# Agentforce Specialist

## Exam Preparation Notes (Set 2)

**Source Material:** Agentforce Specialist OCT7 Pass 1.pdf

*Key Concepts: Data Cloud Indexing, Agent Types, & Logic Control*

---

**Focus Areas: Retrievers, Sales vs. Service, and Prompt Templates**

---

# Contents

# 1 Data Cloud & Retrieval

This section covers how Agentforce interacts with data stored in Data Cloud, specifically regarding indexing and retrieving information.

## 1.1 Search Index Maintenance

- **Scenario:** Product documentation or knowledge articles change frequently.

- **Solution:** You must **Rebuild the search index** to ensure retrieval accuracy.

- **Why?** Leaving embeddings unchanged leads to "stale" data (hallucinations based on old info). Manually deleting chunks is inefficient and prone to error.

## 1.2 Retrievers

- **Default Creation:** When you create a Search Index in Data Cloud, a **Default Retriever** is automatically created.

- **Naming:** This default retriever typically shares the name of the custom search index.

- **Customization:** If you need specific filtering (e.g., "only show Active warranties") or field mapping, you generally create a *custom* retriever, but the default is the starting point.

> **Indexing Rule**
>
> Data Cloud does not "live update" vectors instantly for every small change in source data. Periodic **rebuilding** or scheduled updates of the index are often required for highly dynamic content.

# 2 Agent Configuration & Logic

Choosing the right agent and controlling its behavior is a core exam topic.

## 2.1 Selecting the Right Agent Type

- **Service Agent:** Best for resolving cases, answering FAQs, and handling support tickets.

- **Sales Agent:** Best for **Upselling**, creating Quotes, and handling large-deal escalations (e.g., escalating to an Account Executive if an order $> \$25k$).

- **Scenario (Upgrade Service Plan):** If a customer wants to "upgrade" and the process involves creating quotes and calculating prorated amounts, this is a commercial transaction.

- **Correct Choice: Sales Agent**. Even if it starts as "support," the intent is revenue generation (upsell).

## 2.2 Controlling Action Sequence

- **Problem:** The LLM (Reasoning Engine) executes actions in a random order (e.g., Booking before Verification), causing failures.

- **Solution:** Update **Topic Instructions**.

- **Technique:** Write comprehensive instructions that explicitly detail the *required sequence* (e.g., "First retrieve sessions, THEN verify eligibility, THEN create booking").

- **Alternative:** For extremely strict, complex logic that cannot fail, a Flow is used, but for general sequencing within a topic, clear Instructions are the primary configuration tool.

# 3 Prompt Engineering

## 3.1 Customizing Standard Responses

- **Scenario:** You want to change how the agent summarizes a Case (e.g., "Standard Case Summary").

- **Solution:** Create a **custom Record Summary prompt template** for the Case object.

- **Implementation:** You do not just "edit" the standard template directly; you typically clone or create a new custom template in Prompt Builder and assign it to the action.

> **Prompt Builder Types**
>
> Remember the difference between **Field Generation** (filling one field) and **Record Summary** (summarizing a whole record for a user view). For a "Case Summary" feature, use **Record Summary**.

# 4 Multi-Agent Interoperability

## 4.1 Agent-to-Agent (A2A)

- **Purpose:** Allows one agent (e.g., Service) to call another specialized agent (e.g., Billing or Sales) to handle a specific sub-task.

- **Discovery:** Agents can "discover" each other's capabilities if configured correctly, allowing for modular agent design rather than one massive, complex agent.