

# Agentforce Specialist

## Comprehensive Exam Preparation Notes

**Source Material:** Certification Questions (Set 2)

*Organized by Topic for Easy Revision*

Includes: AI Agents, Data Cloud, Prompt Builder, Testing, and Deployment

## Contents

---

<b>1</b>	<b>Agentforce Testing &amp; Evaluation</b>	<b>2</b>
1.1	Testing Center Capabilities . . . . .	2
1.2	Monitoring & Analytics . . . . .	2
<b>2</b>	<b>Data Cloud &amp; Indexing</b>	<b>2</b>
2.1	Indexing Unstructured Data (PDFs/Knowledge) . . . . .	2
2.2	Search Strategies (Vector vs. Keyword vs. Hybrid) . . . . .	3
<b>3</b>	<b>Prompt Engineering (Prompt Builder)</b>	<b>3</b>
3.1	Prompt Templates Types . . . . .	3
3.2	Grounding Context . . . . .	3
<b>4</b>	<b>Agent Configuration &amp; Logic</b>	<b>4</b>
4.1	Deterministic vs. Probabilistic Behavior . . . . .	4
4.2	Topic & Action Selection . . . . .	4
<b>5</b>	<b>Deployment Lifecycle</b>	<b>4</b>
5.1	Moving to Production . . . . .	4
5.2	Permissions . . . . .	4
<b>6</b>	<b>Advanced Connectivity (Integration)</b>	<b>5</b>
6.1	Connecting Agents Systems . . . . .	5

## 1 Agentforce Testing & Evaluation

Testing is a critical part of the lifecycle. The exam focuses heavily on how to validate agent behavior deterministically and at scale.

### 1.1 Testing Center Capabilities

- **Consistent Pass/Fail Logic:** Use **structured batch testing** with validation per test utterance. This is preferred over manual checks or simple proxies like customer ratings.
- **Synthetic Data:** The Testing Center can generate **AI-generated synthetic test utterances** based on natural language variations to cover a broad range of user phrasing.
- **Environments:** Always use **Sandbox environments** replicated from production. Never test in personal developer orgs (unrepresentative data) or directly in production (risk of modifying real data).

#### Test Case Success Criteria

When defining success criteria for test cases, you should specify the **Expected Topic API Name**. This ensures the agent is routing the query to the correct topic logic.

### 1.2 Monitoring & Analytics

- **Agentforce Analytics:** Use this to get high-level metrics like usage and performance.
- **Utterance Analysis:** Specifically used to analyze what users are saying and how the agent classifies it.
- **Session Tracing:** The tool for deep debugging. It captures the full view: reasoning engine executions, actions taken, prompt inputs/outputs, and error messages.
- **Key Metrics to Monitor:** Response times, accuracy/relevance of answers, and resolution success.

## 2 Data Cloud & Indexing

Data Cloud provides the "brain" and context for the agent. The exam tests specific configurations for retrieval and indexing.

### 2.1 Indexing Unstructured Data (PDFs/Knowledge)

- **PDF Attachments in Knowledge:** To allow an agent to read PDF attachments on Knowledge articles, you must enable '**Include Related Attachments**' for the **Knowledge\_\_kav** object and map the **ContentDocumentLink** unstructured data model object (UDMO).
- **Updating Content:** If content changes frequently, you cannot just rely on old embeddings. While some systems auto-update, often you must ensure the search index is rebuilt or refreshed to avoid "stale" answers.
- **External PDFs:** To use external PDFs (not in Salesforce) with a clicks-only setup, upload them to the **Agentforce Data Library** (File Source). This automatically builds a search index and creates a retriever.

## 2.2 Search Strategies (Vector vs. Keyword vs. Hybrid)

- **Natural Language & Variations:** Use **Vector Search** (Semantic Search) for handling natural language questions and concepts (e.g., "How do I reset...").
- **Specific Identifiers (Model Numbers):** Use **Keyword Search** for precise matches like model numbers or IDs.
- **Best Practice (Hybrid):** For scenarios involving *both* structured queries (model numbers) and natural language (misspellings, concepts), use **Hybrid Search**.

### Retrievers: Standard vs. Custom

- **Standard/Default Retriever:** Good for basic needs, but often lacks specific field controls.
- **Individual/Custom Retriever:** Required when you need to configure **filters** (e.g., Product Line), apply **recency ranking**, or specify exactly which fields are returned.
- **Missing Fields:** If a required field is missing in a retriever, you usually need to **update the search index** to include that field.

## 3 Prompt Engineering (Prompt Builder)

### 3.1 Prompt Templates Types

- **Field Generation:** Use this to populate a single field on a record (e.g., auto-populating a **Description** field).
- **Flex Template:** The most versatile. Use this for building custom agent actions or when you need to combine data from multiple sources (DMOs, Apex, Flow) for complex tasks.
- **Sales Email:** Specifically for generating email drafts.

### 3.2 Grounding Context

- **Grounding with Related Lists:** You can pass related records (e.g., Activities on a Contact) into a prompt template using **related list grounding**.
- **Data Cloud Data:** To access Data Cloud data in Prompt Builder, use **Data Cloud related lists** or fetch data using **prompt-initiated flows**.

### Best Practices for Prompts

- **Clarity:** Use clear roles ("You are a support agent"), tasks ("Summarize the case"), and constraints ("Limit to 5 sentences").
- **Permissions:** Users need the **Prompt Template User** permission set to *use* templates. To *manage/create* them, they need **Prompt Template Manager**.

## 4 Agent Configuration & Logic

---

### 4.1 Deterministic vs. Probabilistic Behavior

AI is probabilistic (it guesses), but businesses need deterministic outcomes (rules).

- **Security/Gating:** If an action (like Booking) requires a specific status (e.g., "Premium User"), do **not** just rely on the LLM instructions. Use a **conditional filter** on the action itself (e.g., Membership Tier = Premium).
- **Verification:** For sensitive data, do not rely on the LLM to ask for ID. Use a **custom variable** (e.g., IsCustomerVerified) set by a flow, and gate the sensitive action using a conditional check on that variable.
- **Action Sequencing:** To force actions to happen in a specific order, use **Topic Instructions** effectively, but for strict logic, Flow orchestration or specific "If/Then" routing logic in the topic instructions is key.

### 4.2 Topic & Action Selection

The "Reasoning Engine" decides what to do based on:

- **Topic Selection:** Driven by the **Topic Name** and the **Classification Description** (not the Scope, which is for the LLM's internal behavior).
- **Action Selection:** Driven by the **Action Instructions**. These tell the engine *when* to call the action.
- **Troubleshooting:** If an action isn't being selected, iterate on the **Action Instructions** or the **Classification Description**.

## 5 Deployment Lifecycle

---

### 5.1 Moving to Production

- **Deployment Tools:** You can use **Change Sets** or **Salesforce CLI/Metadata API**.
- **Components:** You must deploy all dependencies: Flows, Apex, Agent definitions, Topics, and Actions.
- **Activation:** Deploying does **not** auto-activate. You must **manually activate** the agent in production after deployment.

### 5.2 Permissions

- **Agent User:** The "Agent" runs as a user (e.g., Platform Integration User or a specific Agent user). This user needs permissions to do the work.
- **Flows:** If an agent runs a Flow, the Agent User must have the **Run Flows** permission.
- **Knowledge:** The Agent User needs **Access Knowledge records** and **Allow View Knowledge**.

## 6 Advanced Connectivity (Integration)

---

### 6.1 Connecting Agents Systems

- **Model Context Protocol (MCP):** A standard for connecting AI models to data sources (like external product recommendation engines). It abstracts the connection so the agent can "consume" external tools easily.
- **Agent-to-Agent (A2A):** Used when one agent needs to call another agent (e.g., a Service Agent calling a specialized "Banking Agent"). It provides a standardized framework for discovery and communication.
- **Web Integration:** To put an agent on a custom website, use the **Agent API** or **Embedded Service** frameworks.