with a unified work experience.” It is the web client within IBM Cloud Pak for Business Automation that brings business users’ daily content and automation work into a common interface. Users can access and work with content from supported repositories, launch and interact with business automation applications, use dashboards and features, and access configured extensions such as plug-ins. This unified browser experience helps organizations present content-centric work and related automation capabilities in a consistent workspace rather than requiring users to move among separate desktop clients. IBM documentation characterizes Business Automation Navigator as the browser-based user interface for working with content and business automation applications. Its role therefore focuses on providing the user-facing work environment and integration surface for these capabilities. See IBM’s [Business Automation Navigator capability documentation](#) and the [IBM Business Automation Navigator component overview](#).

QUESTION NO: 3

A company uses one Business Automation Navigator deployment for claims processors and records administrators. The two groups require different repository connections, menu options, and access to Navigator features.

Which configuration should be used to provide these tailored user experiences?

- A. Create separate Business Automation Navigator desktops.
- B. Create separate FileNet object stores.
- C. Deploy a separate Cloud Pak for Business Automation instance for each group.
- D. Create a separate Foundation custom resource for each group.

ANSWER: A

Explanation:

Create separate Business Automation Navigator desktops is correct. A desktop defines the Navigator user experience, including the repositories, features, menus, and appearance that are available to its users. Access to each desktop can then be controlled for the appropriate user groups.

Separate object stores address content-repository design, not the Navigator user interface. Separate namespaces, Cloud Pak deployments, or Foundation custom resources would introduce unnecessary infrastructure separation when the requirement is to tailor the experience within one Navigator deployment.

QUESTION NO: 4

Through which interface can web or mobile applications integrate with Automation Decision Services to invoke a decision?

- A. SOAP Web Service
- B. JMS
- C. CORBA
- D. RESTful calls

ANSWER: D

Explanation:

RESTful calls is correct because Automation Decision Services exposes deployed decision services through REST APIs. A web application, mobile client, or another service can submit the required input data in an HTTP request to the decision runtime endpoint and receive the decision result in the HTTP response. This approach is well suited to client applications because REST uses standard web protocols, is language independent, and supports commonly used payload formats such as JSON.

In Automation Decision Services, decision authors build and deploy decision services, after which consuming applications invoke the deployed service at runtime. The REST interface provides a clean boundary between the client application and the decision logic, allowing the decision to be reused without embedding its rules directly in each web or mobile application. It also aligns with cloud-native and microservice integration patterns used by Cloud Pak for Business Automation. IBM documentation describes the decision runtime REST API as the interface for executing decision services and accessing runtime capabilities. See the [IBM Automation Decision Services documentation](#) and the [IBM Decision Service REST API documentation](#).

QUESTION NO: 5

Which Cloud Pak for Business Automation capabilities are specifically designed for Decision Management?

- A. Automation Decision Services and Operational Decision Manager
- B. Automation Decision Services and Process Mining
- C. Advanced Document Processing and Operational Decision Manager

ANSWER: A

Explanation:

Automation Decision Services and Operational Decision Manager are the Cloud Pak for Business Automation capabilities designed for Decision Management. They support the lifecycle of operational business decisions, allowing organizations to define decision logic, manage its governance, validate it, deploy it as a service, and execute it consistently within business applications and automated workflows.

Operational Decision Manager provides established business-rule and decision-management capabilities. It enables teams to externalize decision logic from application code, so policy and rule changes can be governed and maintained by appropriate business and technical stakeholders. Its decision services can then be deployed and invoked at runtime to make repeatable, traceable decisions at scale.

Automation Decision Services complements this capability with a modern, low-code experience for modeling and authoring decisions. Business users can use graphical decision models to represent decision requirements clearly, collaborate on the decision logic, and publish executable decision services. Together, these offerings address both governed rule management and accessible decision-service design, which are central to Decision Management in IBM Cloud Pak for Business Automation. IBM's product documentation describes the decision-authoring capabilities in [Automation Decision Services](#) and the governed rules and decision services offered by [Operational Decision Manager](#).

QUESTION NO: 6

A team uses Automation Decision Services and must collaborate on decision models with developers working from a remote source-control system.

Which two outcomes are supported by configuring the remote repository as a Git provider? (Choose two.)

- A. Version-controlled storage of decision automation artifacts.
- B. Collaboration through branches, commits, merges, and reviews.
- C. Execution of deployed decision services directly from the repository.
- D. Replacement of decision-service review and approval governance.
- E. Publication of decision results to Apache Kafka without further configuration.

ANSWER: A B

Explanation:

Version-controlled storage of decision automation artifacts is supported because the Git repository retains commits and versions of the project content. **Collaboration through standard Git practices** is also supported, allowing team members to work with branches, commits, merges, and review processes appropriate to their repository workflow.

Configuring a Git provider does not itself execute decision services, replace deployment governance, or turn a repository into a runtime event stream. Those functions require the relevant decision runtime, lifecycle configuration, or event-processing capabilities.

QUESTION NO: 7

A customer wants to automate some of the typical operations performed in Decision Center Business Console.

Which two operations can be automated using the Decision Center REST API? (Choose two.)

- A. Create action rules and decision tables.
- B. Retrieve Rule Execution Server statistics.

- C. Create operations and deployment configurations.
- D. Retrieve decision services from the repository.
- E. Build, download, or deploy a RuleApp for a deployment configuration.

ANSWER: D E

Explanation:

Retrieve decision services from the repository and **Build, download, or deploy a RuleApp for a deployment configuration** are supported Decision Center REST API automation activities. The API is intended to let external tools and CI/CD pipelines interact programmatically with Decision Center artifacts and lifecycle processes that users otherwise perform in Business Console. A client can identify and retrieve decision-service content from the Decision Center repository, allowing an automated process to work with the appropriate governed rule assets and associated versions.

The REST API also supports the deployment lifecycle for decision services. An automation client can invoke the build process for a selected deployment configuration, obtain the generated RuleApp archive when needed, and deploy that RuleApp to a configured target. This enables repeatable, script-driven promotion and deployment processes while continuing to use Decision Center as the system of record for business-rule artifacts and deployment configuration. These capabilities are particularly useful when integrating Operational Decision Manager with build servers and release pipelines, where the build and deployment steps must run without manual Business Console interaction.

IBM documents the Decision Center REST API and its decision-service deployment capabilities in the [Decision Center REST API documentation](#) and the [decision-service deployment documentation](#).