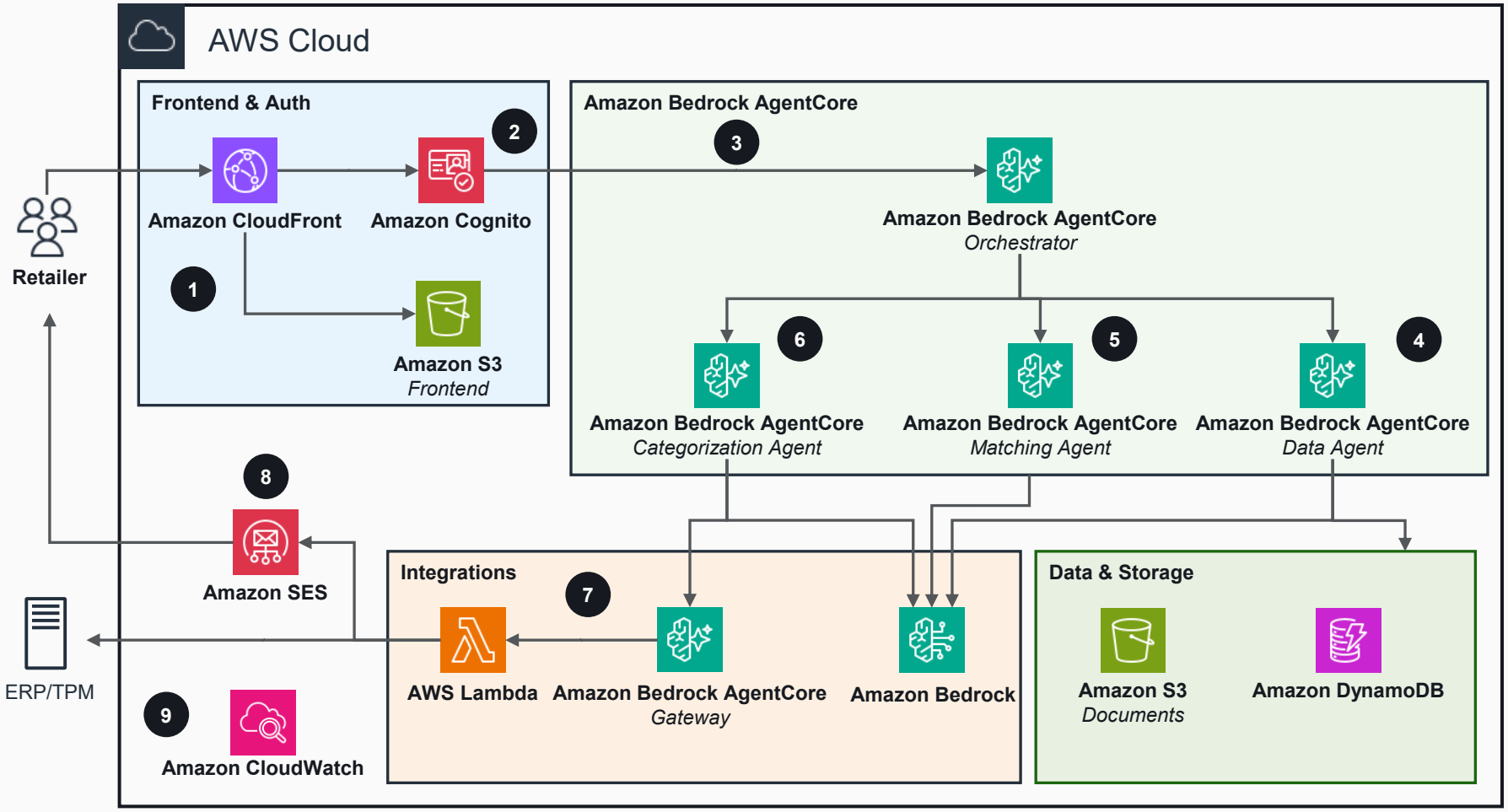
Guidance for Trade Promotion Deduction Agents on AWS

This architecture diagram shows how a multi-agent system built on Amazon Bedrock AgentCore automates trade promotion deduction validation with ERP integration.



1. You access the web application hosted on **Amazon S3** and delivered via **Amazon CloudFront**.
2. You authenticate using **Amazon Cognito User Pools**, which verifies your identity and issues identity tokens. **Amazon Cognito Identity Pools** then exchanges these tokens for temporary, scoped AWS credentials, enabling access to backend AWS services.
3. You upload trade promotion deduction document packages through the frontend web application, which triggers the multi-agent document analysis workflow.
4. The Orchestrator Agent, deployed on **Amazon Bedrock AgentCore Runtime's** serverless, session-isolated environment, delegates to the Data Agent, which extracts data from documents in **Amazon S3** and stores extracted data in **Amazon DynamoDB**.
5. The Matching Agent, running on **Amazon Bedrock AgentCore**, uses **Amazon Bedrock**, AWS's fully managed generative AI service, with Retrieval Augmented Generation (RAG) to perform three-way matching, comparing deduction claims against campaign agreements and purchase order data retrieved from **Amazon DynamoDB**.
6. The Categorization Agent, running on **Amazon Bedrock AgentCore**, uses **Amazon Bedrock** foundation model inference to classify deduction types and writes categorized results to **Amazon DynamoDB**.
7. The agent uses **Amazon Bedrock AgentCore Gateway** which converts the **AWS Lambda** function into an MCP-compatible, agent-ready tool to update the retailer's external ERP/TPM system with validated deduction details.
8. Status updates are sent to the retailer via **Amazon SES**.
9. All agent activities are logged to **Amazon CloudWatch** for monitoring and audit.

Reviewed for technical accuracy June 15, 2026

AWS Reference Architecture