aws

实施指南

AWS 上的生成式人工智能应用程序构建者

AWS 上的生成式人工智能应用程序构建者: 实施指南

Table of Contents

该解决方案有助于生成式人工智能 (AI) 应用程序的开发、快速实验和部署

AWS 上的生成式 AI 应用程序生成器无需深厚的 AI 经验即可促进生成式人工智能 (AI) 应用程序的开发、快速实验和部署。此 AWS 解决方案通过帮助您，加速开发并简化实验：

- 摄取您的业务特定数据和文档
- 评估和比较大型语言模型的性能 (LLMs)
- 使用 AI 代理运行多步骤任务和工作流程
- 快速构建可扩展的应用程序，并使用企业级架构部署这些应用程序

AWS 上的生成式 AI 应用程序生成器包括与以下内容的集成：

- LLMs 可在 Amazon Bedrock 上线
- LLMs 你已经在 Amazon A SageMaker I 上部署的
- 用于检索增强生成 (RAG) 的 Amazon 基岩知识库
- Amazon Bedrock Guardrails 将实施保障措施并减少幻觉
- Amazon Bedrock Agents 将构建可以执行任务协调和完成的代理工作流程
- Amazon Bedrock AgentCore 将构建、部署和管理具有扩展运行时支持的生产就绪型人工智能代理
- 用于企业数据和工具集成的@@ 模型上下文协议 (MCP) 服务器

此外，该解决方案还支持使用 LangChain 连接器连接到您选择的型号。这些连接器可在与解决方案一起部署的 A WS Lambda 函数中找到。你可以从无代码部署向导开始构建生成式 AI 应用程序，用于对话搜索、AI 生成的聊天机器人、文本生成和文本摘要。

本实施指南概述了 AWS 上的生成式 AI 应用程序生成器解决方案、其参考架构和组件、部署规划注意事项以及将该解决方案部署到 Amazon Web Services (AWS) 云的配置步骤。

本指南适用于想要在其环境中在 AWS 上实施生成式 AI 应用程序构建器的解决方案架构师、业务决策者、 DevOps 工程师、数据科学家和云专业人士。

使用以下导航表可快速找到这些问题的答案：

如果您想. . .	阅读. . .
了解运行此解决方案的成本。 运行此解决方案的估计成本因您部署的组件和查询数量而异。 在美国东部（弗吉尼亚北部）地区使用默认参数和 100 个活跃用户运行部署控制面板一个月的费用约为每月 20.12 美元。 对于在 LLM 中每天执行 100 次查询的 1 位企业用户在没有 RAG 的情况下部署文本用例，其费用约为每月 12.39 美元。 使用 Amazon Kendra 索引支持每天 8,000 次互动的 RAG 用例的费用约为每月 204.26 美元，再加上知识库的成本。	成本
了解此解决方案的安全注意事项。	安全性
了解如何为此解决方案规划限额。	配额
了解哪些 AWS 区域支持此解决方案。	支持的 AWS 区域
查看或下载此解决方案中包含的 AWS CloudFormation 模板，以自动部署该解决方案的基础设施资源（“堆栈”）。	AWS CloudFormation 模板
访问源代码，（可选）并使用 AWS Cloud Development Kit（AWS CDK）部署解决方案。	GitHub 存储库

功能和优势

AWS 上的生成式 AI 应用程序生成器解决方案提供以下功能：

快速实验

该解决方案消除了部署具有不同配置的多个实例以及比较输出和性能所需的繁重工作，从而允许用户快速进行实验。尝试各种快速工程 LLMs、企业知识库、护栏、AI 代理和其他参数的多种配置。

选择和可配置性

该解决方案预先构建了连接各种模型（例如 Amazon Bedrock 提供的模型）的连接器，使您可以灵活地部署自己选择的模型以及您喜欢的 AWS 和领先的 FM 服务。 LLMs您还可以启用 Amazon Bedrock Agents 来完成各种任务和工作流程。

代理生成器

构建和部署具有完整生命周期管理功能的生产就绪型 AI 代理。配置系统提示，为企业工具和数据访问集成模型上下文协议 (MCP) 服务器，并启用内存功能，以便在对话中保留上下文。代理部署在 Amazon Bedrock 上 AgentCore ，具有扩展的运行时支持和实时流媒体响应。

工作流程生成器

使用分层委托，将多个 Agent Builder 代理编排成复杂的工作流程。创建一个主管代理，该代理可以自主选择和协调专门的 Agent Builder 代理来处理多步骤任务。在重复使用现有的 Agent Builder 部署的同时，配置代理描述、委派策略和工作流级别的内存。

生产就绪

该解决方案采用 AWS Well-Architected 设计原则构建，提供企业级安全性和可扩展性、高可用性和低延迟，确保以高性能标准无缝集成到您的应用程序中。

可扩展的模块化架构

通过集成您的现有项目或本地连接其他 AWS 服务来扩展此解决方案的功能。由于这是一款开源应用程序，因此您可以使用随附的 LangChain 编排层或 Lambda 函数来连接您选择的服务。

与 Service Catalog AppRegistry 和应用程序管理器集成，这是 AWS Systems Manager 的一项功能

该解决方案包括一个服务目录 AppRegistry资源，用于在 AWS Service Catalog 和 AWS S ystems Manager 应用程序管理器中将解决方案的 CloudFormation 模板及其底层资源注册为应用程序。AppRegistry 通过这种集成，您可以集中管理解决方案的资源。

Agent Builder vs 基岩代理用例

该解决方案为使用 AI 代理提供了两种不同的方法，每种方法都适合不同的用例和要求：

功能	基岩代理用例	代理生成器
目的	调用预先部署的 Amazon 基岩代理	构建、部署和管理自定义代理
配置	仅限代理 ID 和别名 ID	完整的代理配置：系统提示、型号、MCP 服务器、内存
部署	简单的调用层	AgentCore 运行时完成代理生命周期
运行时	亚马逊 Bedrock Agents 服务	AgentCore 带斯特兰兹的 Amazon Bedrock SDK
工具集成	在 Bedrock 代理控制台中配置	对上下文协议 (MCP) 服务器和内置 Strands 工具进行建模
内存	由 Bedrock Agents 管理（最长 30 天）	AgentCore 具有可配置短期和长期保留功能的内存
定制	仅限于预部署的代理设置	完全控制提示、模型、工具和行为
适用于	快速部署现有代理	定制代理开发和生产部署

> **Note**
>
> 这两个选项都支持实时流媒体、对话历史记录和企业级安全性。

工作流程生成器

Workflow Builder 通过创建主管代理，将工作委托给专门的 Agent Builder 代理来实现多代理编排。每个工作流程包括：

- 主管代理：接收用户请求并协调专业代理的入口点代理
- 专业代理：主管可以将任务委派给的 Agent Builder 用例
- 代理即工具模式：主管将每个 Agent Builder 代理注册为工具，并自主选择要使用的代理

功能	代理生成器	工作流程生成器
目的	构建和部署单个自定义代理	编排多个代理生成器代理
代理类型	带有 MCP 工具的单一代理	主管代理 + 多个代理生成器代理
工具集成	MCP 服务器和 Strands 工具	Agent Builder 代理注册为工具
委托	直接调用工具	自主代理选择和委托
复杂度	单代理任务	多步骤、多代理工作流程
代理重用	不适用	重复使用现有的代理生成器部署
适用于	有针对性的单一领域任务	需要多种专业化认证的复杂工作流程

Note

- 作为专业代理代理，工作流至少需要 1 个 Agent Builder 用例
- 所有专业代理都必须是部署在 GAAB 中的代理生成器用例

使用案例

企业数据问题解答

LLMs 其他基础模型已经在大量数据上进行了预训练，使它们能够在许多自然语言处理 (NLP) 任务中表现出色。但是大多数基础模型 LLMs 都是静态的，并且已经过预先训练，这限制了它们准确回答有关新的、专业的或专有主题的问题的能力。使用基于提示的学习，您可以利用法学硕士强大的 NLP 和文本生成功能，通过企业数据提供更丰富的客户体验。

快速生成式 AI 原型设计

该解决方案开箱即用，与各种模型提供商和用例捆绑在一起。借助易于使用的部署向导，客户可以部署预先构建的用例，以实现对不同的生成式 AI 原型和工作负载的快速实验。

多法学硕士学位比较和实验

LLMs 表现不同，考虑到您的应用程序的特定需求，您可能会发现一个LLM比另一个更适合您的应用程序。这可能是出于与性能、准确性、成本、创造力或许多其他因素有关的原因。该解决方案允许您快速部署多个用例，使您能够尝试和比较不同的配置，直到找到满足您需求的配置。

概念和定义

本节介绍重要概念并定义此解决方案特有的术语：

管理员用户

在本指南的背景下，管理员用户是负责管理部署中包含的内容的人。该用户可以访问部署仪表板用户界面，并主要负责策划业务用户体验。这是我们的主要目标客户。

企业用户

在本指南的背景下，业务用户代表已为其部署用例的个人。他们是知识库的消费者，也是负责评估和试验知识库的 LLMs客户。

部署控制面板

部署仪表板是一个 Web 界面，可用作管理员用户查看、管理和创建用例的管理控制台。该仪表板使客户能够利用此仪表板快速试验、迭代和生产各种 AI/ML 工作负载。 LLMs

DevOps 用户

在本指南中， DevOps 用户负责在 AWS 账户中部署解决方案、管理基础设施、更新解决方案、监控性能以及维护解决方案的整体运行状况和生命周期。

用例

用例是与整体解决方案隔离开来的应用程序，这些应用程序与 LLMs 之集成，通过在新的或现有的应用程序中添加自然语言界面来提供更丰富的客户体验。用例可以通过部署仪表板进行部署，也可以单独部署。

> **Note**
>
> 有关 AWS 术语的通用参考，请参阅 AWS 词汇表。

架构概述

本节提供使用此解决方案部署的组件的参考实现架构图。

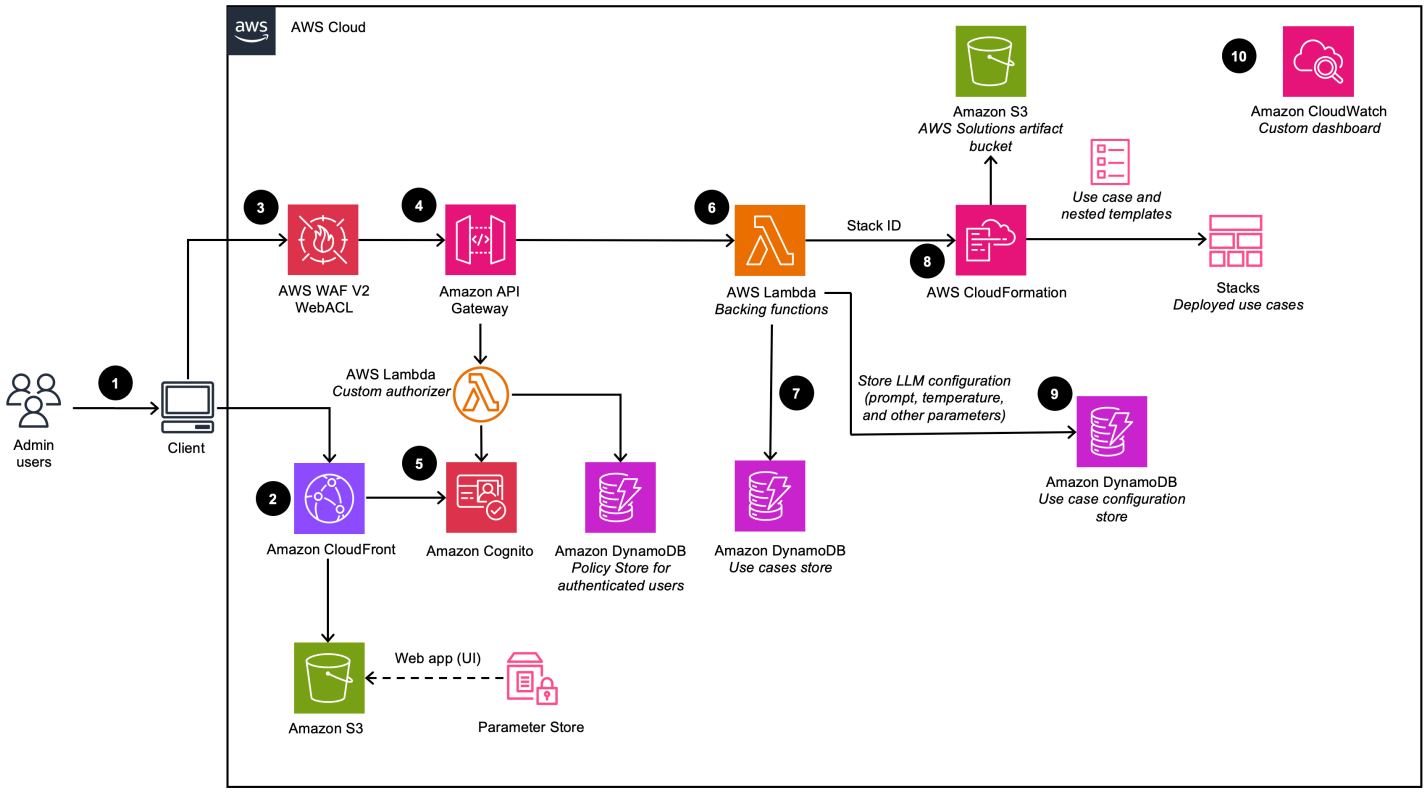
架构图

为了支持多种用例和业务需求，此解决方案提供了六个 AWS CloudFormation 模板：

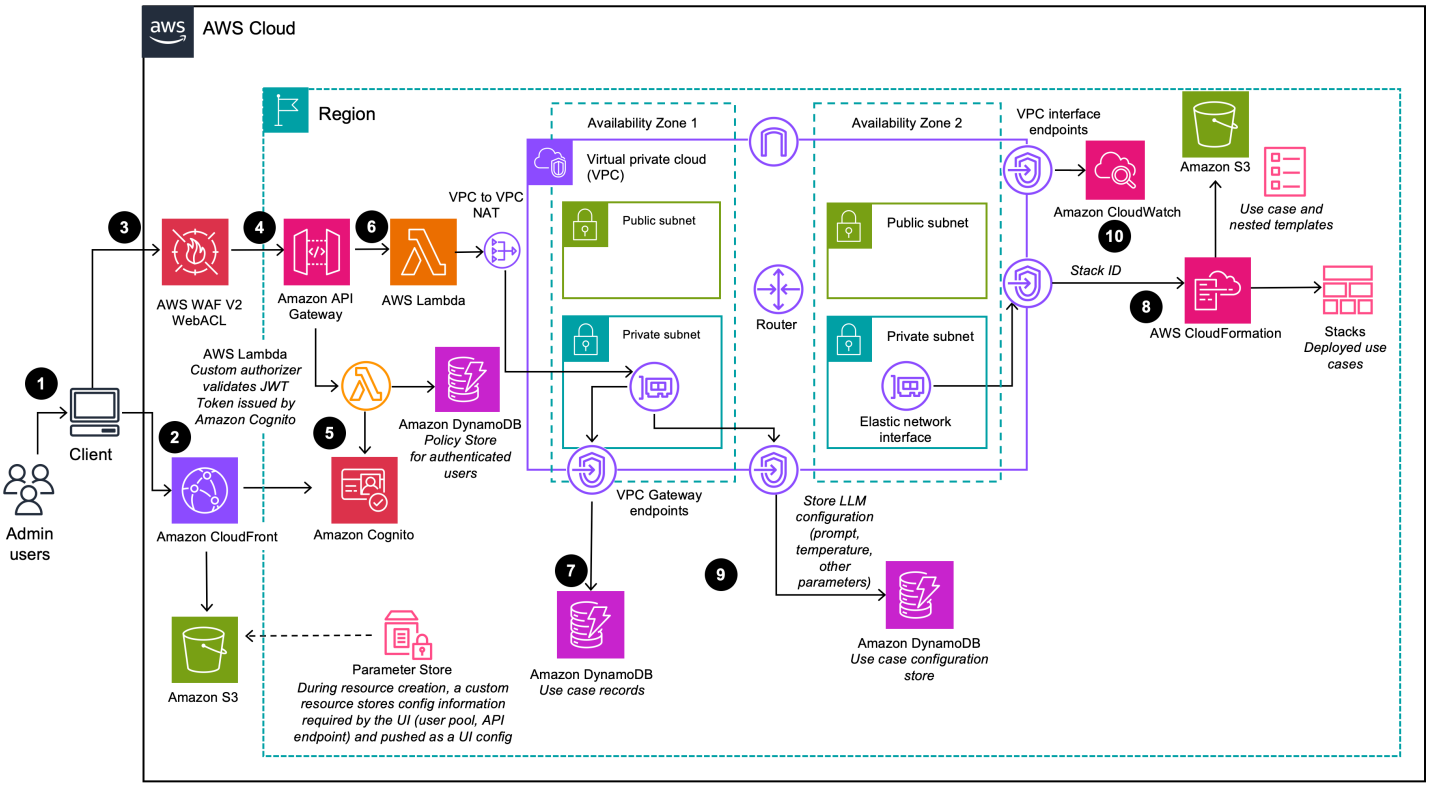
1. 部署仪表板-部署仪表板是一个 Web 界面，可用作管理员用户查看、管理和创建用例的管理控制台。该仪表板使客户能够利用此仪表板快速试验、迭代和生产各种 AI/ML 工作负载。 LLMs
2. 文本用例-文本用例使用户能够使用生成式 AI 体验自然语言界面。此用例可以集成到新的或现有的应用程序中，并且可以通过部署仪表板进行部署，也可以通过提供的 URL 独立部署。
3. Bedrock Agent 用例 ——基岩代理用例允许使用现有的 Bedrock Agent 来完成任务或自动执行重复的工作流程。
4. MCP 服务器-MCP 服务器用例支持部署和管理模型上下文协议服务器，这些服务器为 AI 应用程序提供标准化工具和资源访问权限。支持用于封装现有 Lambda 函数和外部 MCP 服务器的网关方法 APIs，以及用于部署自定义容器化 MCP 服务器的运行时方法。
5. A@@ gent Buil der-Agent Builder 支持在 Amazon Bedrock 上创建和部署可用于生产的 AI 代理，并 AgentCore 具有全面的配置控制、MCP 服务器集成和内存管理功能。
6. W orkflow Builder-Workflow Builder 允许创建主管代理，这些代理使用代理作为工具的委托模式为复杂的多代理工作流编排多个代理生成器代理。

部署控制面板

描述部署控制面板架构（在禁用 VPC 选项的情况下部署时）



描述部署控制面板架构（在启用 VPC 选项的情况下进行部署时）



> **Note**
>
> AWS CloudFormation 资源是基于 AWS Cloud Development Kit (AWS CDK) 结构创建的。

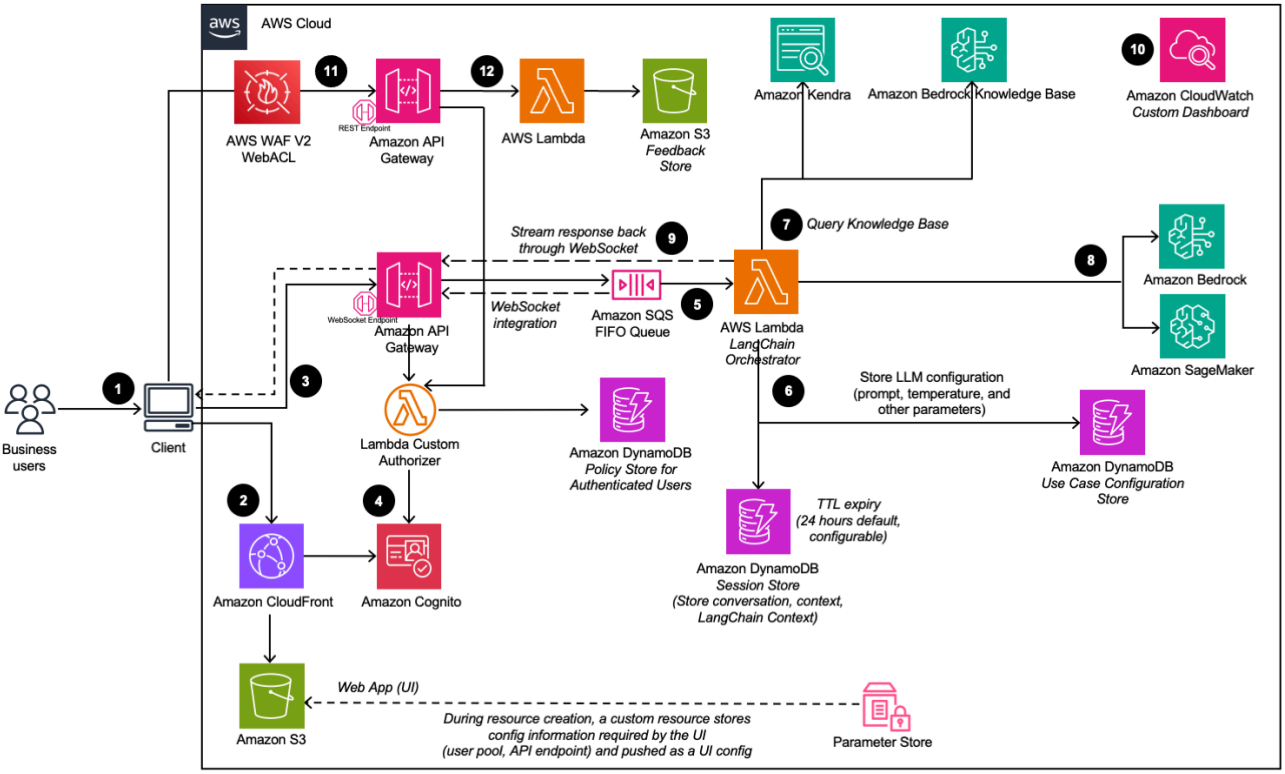
使用 AWS CloudFormation 模板部署的解决方案组件的高级流程如下：

1. 管理员用户登录部署控制面板用户界面 (UI)。
2. 亚马逊 CloudFront提供网络用户界面，该用户界面托管在亚马逊简单存储服务 (Amazon S3) 存储桶中。
3. AWS WAF 可以保护他们 APIs 免受攻击。此解决方案配置了一组名为 Web 访问控制列表 (Web ACL) 的规则，这些规则根据可配置的、用户定义的 Web 安全规则和条件允许、阻止或计数 Web 请求。
4. 网页用户界面利用一组使用 Amazon API Gat APIs ew ay 公开的 REST。
5. Amazon Cognito 对用户进行身份验证并支持 CloudFront 网页用户界面和 API Gateway。
6. AWS Lambda 为 REST 终端节点提供了业务逻辑。这个支持 Lambda 函数管理和创建了使用 AWS 执行用例部署所需的资源。 CloudFormation
7. 亚马逊 Dy namoDB 存储部署列表。
8. 管理员用户创建新用例时，支持的 Lambda 函数会为请求的用例启动 CloudFormation 堆栈创建事件。
9. 管理员用户在部署向导中提供的所有 LLM 配置选项都保存在 DynamoDB 中。部署使用此 DynamoDB 表在运行时配置 LLM。
10. 该解决方案使用 Amazon CloudWatch 从各种服务中收集运营指标，以生成自定义控制面板，使您可以监控解决方案的性能和运行状况。

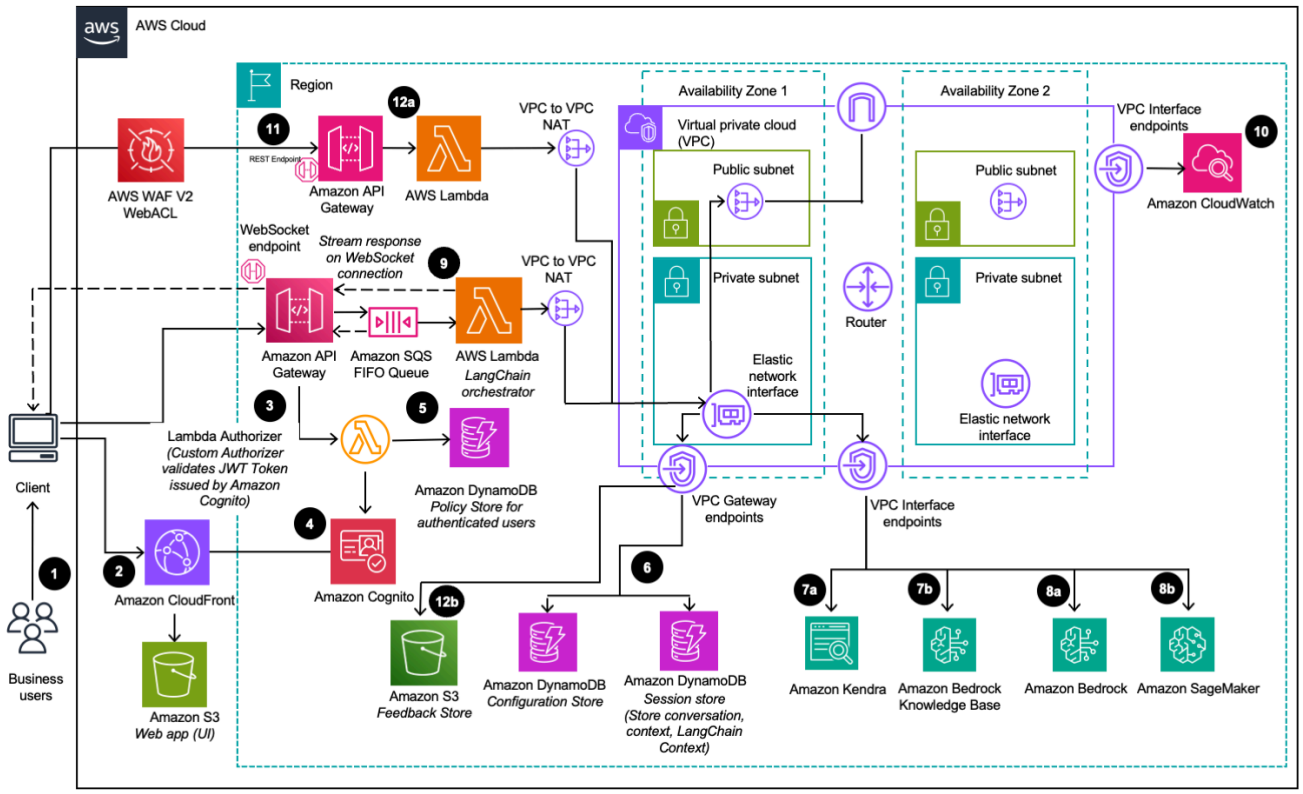
> **Note**
>
> - 如果您选择在 Amazon VPC 中部署此解决方案，则数据将在您的私有网络内路由。
> - 尽管部署控制面板可以在大多数 AWS 区域启动，但根据服务可用性，部署的用例会受到某些限制。有关更多详细信息，请参阅支持的 AWS 区域。

文本用例

描绘文本用例架构（在禁用 VPC 选项的情况下部署时）



描绘文本用例架构（在启用 VPC 选项的情况下部署时）



使用 AWS CloudFormation 模板部署的解决方案组件的高级流程如下：

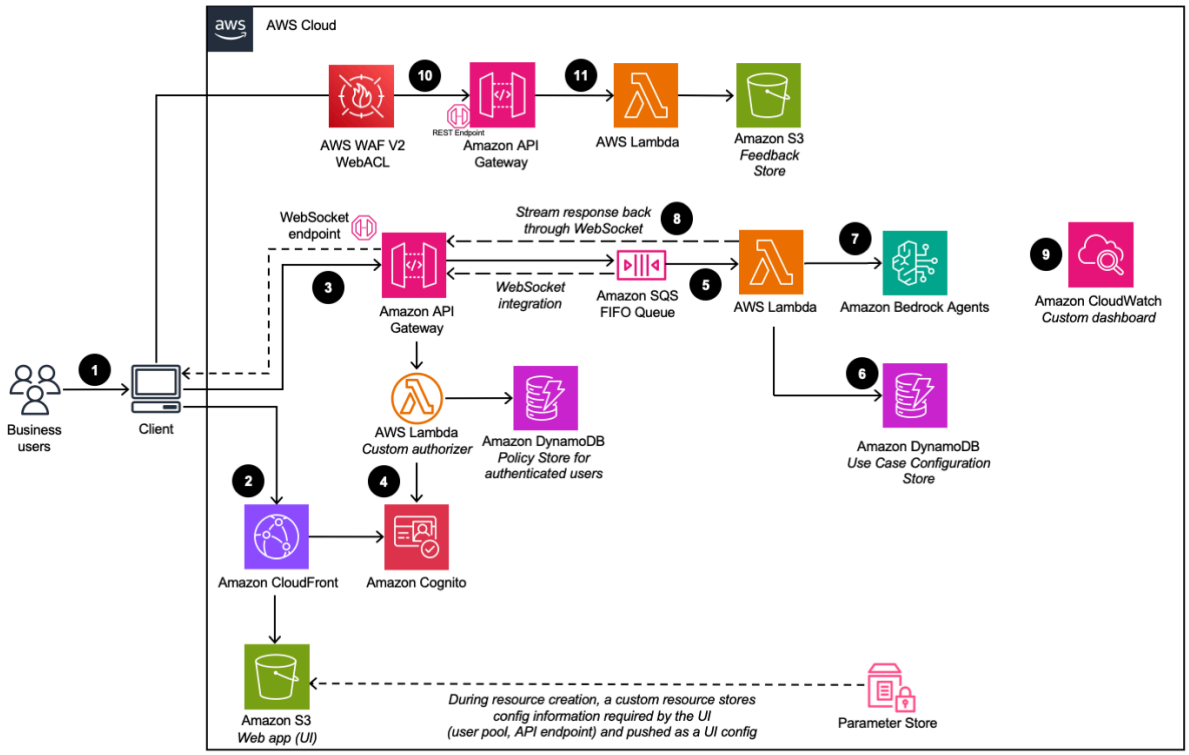
1. 管理员用户使用部署仪表板部署用例。企业用户登录到用例用户界面。
2. CloudFront 提供托管在 S3 存储桶中的 Web 用户界面。
3. 网页用户界面利用了使用 API Gateway 构建的 WebSocket 集成。API Gateway 由自定义 Lambda 授权方函数提供支持，该函数会根据身份验证用户所属的 Amazon Cognito 群组返回相应的 AWS 身份和访问管理 (IAM) 策略。该策略存储在 DynamoDB 中。
4. Amazon Cognito 对用户进行身份验证并支持 CloudFront 网页用户界面和 API Gateway。
5. 来自企业用户的传入请求将从 API Gateway 传递到亚马逊 SQS 队列，然后传递到 Orchestrator。LangChain LangChain Orchestrator 是 Lambda 函数和层的集合，它们为满足来自业务用户的请求提供了业务逻辑。该队列支持 API Gateway 到 Lambda 集成的异步操作。队列将连接信息传递给 Lambda 函数，然后这些函数会将结果直接发布回 API Gateway websocket 连接，以支持长时间运行的推理调用。
6. O LangChain rchestrator 使用 Amazon DynamoDB 来获取配置的 LLM 选项和必要的会话信息（例如聊天记录）。
7. 如果部署启用了知识库，则 LangChain Orchestrator 会利用 Amazon Kendra 或 Amazon Bed rock 知识库来运行搜索查询来检索文档摘录。

8. LangChain Orchestrator 使用知识库中的聊天记录、查询和上下文，创建最终提示并将请求发送到托管在 Amazon Bedrock 或 Amazon SageMaker AI 上的 LLM。
9. 当响应从 LLM 返回时，LangChain Orchestrator 会通过 API Gateway 将响应流回 WebSocket 以供客户端应用程序使用。
10. 该解决方案使用 Amazon CloudWatch 从各种服务中收集操作指标，以生成自定义控制面板，使您可以监控部署的性能和运行状况。
11. 如果启用了反馈收集，则可以使用利用 Amazon API Gateway 的 REST API 终端节点来收集用户反馈。
12. 支持 lambda 的反馈使用其他特定于用例的元数据（例如使用的模型）来补充提交的反馈，并将数据存储在 Amazon S3 中，供用户日后分析和报告。 DevOps

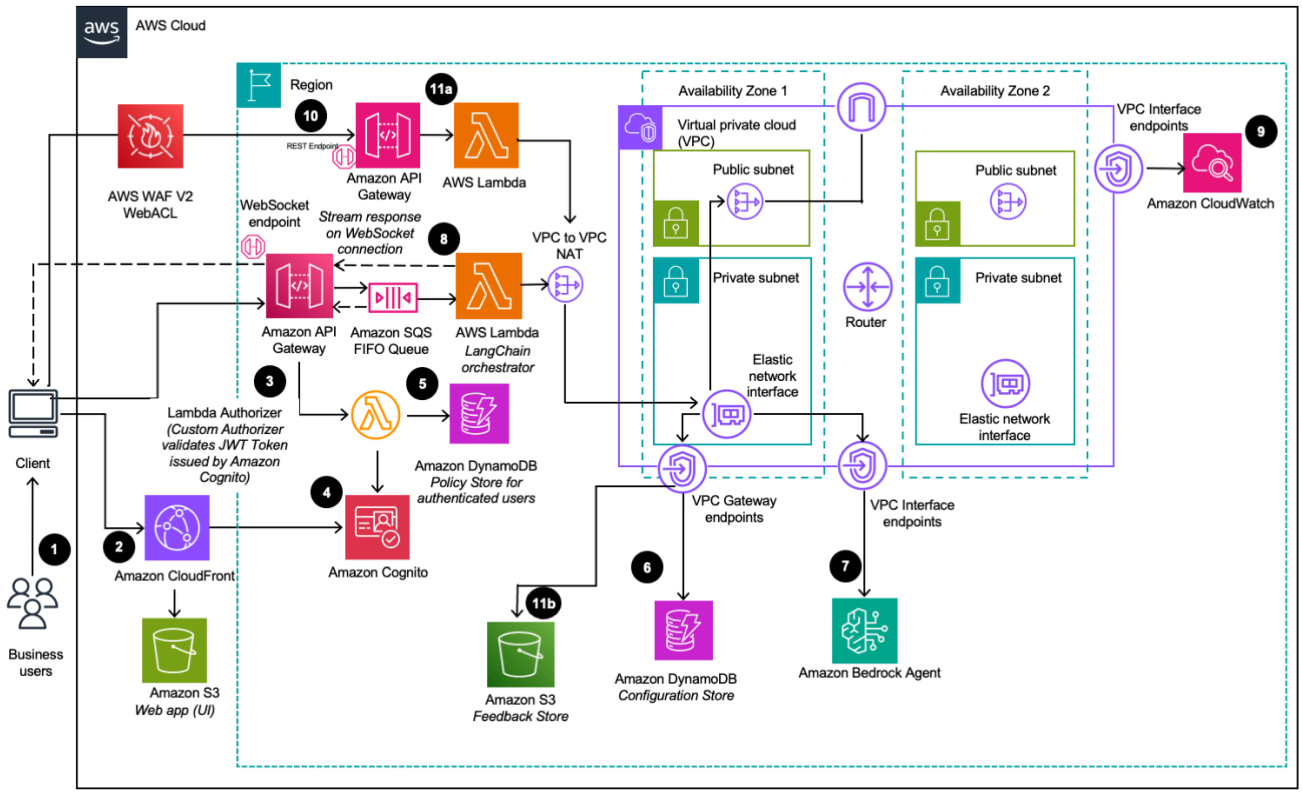
> **Note**
>
> 如果您选择在 Amazon VPC 中部署此解决方案，则数据将路由到您的私有网络。

基岩代理用例

描述了 Bedrock Agent 用例架构（在禁用 VPC 选项的情况下部署时）



描述了 Bedrock Agent 用例架构（在启用 VPC 选项的情况下部署时）



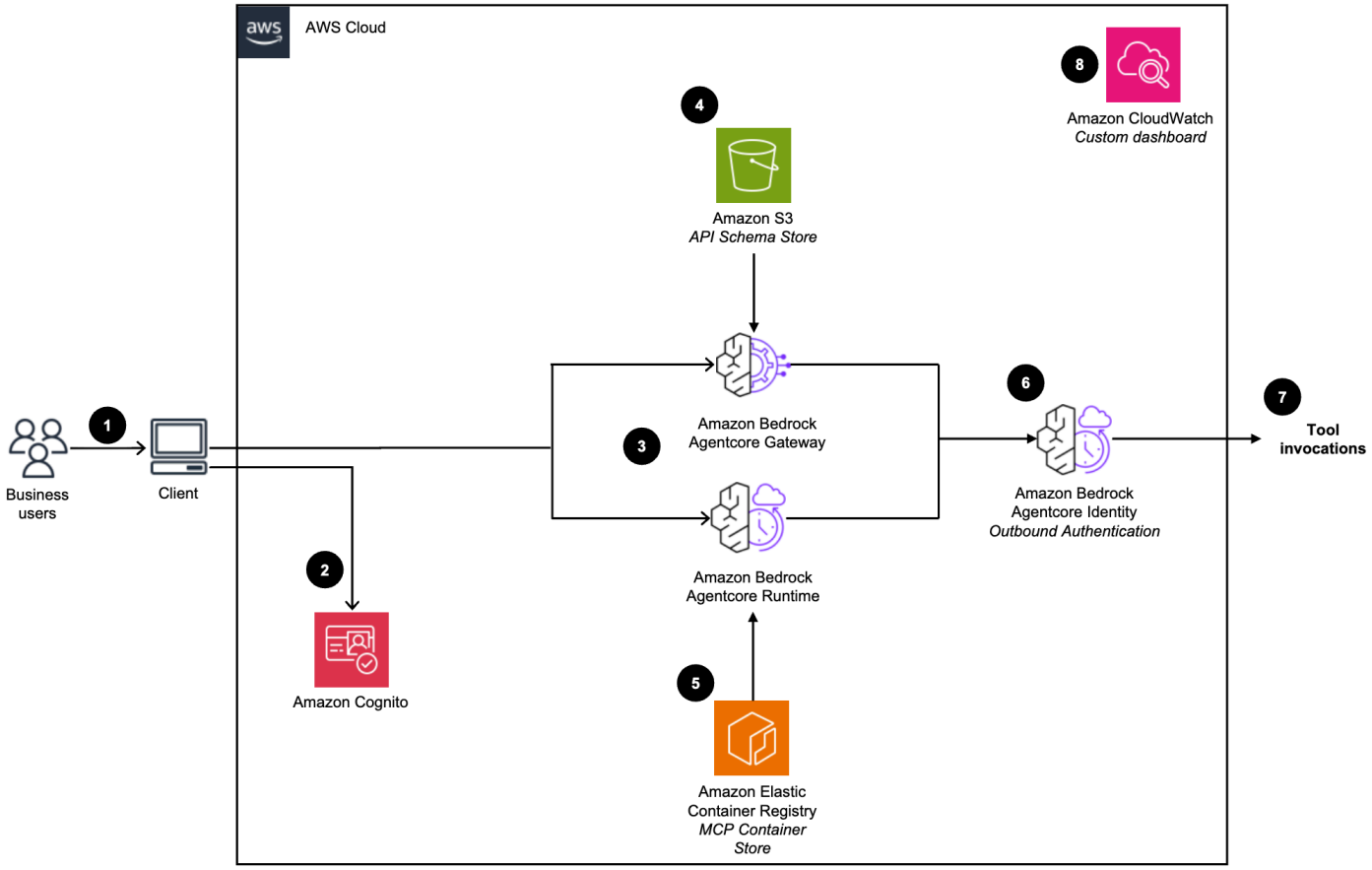
使用 AWS CloudFormation 模板部署的解决方案组件的高级流程如下：

1. 管理员用户使用部署仪表板部署用例。企业用户登录用例用户界面。
2. CloudFront 提供托管在 S3 存储桶中的 Web 用户界面。
3. 网页用户界面利用了使用 API Gateway 构建的 WebSocket 集成。API Gateway 由自定义 Lambda 授权方函数提供支持，该函数会根据身份验证用户所属的 Amazon Cognito 群组返回相应的 AWS 身份和访问管理 (IAM) 策略。该策略存储在 DynamoDB 中。
4. Amazon Cognito 对用户进行身份验证并支持 CloudFront 网页用户界面和 API Gateway。
5. 来自业务用户的传入请求将从 API Gateway 传递到亚马逊 SQS 队列，然后传递到 AWS Lambda 函数。该队列支持 API Gateway 到 Lambda 集成的异步操作。队列将连接信息传递给 Lambda 函数，然后该函数会将结果直接发布回 API Gateway websocket 连接，以支持长时间运行的推理调用。
6. AWS Lambda 函数使用亚马逊 DynamoDB 根据需要获取用例配置
7. AWS Lambda 函数使用用户输入和任何相关的用例配置，构建请求负载并将其发送到已配置的 Amazon Bedrock Agent，以实现用户意图。
8. 当响应从 Amazon Bedrock Agent 返回时，Lambda 函数会通过 API WebSocket Gateway 将响应流回以供客户端应用程序使用。
9. 该解决方案使用 Amazon CloudWatch 从各种服务中收集操作指标，以生成自定义控制面板，使您可以监控部署的性能和运行状况。
10. 如果启用了反馈收集，则可以使用利用 Amazon API Gateway 的 REST API 终端节点来收集用户反馈。
11. 支持 lambda 的反馈使用其他特定于用例的元数据来补充已提交的反馈，并将数据存储在 Amazon S3 中，供用户日后分析和报告。 DevOps

> **Note**
>
> 如果您选择在 Amazon VPC 中部署此解决方案，则数据将在您的私有网络内路由。

MCP 服务器用例

描绘 MCP 服务器用例架构



MCP 服务器用例允许在 Amazon Bedro AgentCore ck 上部署和管理模型上下文协议服务器。MCP 服务器为 AI 应用程序提供了一个标准化接口，用于访问工具、资源和企业数据源。

该解决方案支持两种部署方法：

- 网关方法：将现有 Lambda 函数、 APIs REST 或外部 MCP 服务器封装为 MCP 工具，自动处理协议转换
- 运行时方法：从 Amazon ECR 映像部署自定义容器化 MCP 服务器

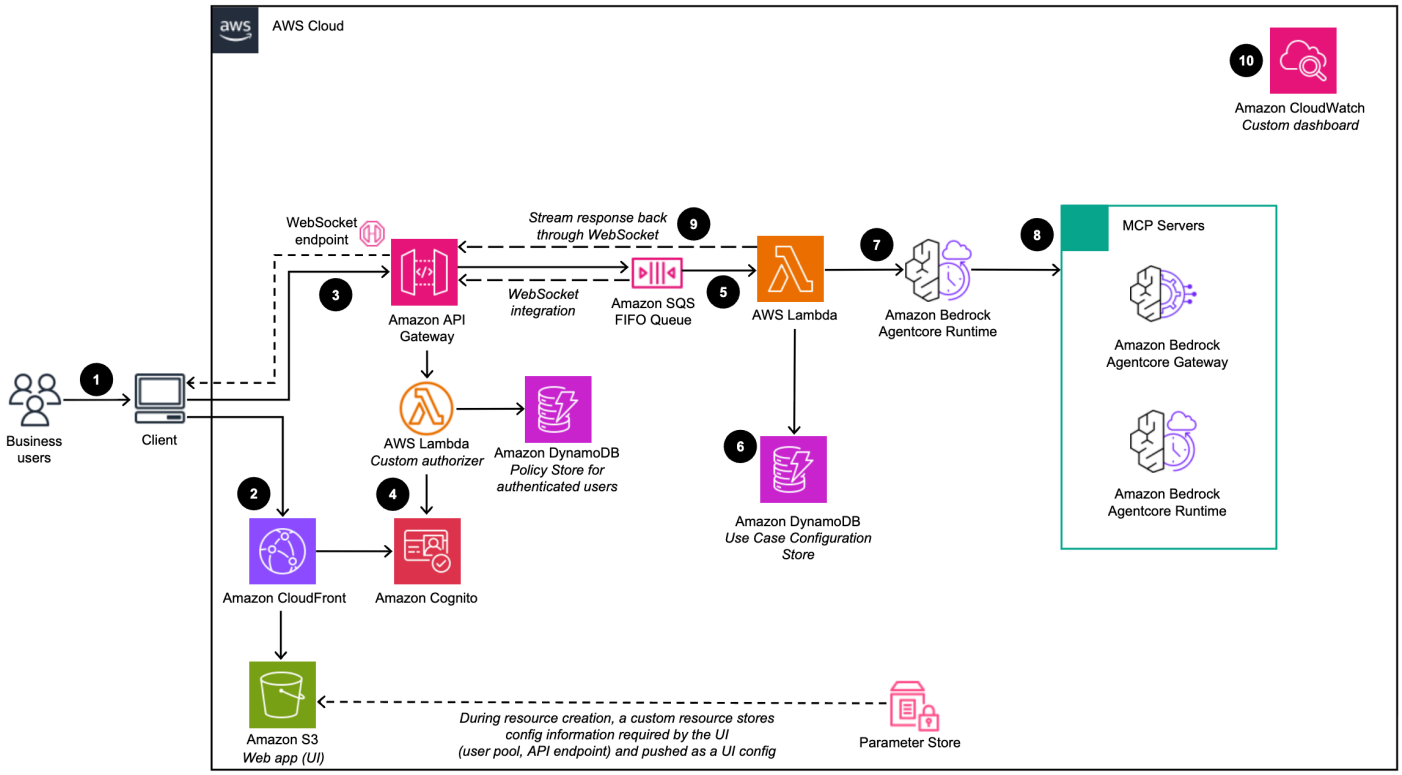
MCP 服务器部署的高级流程如下：

1. 管理员用户使用部署仪表板部署 MCP 服务器用例，选择网关或运行时部署方法。
2. 此操作已通过 Amazon Cognito 进行身份验证。
3. 对于网关部署，该解决方案创建了一个 Amazon Bedrock AgentCore Gateway，用于将现有 Lambda 函数或外部 MCP 服务器转换为 APIs符合 MCP 标准的工具。对于运行时部署，该解决方案使用提供的 ECR 映像在 Amazon Bedrock R AgentCore untime 上部署容器化 MCP 服务器。

4. 网关部署从其在 Amazon S3 中的上传位置检索必要的API/Lambda/Smithy架构，或者直接连接到 MCP 服务器 URL 终端节点。
5. 运行时部署从 Amazon 弹性容器注册表 (ECR) 检索用户提供的容器化 MCP 服务器
6. MCP 服务器装有 Amazon Bedro AgentCore ck Identity 客户端 OAuth
7. MCP 服务器在 /mcp 端点提供相关工具供代理发现。
8. Amazon 从 MCP 服务器部署中 CloudWatch 收集操作指标和日志，用于监控和故障排除。

代理生成器用例

描绘代理生成器架构



使用 AWS CloudFormation 模板部署的 Agent Builder 组件的高级流程如下：

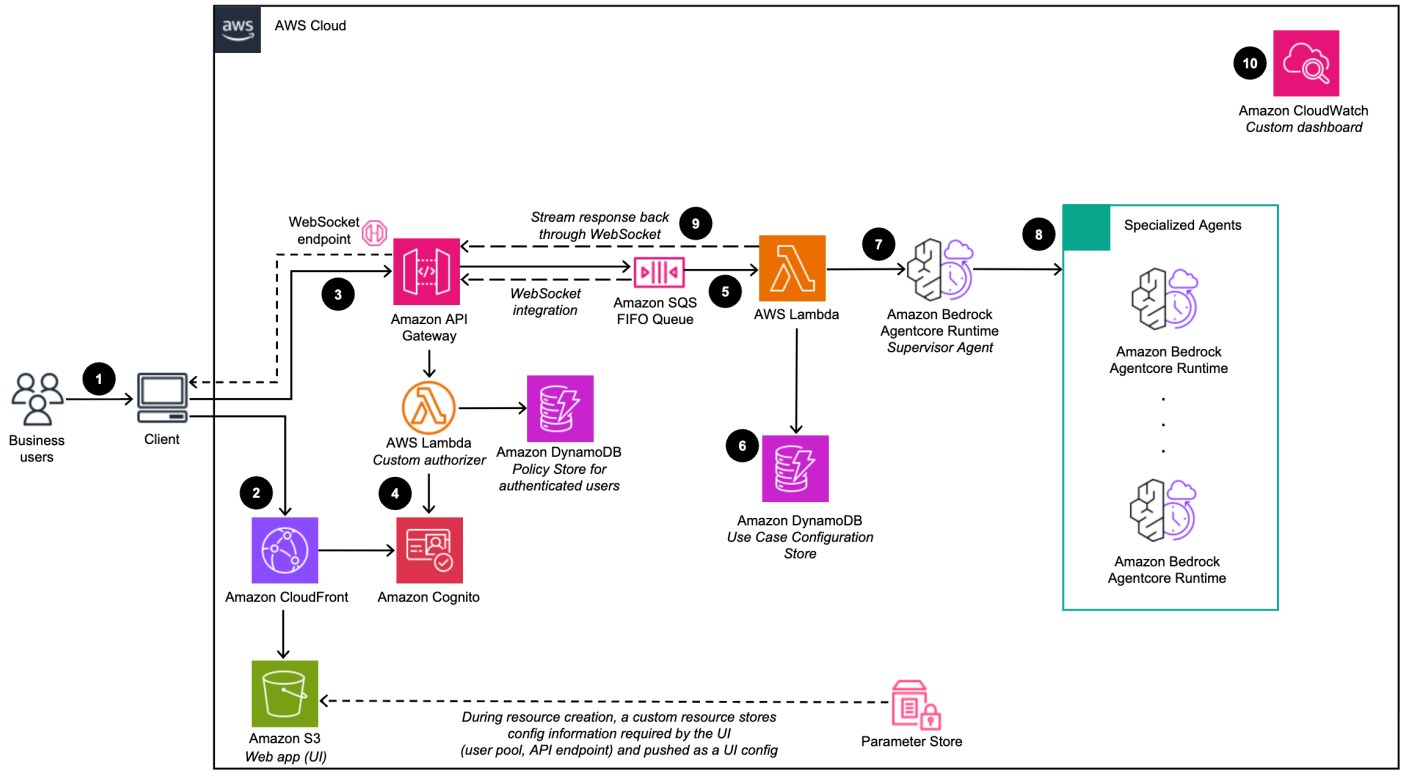
1. 管理员用户使用部署仪表板部署用例。企业用户登录用例用户界面。
2. CloudFront 提供托管在 S3 存储桶中的 Web 用户界面。
3. 网页用户界面利用了使用 API Gateway 构建的 WebSocket 集成。API Gateway 由自定义 Lambda 授权方函数提供支持，该函数会根据身份验证用户所属的 Amazon Cognito 群组返回相应的 AWS 身份和访问管理 (IAM) 策略。该策略存储在 DynamoDB 中。

4. Amazon Cognito 对用户进行身份验证并支持 CloudFront 网页用户界面和 API Gateway。
5. 来自业务用户的传入请求将从 API Gateway 传递到亚马逊 SQS 队列，然后传递到 AWS Lambda 函数。该队列支持 API Gateway 到 Lambda 集成的异步操作。队列将连接信息传递给 Lambda 函数，然后该函数会将结果直接发布回 API Gateway websocket 连接，以支持长时间运行的推理调用。
6. AWS Lambda 函数从 DynamoDB 检索代理配置。
7. AWS Lambda 函数使用用户输入和任何相关的用例配置，生成请求负载并将其发送给在 A mazon Bedro AgentCore ck Runtime 上运行的代理。
8. 代理连接到关联的 MCP 服务器，并将工具注册到 strands 代理实例。然后，代理会根据工具描述和任务要求自主选择和执行操作。
9. 当响应从 Amazon Bedrock AgentCore 运行时返回时，Lambda 函数会通过 API WebSocket Gateway 将响应流回以供客户端应用程序使用。

> **Note**
>
> - 代理处理仅限于 Lambda 执行超时（15 分钟）。

工作流程生成器用例

描绘工作流生成器架构



使用 AWS CloudFormation 模板部署的 Workflow Builder 组件的高级流程如下：

1. 管理员用户使用部署仪表板部署工作流，选择要包含为专业代理的 Agent Builder 代理。
2. CloudFront 提供托管在 S3 存储桶中的 Web 用户界面。
3. 网页用户界面利用了使用 API Gateway 构建的 WebSocket 集成。API Gateway 由自定义 Lambda 授权方函数提供支持，该函数会根据身份验证用户所属的 Amazon Cognito 群组返回相应的 AWS 身份和访问管理 (IAM) 策略。该策略存储在 DynamoDB 中。
4. Amazon Cognito 对用户进行身份验证并支持 CloudFront 网页用户界面和 API Gateway。
5. 来自业务用户的传入请求将从 API Gateway 传递到亚马逊 SQS 队列，然后传递到 AWS Lambda 函数。该队列支持 API Gateway 到 Lambda 集成的异步操作。
6. AWS Lambda 函数从 DynamoDB 检索工作流程配置，包括专门的代理生成器代理列表。
7. Lambda 使用用户输入和工作流程配置，向托管主管代理的 A mazon Bedrock AgentCore 运行时发送请求。
8. 主管代理在 AgentCore 运行时环境中创建所有专门的 Agent Builder 代理的本地实例。这些专业代理使用 “代理即工具” 模式注册为工具。然后，主管根据代理描述和任务要求自主选择工作并将其委派给专业代理。

9. 主管代理汇总来自专业代理的结果并制定最终响应，然后将其返回到 Lambda，然后通过 API Gateway Websocket 流式传输回客户端应用程序。

> **Note**
>
> - 工作流程处理仅限于 Lambda 执行超时（15 分钟）。

AWS Well-Architected 的设计注意事项

该解决方案是根据 AWS Well-Architected Fram ework 中的最佳实践设计的，可帮助客户在云中设计和运行可靠、安全、高效且经济实惠的工作负载。

此部分介绍在构建该解决方案时如何应用 Well-Architected Framework 的设计原则和最佳实践。

卓越运营

本节介绍我们是如何使用卓越运营支柱的原则和最佳实践来设计此解决方案的。

- 我们 infrastructure-as-code使用 Amazon 构建了解决方案 CloudFormation。
- Lambda 函数将自定义指标推送到 CloudWatch 自定义 CloudWatch 控制面板，以监控解决方案的运行状况。
- 解决方案组件高度模块化，可灵活选择要部署的组件。

安全性

本节介绍我们是如何使用安全性支柱的原则和最佳实践来设计此解决方案的。

- 部署控制面板和所有用例均已通过 Amazon Cognito 进行身份验证和授权。
- 所有服务间通信都使用 AWS IAM 角色。
- 所有解决方案角色都遵循最低权限访问权限；也就是说，仅授予所需的最低权限。
- 包括 S3 存储桶、DynamoDB 和 Amazon Kendra 在内的所有数据存储都处于静态加密状态。

可靠性

本节介绍我们是如何使用可靠性支柱的原则和最佳实践来设计此解决方案的。

- 基于无服务器范式的架构。
- 我们构建了按需扩展、横向扩展和从底层基础架构故障中自动恢复的架构。
- 该架构包括缓冲和限制请求，以免使底层端点不堪重负。

性能效率

本节介绍我们是如何使用性能效率支柱的原则和最佳实践来设计此解决方案的。

- 该解决方案使用 DynamoDB，这是一种完全托管的无服务器 NoSQL 数据库，可按需扩展。
- 该解决方案使用 Amazon S3 进行对象存储和（通过 CloudFront）托管网站，以提供低成本、可扩展和 11 9 的持久性。

成本优化

本节介绍我们是如何使用成本优化支柱的原则和最佳实践来设计此解决方案的。

- 在可能的情况下，我们将解决方案构建为使用无服务器架构；因此，您只需为实际用量付费。

可持续性

本节介绍我们是如何使用可持续性支柱的原则和最佳实践来设计此解决方案的。

- 该解决方案的模块化组件化架构使您可以灵活地自定义要为单个用例配置的资源。
- 该架构使用无服务器计算和存储，从而优化了资源利用率。
- 作为基于云的解决方案，该解决方案受益于共享资源、网络、电源冷却和物理设施。

架构详情

本节介绍构成此解决方案的组件和 AWS 服务，以及这些组件如何协同工作的架构详情。

此解决方案中的 AWS 服务

AWS 服务	说明
Amazon API Gateway	核心。该服务为部署仪表板提供 REST APIs 和用例的 WebSocket API。
AWS CloudFormation	核心。此解决方案以 CloudFormation 模板形式分发，并 CloudFormation 部署该解决方案的 AWS 资源。
Amazon CloudFront	核心。 CloudFront 提供托管在 Amazon S3 中的网页内容。
Amazon Cognito	核心。该服务处理 API 的用户管理和身份验证。
Amazon DynamoDB	核心。DynamoDB 存储部署控制面板的部署信息和配置详细信息。它将聊天记录和对话存储 IDs 在文本用例中，以启用对话历史记录和查询消歧义。
AWS Lambda	核心。该解决方案使用 Lambda 函数来： * 返回 REST 和 WebSocket API 端点 * 处理每个用例协调器的核心逻辑 * 在部署期间实现自定义资源 CloudFormation
Amazon S3	核心。Amazon S3 托管静态网页内容。
Amazon CloudWatch	支持。此解决方案将解决方案资源中的日志发布到CloudWatch 日志，并将指标发布到CloudWatch 指标。该解决方案还会创建一个CloudWatch 仪表板来查看这些数据。

AWS 服务	说明
AWS Systems Manager	支持。Systems Manager 提供应用程序级资源监控，并可视化资源操作和成本数据。也用于在参数存储中存储配置数据。
AWS WAF	支持。为了保护它，AWS WAF 部署在 API Gateway 部署之前部署。
Amazon Bedrock	可选。该解决方案利用 Amazon Bedrock 访问基础模型或定制模型、亚马逊 Bedrock Agents、Amazon Bedrock 知识库。建议使用 Amazon Bedrock 进行集成，以防止您的数据离开 AWS 网络。
Amazon Bedrock AgentCore	可选该解决方案利用 Amazon Bedrock AgentCore 来运行和支持 MCP 服务器连接以及代理生成器和工作流程用例。
Amazon Elastic Container Registry（Amazon ECR）	可选。对于代理生成器部署，ECR 存储和分发代理容器映像。该解决方案使用 ECR Pull-Through Cache 自动从 GAAB 团队的公共 ECR 存储库中检索预先构建的代理映像。
适用于 OpenTelemetry (ADOT) 的 AWS 发行版	可选。对于 Agent Builder 部署，ADOT 为代理可观察性提供自动检测工具，从而为代理操作启用分布式跟踪和结构化日志记录。
Amazon Kendra	可选。在文本用例中，管理员用户可以选择连接 Amazon Kendra 索引，将其用作与 LLM 对话的知识库。这可以用来向法学硕士注入新信息，使其能够在响应中使用这些信息。

AWS 服务	说明
亚马逊 SageMaker AI	可选。该解决方案可以与托管在您的 AWS 账户和区域中的 Amazon A SageMaker I 推理终端节点集成以进行访问 FMs ，并且是防止您的数据离开 AWS 网络的首选集成。 **Note** 您必须在推理终端节点可用的同一区域部署解决方案。
Amazon Virtual Private Cloud	可选。该解决方案提供了使用支持 VPC 的配置来部署组件的选项。在使用支持 VPC 的配置部署解决方案时，您可以选择让解决方案为您创建 VPC，也可以选择使用存在于部署解决方案的同一账户和区域中的现有 VPC（自带 VPC）。如果解决方案创建 VPC，则会创建必要的网络组件，包括子网、安全组及其规则、路由表、网络 ACLs、NAT 网关、Internet 网关、VPC 终端节点及其策略。

部署控制面板

API Gateway 自定义授权方

从表面上看，所有 API 调用（包括 RESTful 和 WebSocket 基于）都使用 API Gateway 的 Lambda 自定义授权器，以验证给定用户是否有权根据他们所属的组执行操作。此自定义授权方由 DynamoDB 表提供支持，该表包含每个群组的策略。在调用 API 时，API Gateway 会调用自定义授权方 Lambda 函数，该函数对提供的 Amazon Cognito 访问令牌进行解码，以确定该用户属于哪些用户组。然后按组名查询策略表，以返回该组的相关策略。

在每个新的用例部署中，管理员策略都会更新以存储一条新语句，允许对该用例的 API 执行 execute-API: Invoke 操作。删除用例后，相应的语句将从策略中删除。

对于为单个用例创建的群组，策略中仅存在一条语句，仅允许对该用例的 API 执行 Execute-API: Invoke 操作。

由于这种结构，属于用例组的任何用户都可以访问该用例的 API。也可以将单个用户手动添加到多个群组，以允许该用户使用多个用例。

> **Warning**
>
> 如果您想向现有用户组授予对新用例的访问权限，也可以在策略表中手动编辑给定组的策略。用例组会在删除用例时删除（即使您进行了手动编辑），因此在删除用例时请谨慎行事。

如果用例堆栈是独立部署的（不使用部署控制面板），则会为该部署创建一个 Amazon Cognito 用户池，其中包含有权访问该用例的 API 的单个用户。此用户池仅属于此用例，不与其他独立部署共享。

文本用例

直播支持

在聊天应用程序中，延迟是实现响应式用户体验的重要指标。法学硕士推断可能需要几秒钟到几分钟，这给如何最好地向客户提供内容带来了挑战。因此，一些LLM提供商允许将响应流式传输回呼叫者。无需等待整个推理完成后再返回响应，而是在每个令牌可用时返回。

为了支持此功能的使用，文本用例被设计为使用 WebSocket API 来支持聊天体验。 WebSocket 这是通过 API Gateway 部署的。使用 WebSocket API 可以在聊天会话开始时创建连接，并通过该套接字流式传输响应。这使前端应用程序能够提供更好的用户体验。

> **Note**
>
> 即使模型提供流媒体支持，这也不一定意味着该解决方案能够通过 WebSocket API 将响应流回来。该解决方案需要启用自定义逻辑，以支持每个模型提供商的流媒体。如果直播可用，管理员用户将能够在部署时使用 enable/disable 此功能。

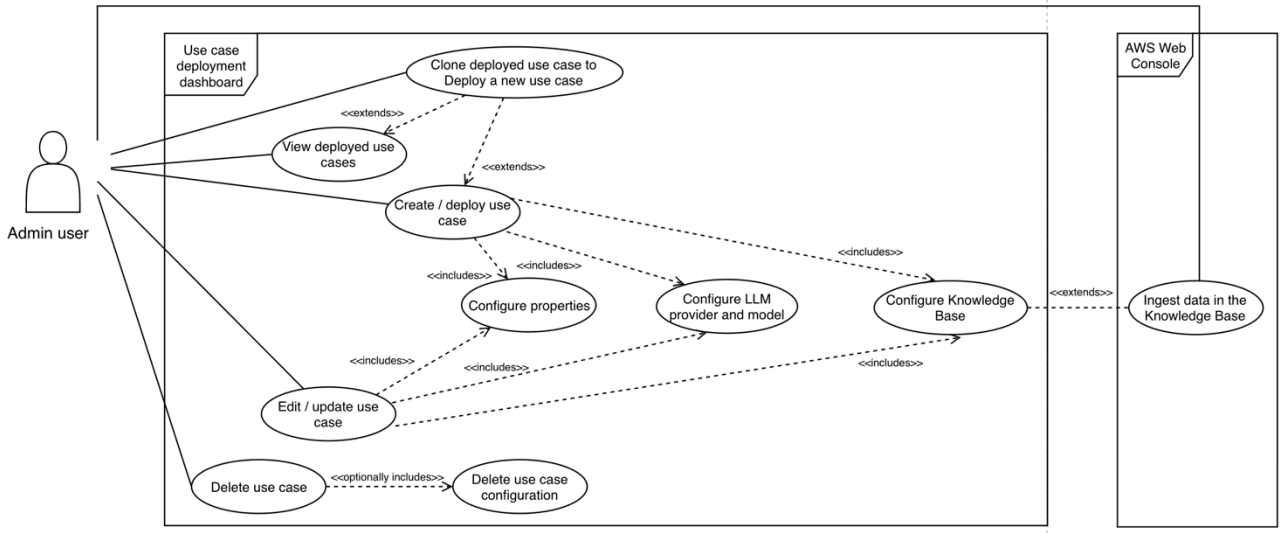
AWS 上的生成式 AI 应用程序生成器解决方案的工作原理

管理员用户主要与部署仪表板交互以查看、创建和管理新的和现有的用例部署。通过此控制面板，管理员用户可以访问以下操作：

- 查看部署列表
- 创建新部署

- 编辑现有部署
- 克隆部署的配置以创建新部署
- 删除部署（通过 CloudFormation 删除取消配置资源）
- 永久删除部署的配置详细信息

描绘部署仪表板管理员用户的用例图

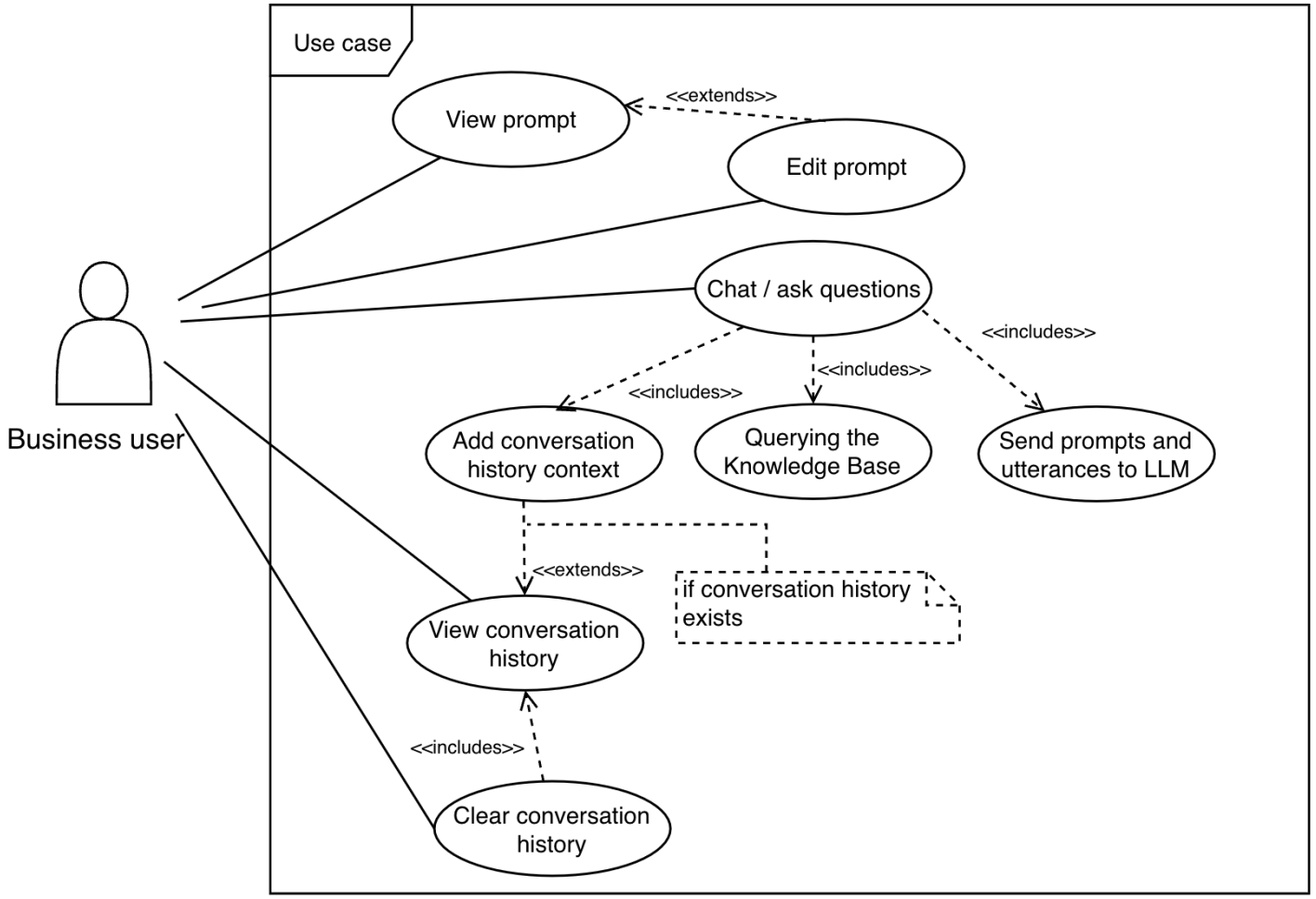


> **Note**
>
> 管理员用户可能无法直接访问 AWS 控制台。在这种情况下，管理员用户必须与用户合作以支持诸如将数据提取到 Kendra 知识库之类的操作。 DevOps

对于文本用例，业务用户可以访问用户界面，使他们能够与法学硕士聊天。此配置的细节由管理员用户配置的部署设置控制。在文本用例中，业务用户可以访问以下操作：

- 通过聊天界面发送消息
- 查看对话历史记录
- 清除对话历史记录
- 查看提示
- 编辑提示

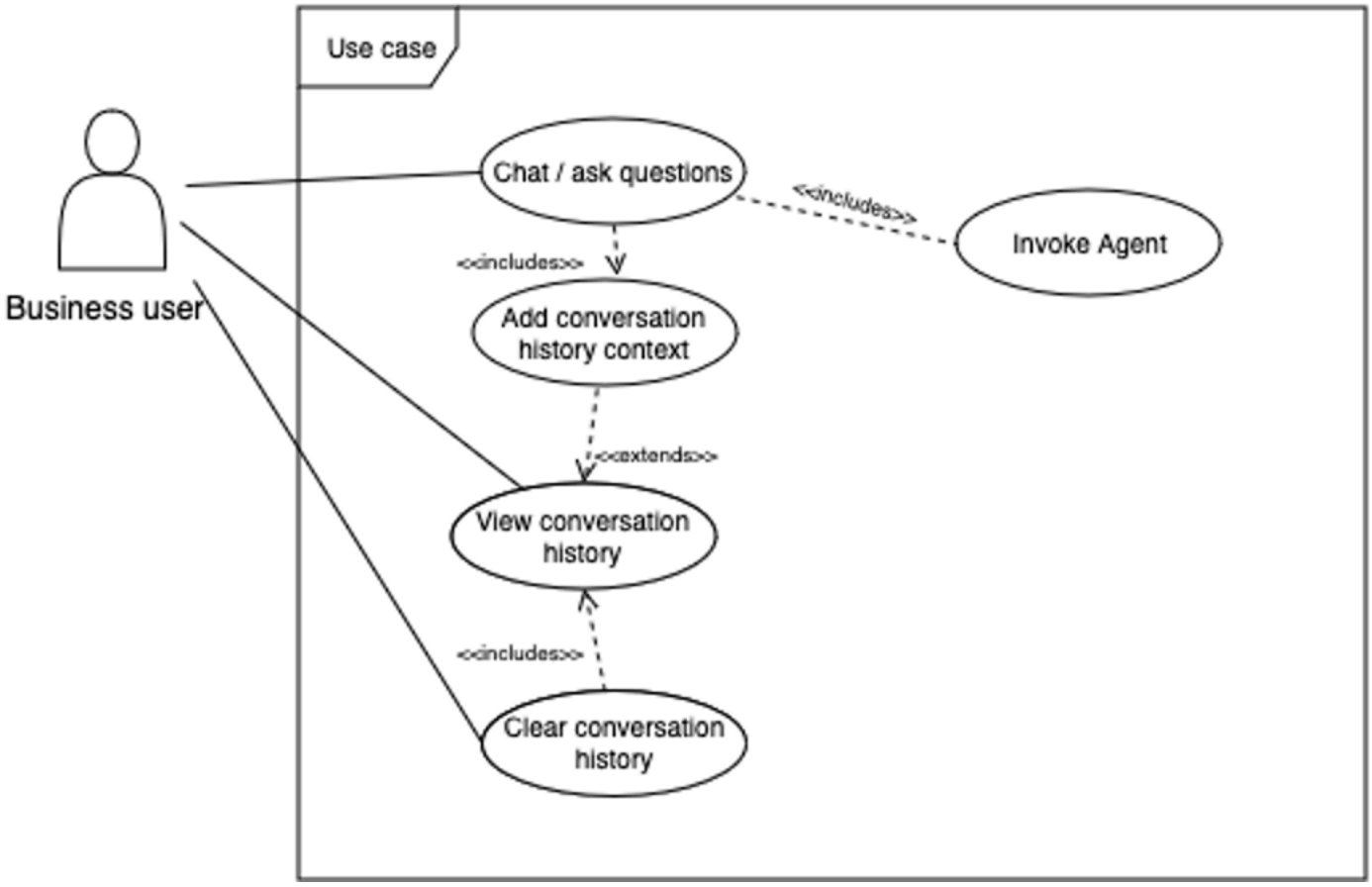
描绘了文本用例的业务用户的用例图



通过 Bedrock Agent 用例，企业用户可以访问用户界面，与配置的 Amazon Bedrock Agent 聊天。管理员用户可以在部署设置中配置这些细节。在 Bedrock Agent 用例中，业务用户可以访问以下操作：

- 通过聊天界面发送消息
- 查看对话历史记录
- 清除对话历史记录

描绘了 Bedrock Agent 用例的业务用户的用例图



代理生成器

代理生成器提供了一个平台，用于在 Amazon Bedrock 上创建、部署和管理可用于生产的 AI 代理。AgentCore本节介绍技术组件和实现细节。

AgentCore 整合

Agent Builder 使用基于配置的部署方法和预先构建的代理映像，以实现快速、安全和可扩展的代理部署。

预先构建的代理映像

代理容器镜像由 GAAB 团队在 CI/CD 管道期间构建，并发布到公共 ECR 存储库。每个图像版本都与 GAAB 解决方案版本相关联（例如，v4.0.0 →: v4.0.0）。 gaab-strands-agent图片基于 Strands SDK，包括：

- 代理运行时环境
- MCP 客户端集成
- 内存管理功能
- OpenTelemetry 仪器

ECR 直通缓存

该解决方案使用 ECR Pull-Through Cache 自动将代理映像从公共 ECR 存储库分发到客户的私有 ECR。这项 AWS 托管的服务：

- 首次拉取时缓存图像（延迟 2-5 分钟）
- 消除了自定义图像复制逻辑
- 为后续部署提供本地映像可用性
- 为每个部署创建唯一的缓存规则以避免冲突

配置存储

代理配置与现有用例配置一起存储在 DynamoDB 中。每种配置包括：

- 系统提示模板
- 模型提供商和模型 ID
- 模型参数（温度、max_tokens）
- MCP 服务器参考和端点
- 内存设置（切换长期内存）
- 部署元数据

镜像版本注册表

DynamoDB 表可跟踪可用的代理映像版本及其 URIs缓存，从而实现版本管理和向后兼容。

代理配置

系统提示

系统提示定义了代理的行为、个性和能力。管理员用户可以：

- 通过 Agent Builder 用户界面编辑默认模板
- 包括工具使用说明和响应格式说明
- 随时重置为默认模板

型号选择

Agent Builder 支持 v4.0.0 中的 Amazon Bedrock 模型：

- 模型提供商：Amazon Bedrock（v4.0.0 中唯一的选项）
- 模型选择：Claude、Nova 和其他贝德罗克模型
- 模型参数：温度、max_tokens、top_p 和特定于模型的设置

MCP 服务器集成

模型上下文协议服务器为代理提供访问企业工具和数据的权限：

- 通过 GET /mcp API 端点发现服务器
- 无需更改代码即可进行动态配置
- 身份验证和端点管理
- 工具能力暴露于特工

流媒体和处理

实时直播

Agent Builder 使用从 AgentCore 桥接到的服务器发送事件 (SSE) WebSocket 进行实时响应流：

- Lambda 函数建立与运行时的 SSE 连接 AgentCore
- 直播已桥接到 API Gateway WebSocket
- 允许向客户端传送 token-by-token响应
- 为长时间运行的请求保持连接

处理限制

v4.0.0 中的代理处理仅限于 Lambda 执行超时：

- 最长处理时间：15 分钟
- 同步处理模型
- 适用于对话代理和中等工作流程
- 计划在 v4.1+ 中提供扩展的异步支持

内存管理

短期记忆

默认情况下，所有使用自定义的代理都处于启用状态 MemoryHookProvider：

- 通过 Strands 的回调处理程序捕获对话事件
- 按 actorID 和 sessionID 进行组织以实现上下文隔离
- 维护会话中的对话背景
- 自动与 AgentCore 内存集成

长期记忆

使用 strands_tools 中的 AgentCore 记忆工具的可选功能：

- 只需在 Agent Builder UI
- 使用默认设置的语义记忆策略
- 通过自然工具调用由代理控制访问
- 存储跨会话提取的见解
- 使用 conversationID 作为会话 ID

可观测性

AWS OpenTelemetry 发行版 (ADOT)

在容器构建过程中，会自动检测代理：

- 为代理操作自动生成跟踪
- 跨服务边界的分布式跟踪
- 带关联的结构化日志 IDs

- 与 CloudWatch 交易搜索集成

身份验证流程

用户使用经自定义 Lambda 授权机构验证的 JWT 令牌通过 Amazon Cognito 进行身份验证，这些授权机构根据用户组从 DynamoDB 检索 IAM 策略。

工作流程生成器

Workflow Builder 通过创建一个主管代理，使用代理即工具委托模式来协调多个代理生成器代理，从而实现多代理编排。

工作流程架构

关键组件

- 主管代理：接收用户请求和委托给专业代理的 Entrypoint 代理
- 专业代理：Agent Builder 用例注册为主管的工具
- 代理注册表：存储代理配置和元数据的 DynamoDB 表
- 编排层：Strands SDK 实现代理即工具模式

代理实例化

创建本地代理

所有专业代理都是在同一个 AgentCore 运行时本地实例化的：

1. 从 DynamoDB 检索代理配置
2. 创建每个 Agent Builder 代理的本地实例
3. 每个代理都维护自己的 MCP 服务器连接
4. 主管代理将专业代理注册为工具
5. Strands SDK 管理代理选择和委派

规划您的部署

本节介绍规划部署时的成本、安全性、区域和配额注意事项。

> **Important**
>
> 该解决方案利用 Amazon Bedrock 作为访问人工智能生成的模型的主要服务。必须先申请模型的访问权限，然后才能在解决方案中使用它们。有关详情，请参阅 Amazon Bedrock 用户指南中的模型访问权限。

支持的 AWS 区域

> **Important**
>
> 该解决方案可以选择使用 Amazon Bedrock 和 Amazon Kendra 服务，这些服务目前并非在所有 AWS 区域都可用。您必须在提供这些服务的 AWS 地区启动此解决方案。要了解按地区划分的 AWS 服务的最新可用性，请参阅 AWS 区域服务列表。

以下 AWS 区域支持 AWS 上的生成式 AI 应用程序生成器：

区域名称	
美国东部（俄亥俄州）	加拿大（中部）
美国东部（弗吉尼亚州北部）	欧洲地区（法兰克福）
美国西部（加利福尼亚北部）	欧洲地区（爱尔兰）
美国西部（俄勒冈州）	欧洲地区（伦敦）
亚太地区（孟买）	欧洲地区（米兰）
亚太地区（首尔）	欧洲地区（巴黎）
亚太地区（新加坡）	欧洲地区（斯德哥尔摩）

区域名称	
亚太地区（悉尼）	中东（巴林）
亚太地区（东京）	南美洲（圣保罗）

> **Note**
>
> 如果在您的部署中使用在 AWS 之外访问的基础模型，请向模型提供商咨询它们 APIs 在哪些区域可用。如果 APIs 它们仅在某些区域可用，则您可能会遇到高延迟甚至超时的形式的不稳定性。同样重要的是要咨询贵组织的法律和合规团队，以评估跨区域数据的注意事项。

成本

使用此 AWS 解决方案，您只需为使用的资源付费，没有最低费用或安装费。用户需要为用于启动生成式人工智能用例的仪表板以及部署的任何用例付费。部署用例的成本取决于配置。配置示例：

1. 一个简单的部署控制面板，每月费用约为20美元。
2. 一个简单的生产就绪聊天机器人用例，使用默认设置在美国东部（弗吉尼亚北部）运行，由Amazon Bedrock提供支持，无法访问文档，每月的费用也约为200美元。
3. Amazon VPC 用例中的一个扩展系统，每天支持对成千上万个文档进行 8,000 次查询，每月费用约为 1,500 美元。用例的成本将因配置而异，例如使用不同模型提供者的文本用例，启用或不启用检索增强生成 (RAG)，等等。

工作负载说明	预计成本（美元/月）
部署控制面板的费用示例	每月 20 美元
基于文本的概念验证的样本成本 （包括部署仪表板和 1 个文本用例，每天大约 100 次互动）	40 美元/月
高度可扩展的生成式 AI 查询引擎的成本示例	每月 1,500 美元

工作负载说明	预计成本（美元/月）
（包括部署控制面板、1 个文本用例和一个 Amazon Kendra 索引，用于存放 RAG 多达 10 万个文档，在启用 VPC 的情况下，每天查询量约为 8000 次	
基于代理的概念验证的样本成本 （包括部署控制面板、1 个启用 Amazon Bedrock 知识库和亚马逊 Bedrock Guardrails 的 Bedrock Agent 用例，每天大约 100 次互动）	840 美元/月
MCP 服务器的费用示例 （包括部署控制面板、1 个 MCP 服务器用例，其中包含用于 Lambda 集成的网关方法，每天大约 100 次工具调用）	22美元/月
代理生成器的费用示例 （包括部署仪表板、1 个启用 MCP 集成和长期内存的 Agent Builder 用例，每天大约 100 次交互）	55 美元/月
工作流生成器的费用示例 （包括部署控制面板、1 个包含 3 个 Agent Builder 代理的工作流程、每天约 100 次交互）	每月 109 美元

Important

这些示例仅用于帮助您估算特定工作负载的成本。使用不同的 LLMs配置或 AWS 服务可能会改变您的成本（例如，已serverless/on-demand billing vs. provisioned/time计费）。为了管理成本，我们建议通过 AWS Cost Explorer 创建预算。价格可能会发生变化。有关完整详情，请参阅本解决方案中使用的每项 AWS 服务的定价网页。

运行部署控制面板的费用示例

下表提供了具有默认参数的部署控制面板的成本明细，在美国东部（弗吉尼亚北部）地区有 100 个活跃用户，为期一个月，费用约为 20 美元。

AWS 服务	Dimensions	成本 [美元]
API Gateway、DynamoDB、、 CloudFront亚马逊 S3、Lambda、Systems Manager 参数存储	在未启用缓存的情况下，每月调用 5,000 次 512 KB 的 REST	1.97 美元
Amazon Cognito	每月 100 个活跃用户启用了高级安全功能，且没有用户通过 SAML 或 OIDC 联合登录	5.55 美元
AWS WAF	通过 1 个 Web ACL 和 7 个已定义的规则（不含任何规则组）发出 10,000 个 Web 请求	12.60 美元
部署控制面板总成本		20.12 美元

基于文本的概念验证的样本成本

部署仪表板可以在给定时间部署许多用例。下表显示了在没有 RAG 的情况下部署的用例的成本明细，该用例针对 1 个企业用户每天使用 LLM 执行 100 次查询。假设已启用流式传输，查询在 WebSocket 上以短信形式发送，响应以令牌的形式流式传回。使用亚马逊 Bedrock Nova Pro 型号，运行此用例的成本约为 20 美元/月。

AWS 服务	Dimensions	成本 [美元]
API Gateway (WebSocket)、Lambda CloudFront、亚马逊 S3、AWS Systems Manager Parameter Systems Store	每天 100 次聊天互动。每条消息的平均消息大小为 32 KB，每次连接 5 分钟。	0.61 美元

AWS 服务	Dimensions	成本 [美元]
CloudWatch	开启详细模式的 1.5 GB CloudWatch 日志，用于实验	7.23 美元
Amazon DynamoDB	对话历史记录表，1 GB 存储空间 LLM 配置表，1 GB 存储空间	3.05 美元
用例成本小计（不包括 LLMs）		10.89 美元
亚马逊 Bedrock (Nova Pro)	假设每天 100 次互动： * 每天 19 万个输入代币的月度成本 = 0.152 美元 × 30 美元 * 每天 1.6 万个输出代币的每月成本 = 0.0512 美元 × 30	6.10 美元
使用 Amazon Bedrock（Nova Pro）的总申请费用	10.89 美元（用例成本）+ 6.10 美元（亚马逊 Bedrock 成本）	17.00 美元

> **Note**
>
> 这些估算中不包括对 AWS 网络之外的服务进行推理调用的费用。如果您不使用 AWS 模型提供商，请参阅 LLM 提供商的定价指南。
> AWS 服务的定价指南可在以下网址找到：亚马逊 Bedrock 定价和亚马逊 A SageMaker I 定价。

高度可扩展的生成式 AI 查询引擎的成本示例

下表提供了支持 RAG 的用例的成本明细，其中亚马逊 Bedrock 的 Nova Pro 机型作为 LLM。添加 Bedrock 知识库后，此用例的费用约为 1300 美元/月

AWS 服务	Dimensions	成本 [美元]
API Gateway (WebSocket)	每天有 8000 次聊天互动。每条消息的平均消息大小为 32 KB，每次连接 5 分钟。	38.89 美元
CloudFront	每月 240,000 个请求，其中 100 GB 的数据传输到互联网，1 GB 的数据传输到源站	8.76 美元
亚马逊 Bedrock (Nova Pro)	假设： 输入标记 = promptTemplate (400) + 上下文 (400) + ChathiStory (1080) + 查询输入标记 (20) = 1,900 输出代币 = 160（平均值） 每天有8,000笔交易， 每日输入代币成本（1,900 x 8,000 = 15,200,000 个代币 x 每个代币的价格为 0.0008/1000） 每日产出代币成本（160 x 8,000 = 1,280,000 个代币 x 每个代币的价格为 0.0032/1000） 每月费用（（12.16 美元 + 4.10 美元）x 30）	487.80 美元
CloudWatch	24 个指标，使用为日志提取的 5 GB 数据和 1 个控制面板	9.72 美元
DynamoDB	DynamoDB 表用于跟踪对话历史记录，每条记录最多 1 KB	11.70 美元

AWS 服务	Dimensions	成本 [美元]
	数据，每天读取和写入 8,000 次	
Lambda	容器大小——128 MB，512 MB 临时性 存储，2 个 Lambda 函数用于授权 容器大小-256 MB，512 MB 临时存储空间，每秒 5 个请求，平均计算时间 20 秒	20.89 美元
用例总成本		577.76 美元/月 + 知识库成本（见下文）

> **Note**
>
> 这些估算中不包括对 AWS 网络以外的任何服务进行 API 调用的费用。如果不使用 Amazon Bedrock，请参阅您的法学硕士提供商的定价指南。

添加知识库的成本

知识库成本将根据所使用的知识库类型以及（对于 Bedrock 而言）知识库使用的支持向量存储而有所不同。配置和管理知识库超出了解决方案的范围。

Amazon 基岩知识库

该解决方案不管理或预配置与 Amazon Bedrock 知识库相关的任何资源。Amazon Bedrock 不会因为使用知识库功能本身而产生费用，但是您需要为使用案例在每次查询中使用的嵌入模型的使用付费。此外，您的知识库的支持向量存储（例如，亚马逊 OpenSearch 服务中的索引或亚马逊关系数据库服务中的数据库）将产生相关成本，此处无法提供或计算。

对于上述高度可扩展的生成式 AI 查询引擎场景，此服务调用 Amazon Bedrock 嵌入模型所产生的成本如下：

AWS 服务	Dimensions	成本 [美元]
亚马逊 Bedrock（亚马逊 Titan 文本嵌入 V2）	每天 8,000 次查询，每次查询 1,900 个输入令牌 = 15,200,000 个代币 = 每天 0.30 美元。 每日费用 x 30 天 = 每月费用 9.00 美元	9.00 美元
亚马逊 OpenSearch 服务（无服务器）使用示例	带有 4 x OpenSearch 计算单元 (OCU)（最低计费）的基本无服务器配置 = 每天 23.04 美元 每日费用 x 30 天 = 691.20 美元 **Note** 这提供了一个粗略的估计，因为某些工作负载将需要更多的工作负载 OCUs，而拥有现有预配置 OpenSearch 资源的客户在此处花费的成本会更少。	691.20 美元
额外费用总额		700.20 美元

Amazon Kendra

该解决方案可以为您配置 Kendra 索引，也可以自带索引。运行适合上述高度可扩展的生成式 AI 查询引擎的配置的成本如下：

AWS 服务	Dimensions	成本 [美元]
Amazon Kendra	使用具有 0-50 个数据源的亚马逊 Kendra 企业版，每天 0-8,000 次查询，多达 100,000 个文档	1,008.00 美元

> **Note**
>
> 您可以在用例之间共享 Amazon Kendra 索引，但这可能会增加每个索引的查询数量。如果这不属于亚马逊 Kendra Enterprise 版，则将收取额外费用。

为用例启用 Amazon VPC 的增量成本

下表提供了为一分为二的使用案例启用 Amazon VPC 的成本明细 AZs。

AWS 服务	Dimensions	成本 [美元]
亚马逊 NAT 网关	假设：2 个可用区部署，每个可用区中有一个 NAT 网关。通过 NAT Gateway 处理 100 GB 的数据 730 小时，每月处理 100 GB 的数据	74.70 美元
AWS PrivateLink （VPC 终端节点）	假设：2 个可用区部署，每个可用区中有 1 个私有子网，1 个 VPC 终端节点有 2 个弹性网络接口 (ENIs)。 6 个 VPC 终端节点， ENIs 每个 VPC 终端节点 2 个，730 小时，一个月内处理 1,024 GB 的数据	97.84 美元
公共 IPv4 地址	假设：2 个可用区部署，每个可用区中有 1 个公有子网，每	7.30 美元

AWS 服务	Dimensions	成本 [美元]
	个公有子网中有一个 NAT 网关。每个 NAT 网关配置有 1 个活跃的公共网关 IPv4。 2 个活跃的公共 IPv4 地址 x 一个月 730 个小时 x 每小时 0.005 美元的费用 = 7.3 美元	
额外费用 （适用于亚马逊 VPC）		179.93 美元

使用预置吞吐量时的成本影响

预配置吞吐量成本将根据您预配置的模型类型、承诺期以及为承诺期选择的模型单位而有所不同。使用预置吞吐量会产生额外费用。

如需了解更多信息和最 up-to-date优惠的价格，您可以参阅 B edrock 定价。

使用跨区域推理的成本

使用跨区域推理不会产生额外的路由或数据传输费用。您为模型支付的每个代币价格与来源地区或主要区域相同。

基于代理的概念验证的样本成本

当您使用 Amazon Bedrock Agents 时，将根据构成代理的组件（例如支持模型和知识库（如果启用了 RAG）以及您添加的其他功能向您收费。下表显示了配置按需 Claude 3.5 Sonnet 模型、Amazon Bedrock 知识库和亚马逊 Bedrock Guardrails 的 Bedrock Agent 用例的成本明细。

与添加 Amazon Bedrock 知识库的成本类似，此解决方案不管理或配置与 Amazon Bedrock Agents 相关的资源。该解决方案也不会因使用 Amazon Bedrock 知识库而产生费用，但会产生以下费用：

- 对发送给它的每个查询使用嵌入模型
- 您的知识库的支持向量存储（例如，亚马逊 OpenSearch 服务中的索引或 Amazon RDS 中的数据库）

下表假设每天有 100 次互动，每次查询 1,900 个输入令牌和 160 个输出令牌。

> **Note**
>
> 对于此示例 Bedrock Agent 用例，如果将操作组配置为使用外部 API，则这些成本将是额外的。它们不在本表的计算范围之内。

AWS 服务	Dimensions	成本 [美元]
API Gateway (WebSocket) CloudFront、Lambda、亚马逊 S3、Systems Manager 参数存储	每天 100 次聊天互动，每条消息的平均大小为 32 KB，每次连接 5 分钟	0.61 美元
CloudWatch	1.5 GB 开启详细模式的 CloudWatch 日志，用于实验	7.23 美元
DynamoDB	适用于 1KB 记录大小和 1 GB 存储空间的 LLM 配置表	0.25 美元
费用小计（不包括 LLMs）		8.09 美元
Anthropic Claude 3.5	* 每天 19 万个输入代币（0.003/1,000 个代币）的每日费用 = 0.57 美元以上 每日成本 × 30 天 = 17.10 美元 * 每天 1.6 万个输出代币（0.015/1,000 个代币）的每日成本 = 0.24 美元以上 每日费用 × 30 天 = 7.20 美元	24.30 美元
适用于亚马逊 Bedrock 知识库的 Amazon Bedrock（亚马逊 Titan 文本嵌入 V2）	每天 19 万个输入代币（0.00002/1000 个代币）的每日费用 = 0.004 每日费用 × 30 天 = 0.12 美元	0.12 美元

AWS 服务	Dimensions	成本 [美元]
亚马逊 OpenSearch 服务（无服务器）使用示例	基本无服务器配置，4 × OpenSearch 计算单位 (OCU)（最低计费）= 每天 23.04 美元 每日费用 × 30 天 = 691.20 美元	691.20 美元
Amazon Bedrock 护栏	19 万个代币大致相当于 76 万（190,000 × 4）个字符和 3,800 个文本单元（76 万个字符/200 个） 考虑配置有内容过滤器、个人身份信息 (PII) 过滤器、敏感信息过滤器（正则表达式）和单词过滤器的护栏 每日内容过滤器成本（0.75/1000 个文本单位）+ PII 过滤器成本（0.1/1000 个文本单位）+ 敏感信息过滤器（正则表达式）+ 单词过滤器 = 2.85 美元 + 0.38 美元 + 0 美元 + 0 美元 + 0 美元 每月费用 = 每日费用 × 30 天 = 96.90 美元	96.90 美元
由 Anthropic Claude 3.5 Sonnet 支持的代理的总申请成本	8.09 美元（用例成本）+ 812.52 美元（其他代理配置）	820.61 美元

> **Note**
>
> 如果您不使用 AWS 模型提供商，请参阅 LLM 提供商的定价指南。AWS 服务的定价指南可在以下网址找到：亚马逊 Bedrock 定价和亚马逊 A SageMaker I 定价。

MCP 服务器的费用示例

MCP 服务器用例允许在 Amazon Bedro AgentCore ck 上部署和管理模型上下文协议服务器。下表显示了使用网关方法封装现有 Lambda 函数的 MCP 服务器用例的成本明细。

该解决方案管理 AgentCore 网关的部署和配置。您需要支付以下费用：

- 基础设施成本（API Gateway、Lambda、DynamoDB、S3） CloudWatch
- AgentCore 网关消耗（每次工具调用）
- Lambda 函数执行成本（适用于具有 Lambda 目标的网关方法）
- 外部 API 成本（适用于具有 API 或 MCP 服务器目标的网关方法，如果适用）

Item	计算	成本
亚马逊 API Gateway (REST API)	每天 100 次工具调用 × 30 天 = 每月 3,000 次请求	0.05 美元
AWS Lambda（编排）	每天 100 次调用 × 30 天 × 平均值 1 秒 × 512 MB = 每月 3,000 GB 秒	0.05 美元
Amazon DynamoDB	每月 3,000 个 read/write 请求 + 1 GB 存储空间	0.15 美元
Amazon CloudWatch	3,000 次调用的标准监控和日志记录	1.00 美元
Amazon S3	配置存储和日志（使用量最小）	0.25 美元
Amazon 基岩网关 AgentCore	每月 3,000 次工具调用	0.05 美元

Item	计算	成本
目标 Lambda 函数	每天 100 次调用 × 30 天 × 0.5 秒 × 128 MB = 每月 1,500 GB 秒	0.25 美元
每月总费用	1.75 美元（基础架构）+ 0.05 美元（AgentCore 网关）	1.80 美元

> **Note**
>
> 成本因部署方法（网关与运行时）、目标类型和使用模式而异。运行时方法部署会产生 AgentCore 运行时费用，而不是网关费用。外部 API 费用和自定义容器托管费用是额外的。

代理生成器的费用示例

代理生成器允许您在 Amazon Bedrock AgentCore 上创建和部署自定义代理。下表显示了配置了 Claude 3.5 Sonnet、MCP 服务器集成和启用长期内存的 Agent Builder 用例的成本明细。

该解决方案管理 AgentCore 运行时部署和配置。您需要支付以下费用：

- 基础设施成本（API Gateway、Lambda、DynamoDB、S3） CloudWatch
- AgentCore 运行时消耗（CPU 和内存小时数基于实际代理执行时间）
- 基础模型推断（输入和输出标记）
- AgentCore 记忆（短期事件和长期存储/检索）

下表假设每天 100 次交互，每次查询 1,900 个输入令牌和 160 个输出令牌，每次交互的平均代理执行时间为 5 秒。

AWS 服务	Dimensions	成本 [美元]
API Gateway (WebSocket) CloudFront、Lambda、亚马逊 S3、Systems Manager 参数存储	每天 100 次聊天互动，每条消息的平均大小为 32 KB，每次连接 5 分钟	0.61 美元

AWS 服务	Dimensions	成本 [美元]
CloudWatch	1.5 GB 开启详细模式的 CloudWatch 日志，用于实验	7.23 美元
DynamoDB	适用于 1KB 记录大小和 1 GB 存储空间的 LLM 配置表	0.25 美元
基础设施成本小计		8.09 美元
亚马逊 Bedrock 运行 AgentCore 时	* CPU：1 vCPU × 5 秒 × 100 次互动 = 125 个 vCPU-seconds/day = 0.140 vCPU-hours/day + 每日成本：0.140 × 0.0895 美元 = 0.013 美元 + 每月成本：0.013 美元 × 30 = 0.38 美元 * 内存：512 MB (0.5 GB) × 5 秒 × 100 次互动 = 250 GB-seconds/day = 0.069 GB-hours/day + 每日费用：0.069 × 0.00945 = 0.0007 美元 + 每月费用：0.0007 × 30 = 0.02 美元	0.40 美元
Anthropic Claude 3.5	* 每天 19 万个输入代币（0.003/1,000 个代币）的每日费用 = 0.57 美元 + 每日费用 × 30 天 = 17.10 美元 * 每天 1.6 万个输出代币（0.015/1,000 个代币）的每日成本 = 0.24 美元 + 每日成本 × 30 天 = 7.20 美元	24.30 美元

AWS 服务	Dimensions	成本 [美元]
亚马逊 Bedrock AgentCore Memory	* 短期记忆：100 个新 events/day × 0.25美元/1,000 个事件 = 0.025 美元/天 + 每月成本：0.025 美元 × 30 = 0.75 美元 * 长期内存存储（内置策略）：100 条记录 × 0.75/1,000 美元 = 0.075 records/month 美元/月 * 长期内存检索：100 retrievals/day × 0.50美元/1,000 美元检索 = 0.05 美元/天 + 每月费用：0.05 美元 × 30 = 1.50 美元	2.33 美元
带有 Claude 3.5 Sonnet 的 Agent Builder	8.09 美元（基础架构）+ 0.40 美元（AgentCore 运行时）+ 24.30 美元（模型）+ 2.33 美元（内存）	35.12 美元

Note

AgentCore 运行时定价是基于消耗量的。实际成本取决于：

- 代理执行时间（活动处理期间的 CPU 和内存使用情况）
- 交互次数及其复杂性
- MCP 工具使用情况（工具执行额外 CPU/memory 使用）
- 内存配置（启用短期内存与长期内存）

有关详细 AgentCore 定价，请参阅 Amazon Bedrock 定价。

> **Note**
>
> 如果使用调用外部 APIs 或服务的 MCP 服务器，则这些费用是额外的，不在此计算范围内。同样，如果使用 AgentCore 浏览器或代码解释器工具，则基于消耗的费用为每 vCPU 小时 0.0895 美元，每 GB 小时 0.00945 美元。

工作流生成器的费用示例

Workflow Builder 创建了一个主管代理，用于协调多个代理生成器代理。下表显示了包含 1 个主管代理和 3 个专门的 Agent Builder 代理的工作流程的成本明细，所有这些代理都配置了 Claude 3.5 Sonnet 并启用了长期内存。

假设：每天 100 次互动，平均每次互动 2 次代理委托，每个代理执行时间 5 秒。

AWS 服务	Dimensions	成本 [美元]
API Gateway (WebSocket) CloudFront、Lambda、亚马逊 S3、Systems Manager 参数存储	每天 100 次聊天互动，每条消息的平均大小为 32 KB，每次连接 5 分钟	0.61 美元
CloudWatch	1.5 GB 开启详细模式的 CloudWatch 日志，用于实验	7.23 美元
DynamoDB	适用于 1KB 记录大小和 1 GB 存储空间的 LLM 配置表	0.25 美元
基础设施成本小计		8.09 美元
Amazon Bedrock AgentCore Runtime（主管代理）	* CPU：1 vCPU × 5 秒 × 100 次互动 = 0.140 vCP hours/day × 30 = $0.38 * Memory: 0.5 GB × 5 seconds × 100 interactions = 0.069 GB-hours/day U-× 30 = 0.02 美元	0.40 美元
Amazon Bedrock AgentCore Runtime（3 个专业代理）	* 平均每次互动 2 个委托 = 200 个代理人 executions/day	0.79 美元

AWS 服务	Dimensions	成本 [美元]
	* CPU: 1 vCPU × 5 seconds × 200 = 0.278 vCPU-hours/day × 30 = $0.75 * Memory: 0.5 GB × 5 seconds × 200 = 0.139 GB-hours/day × 30 = 0.04 美元	
Anthropic Claude 3.5 Sonnet（主管	* 输入：19万 tokens/day × 0.003/1K = 0.57 美元/天 × 30 = 17.10 美元 * 产出：1.6K × 0.015/1K = 0.24美元/天 × 30 = 7.20 美元 tokens/day	24.30 美元
Anthropic Claude 3.5 十四行诗（特工）	* 平均每次互动 2 个代表团 * 投入：38 万 tokens/day × 0.003/1K = 1.14 美元/天 × 30 = 34.20 美元 * 产出：32K × 0.015/1K = 0.48 美元/天 × 30 = 14.40 美元 tokens/day	48.60 美元
Amazon Bedrock AgentCore Memory（主管代理）	* 短期：100 events/day × 0.25 美元/1K × 30 = 0.75 美元 * 长期存储：100 条记录 × 0.75/1K = 0.08 美元 * 长期检索：100 × 0.50美元/1K × 30 = 1.50 美元 = 1.50 美元 retrievals/day	2.33 美元
Amazon Bedrock AgentCore Memory（专业代理）	* 短期：200 events/day × 0.25 美元/1K × 30 = 1.50 美元 * 长期存储：200 条记录 × 0.75/1K = 0.15 美元 * 长期检索：200 × 0.50美元/1K × 30 = 3.00 美元 retrievals/day	4.65 美元

AWS 服务	Dimensions	成本 [美元]
包含 3 个代理的工作流生成器的应用程序总成本	8.09 美元（基础设施）+ 1.19 美元（AgentCore 运行时）+ 72.90 美元（模型）+ 6.98 美元（内存）	89.16 美元

> **Note**
>
> - 较高的委托率会按比例增加代币消耗
>
> 有关详细 AgentCore 定价，请参阅 Amazon Bedrock 定价。

安全性

当您在 AWS 基础设施上构建系统时，AWS 和您如何共同分担安全责任。这种分担责任模式减轻了您的运营负担，因为 AWS 运营、管理和控制组件，包括主机操作系统、虚拟化层和服务运行设施的物理安全。有关 AWS 安全性的更多信息，请访问 AWS 云安全性。

在 Amazon Bedrock 上使用基础模型

Amazon Bedrock 托管了一系列模型，从亚马逊 Nova 模型到其他领先的基础模型（FMs）。使用 Amazon Bedrock 时，所有模型都托管在 AWS 基础设施中。这意味着，当使用 Amazon Bedrock 作为 LLM 提供商时，您的所有推理请求都将保留在 AWS 网络中，并且网络流量不会离开您的区域。

> **Note**
>
> 通过 Amazon Bedrock 提供的所有基础模型 (FMs) 都直接托管在 AWS 管理和拥有的 AWS 基础设施上。模型提供者无法访问客户数据，例如提示和延续，也无法访问 Amazon Bedrock 服务日志。有关亚马逊 Bedrock 安全态势的更多信息，请参阅《亚马逊 Bedrock 用户指南》中的 Amazon Bedroc k 中的数据保护。

IAM 角色

IAM 角色允许客户向 AWS 云上的服务和用户分配精细的访问策略和权限。此解决方案创建 IAM 角色，这些角色向解决方案的 Lambda 函数授予创建区域资源的访问权限。

CloudWatch 日志

在部署用例时，您可以使用 “其他设置” 下的 “部署控制面板” 模型选择页面启用详细模式。Verbose 模式启用详细 CloudWatch 日志，这有助于调试和提示实验。

> **Note**
>
> 启用详细模式后，还将记录从知识库检索到的文档（如果启用了 RAG）和提示，其中可能包含敏感信息。

VPC

该解决方案为 Amazon VPC 配置提供了两个选项：

1. 让该解决方案为您构建 Amazon VPC。
2. 管理并引入您自己的 Amazon VPC 以在解决方案中使用。

让解决方案为您构建 Amazon VPC

如果您选择让解决方案构建 Amazon VPC 的选项，则默认情况下，它将部署为双可用区架构，CIDR 范围为 10.10.0.0/20。您可以选择使用 Amazon VPC IP 地址管理器 (IPAM)，每个可用区中有 1 个公有子网和 1 个私有子网。该解决方案在每个公有子网中创建 NAT 网关，并配置 Lambda 函数以在私有子网ENIs中创建。此外，此配置还会创建路由表及其条目、安全组及其规则、网络 ACLs、VPC 终端节点（网关和接口终端节点）。

管理您自己的亚马逊 VPC

在使用 Amazon VPC 部署解决方案时，您可以选择使用您的 AWS 账户和区域中的现有亚马逊 VPC。为了确保高可用性，我们建议您至少在两个可用区中提供您的 VPC。您的 VPC 还必须具有以下 VPC 终端节点及其关联的 IAM 策略，这些策略适用于您的 VPC 和路由表配置。

对于部署控制面板 Amazon VPC

1. DynamoDB 的网关终端节点。
2. S3 的网关终端节点。
3. 的@@ 接口终端节点 CloudWatch。
4. AWS 的接口终端节点 CloudFormation。

对于使用案例 Amazon VPC

1. DynamoDB 的网关终端节点。
2. S3 的网关终端节点。
3. 的@@ 接口终端节点 CloudWatch。
4. Systems Manager 参数存储区的接口端点。

> **Note**
>
> 该解决方案仅需要`com.amazonaws.region.ssm`。

5. Amazon Bedrock 的接口终端节点（基底运行时、代理运行时、）。 bedrock-agent-runtime
6. 可选：如果部署将使用 Amazon Kendra 作为知识库，则需要 A mazon Kendra 的接口终端节点。
7. 可选：如果部署将使用 Amazon Bedrock 下的任何 LLM，则需要一个适用于 Ama zon Bedrock 的接口终端节点。

> **Note**
>
> 该解决方案仅需要`com.amazonaws.region.bedrock-runtime`。

8. 可选：如果部署将使用 Amazon A SageMaker I 进行 LLM，则需要一个适用于 Amazon A SageMaker I 的接口终端节点。

> **Note**
>
> 使用自带 VPC 部署选项时，该解决方案不会删除或修改 VPC 配置。但是，它将删除解决方案在“为我创建 VPC”选项中创建的所有 VPCs 内容。因此，在跨堆栈/部署共享解决方案托管的 VPC 时，必须谨慎行事。

例如，部署 A 使用 “为我创建 VPC” 选项。部署 B 使用部署 A 创建的 VPC 使用我自己的 VPC。如果部署 A 在部署 B 之前被删除，则部署 B 将不再起作用，因为 VPC 已被删除。此外，由于部署 B 使用的是由 Lambda 函数 ENIs 创建的，因此删除部署 A 可能会出现错误并会保留剩余资源。

Amazon CloudFront

此解决方案部署了托管在 Amazon S3 存储桶中的 Web 控制台。为了帮助减少延迟和提高安全性，该解决方案包括一个具有原始访问身份的 CloudFront 分发，即提供对解决方案网站存储桶内容的公开访问权限的 CloudFront 用户。有关更多信息，请参阅《亚马逊 CloudFront 开发者指南》中的使用源站访问身份限制对 Amazon S3 内容的访问。

> **Note**
>
> CloudFront 账户级别的软配额限制为 20 个响应标头策略。出于安全考虑，此解决方案创建了自定义响应标头策略。如果您在 AWS 上部署了超过 20 个生成式 AI 应用程序构建器或其用例，则新的部署可能会因为达到配额限制而失败。

要解决此问题，您可以按照以下步骤在 AWS Service Quotas 控制台中请求增加响应标头策略配额的配额：

1. 打开 AWS Service Quotas 控制台。
2. 在导航窗格中，选择 Amazon services（亚马逊云科技服务）。
3. 搜索并选择 Amazon CloudFront。
4. 滚动到 “响应标头策略” 配额，然后选择 “请求增加配额”。
5. 按照提示请求提高您的 AWS 账户的配额限制。

通过增加响应标头策略配额，您可以确保在 AWS 上新部署的生成式 AI 应用程序构建器或其用例不会因为配额限制而失败。

配额

服务配额（也称为限制）是您的 AWS 账户使用的服务资源或操作的最大数量。

此解决方案中的 AWS 服务的配额

请确保此解决方案中实施的每项服务都有足够的配额。有关更多信息，请参阅 AWS 服务配额。

通过以下链接转到相关服务的页面。要在不切换页面的情况下查看文档中所有 AWS 服务的服务配额，请改为在 PDF 中查看服务端点和配额页面的信息。

Amazon Bedrock AgentCore 配额

对于代理生成器部署，请注意以下 Amazon B edrock AgentCore 服务配额：

配额	美国东部（弗吉尼亚州北部）	其他区域
每个账户的活动会话工作量	1000	500
每个账户的代理人总数	1000	1000
每个账户的版本	1000	1000

部署解决方案

该解决方案使用 AWS CloudFormation 模板和堆栈来自动部署。该 CloudFormation 模板指定了此解决方案中包含的 AWS 资源及其属性。 CloudFormation 堆栈预置模板中描述的资源。

部署流程概述

在启动解决方案之前，请查看本指南中讨论的成本、架构、安全性和其他注意事项。

> **Important**
>
> 如果您计划使用 Amazon Bedrock，则必须在模型可供使用之前申请访问权限。有关更多详细信息，请参阅 Amazon Bedrock 用户指南中的模型访问权限。

部署时间：大约 10 分钟

步骤 1：启动部署仪表板堆栈

步骤 2：部署用例

步骤 3：使用部署仪表板向导部署用例

步骤 4：部署后配置

或者，如果您不想使用部署仪表板用户界面或，则可以将用例与解决方案分开部署 APIs。

- 部署独立的文本用例
- 部署独立的 Bedrock Agent 用例

您也可以提供 DynamoDB 聊天配置。

> **Important**
>
> 此解决方案向 AWS 发送有关该解决方案使用情况的运营指标（“数据”）。我们使用这些数据来更好地了解客户如何使用此解决方案以及相关服务和产品。AWS 对这些数据的收集受 AWS 隐私政策的约束。

AWS CloudFormation 模板

您可以先下载此解决方案的 CloudFormation 模板，然后再进行部署。

View template

generative-ai-application-builder-on-aws.template-使用此模板启动解决方案和所有关联组件。默认配置部署了本解决方案部分的 AWS 服务中的核心和支持解决方案，但您可以自定义模板以满足您的特定需求。

> **Note**
>
> AWS CloudFormation 资源是基于 AWS Cloud Development Kit (AWS CDK) 结构创建的。

此 AWS CloudFormation 模板在 AWS 云中的 AWS 上部署生成式 AI 应用程序生成器。

步骤 1：启动部署控制面板堆栈

按照本节中的 step-by-step说明配置解决方案并将其部署到您的账户。

部署时间：大约 10 分钟

1. 登录 AWS 管理控制台并选择启动`generative-ai-application-builder-on-aws.template` CloudFormation 模板的按钮。

   Launch solution

2. 默认情况下，该模板在美国东部（弗吉尼亚州北部）区域启动。要在其他 AWS 区域启动解决方案，请使用控制台导航栏中的区域选择器。

   > **Note**
   >
   > 该解决方案使用 Amazon Kendra 和 Amazon Bedrock，它们目前并非在所有 AWS 区域都可用。如果使用这些功能，则必须在提供这些服务的 AWS 地区启动此解决方案。有关各地区的最新可用性，请参阅 AWS 区域服务列表。

3. 在创建堆栈页面上，确认 Amazon S3 URL 文本框中已有正确的模板 URL，然后选择下一步。
4. 在指定堆栈详细信息页面上，为您的解决方案堆栈分配一个名称。有关命名字符限制的信息，请参阅 AWS Identity and A ccess Management 用户指南中的 IAM 和 STS 限制。

5. 在参数下，检查该解决方案模板的参数，并根据需要进行修改。该解决方案使用以下默认值。

参数	默认值	说明
管理员用户电子邮件	No	将有权访问部署仪表板的管理员用户的电子邮件地址。如果提供，则将创建一个拥有部署和管理用例权限的 Amazon Cognito 群组和用户。您也可以使用`placeholder@example.com` 创建群组，但不能使用创建用户。有关设置用户池的信息，请参阅手动用户池配置。
VpcEnabled	No	部署控制面板是否应该部署在 VPC 中
CreateNewVpc	No	仅在可用时可VpcEnabled用`Yes`。如果值为`Yes`，则堆栈将创建 VPC 并在创建的 VPC 中部署解决方案。 如果VpcEnabledCreateNewVpc是 `Yes`No，则必须提供现有 VPC 配置（ExistingVpcId、ExistingPrivateSubnetIds、ExistingSecurityGroupIds、VpcAzs）。
IPAMPoolId	（可选输入）	您可以配置 IPAM 并提供创建的 ID 作为输入，以分配部署此堆栈时应使用的 IP 地址范围。有关 IPAM 的详细信息，请参阅 IPAM 的工作原理。

参数	默认值	说明
部署用户界面	Yes	您可以选择在没有 Web 用户界面（以及 Web 部署所需的 AWS 资源）的情况下部署部署控制面板。在这种情况下，该解决方案将部署所有基础架构，包括 REST API 端点。此选项可用于将您自己的 Web 界面与部署仪表板集成 APIs。
ExistingVpcId	（可选输入）	仅当您要在已创建的现有 VPC 中部署解决方案时才需要此选项。
ExistingPrivateSubnetIds	（可选输入）	仅当您要在已创建的现有 VPC 中部署解决方案时才需要此选项。Lambda 函数将部署在该子网中。
ExistingSecurityGroupIds	（可选输入）	仅当您要在已创建的现有 VPC 中部署解决方案时才需要此选项。确保安全组拥有出站 TCP 连接的权限。
VpcAzs	（可选输入）	仅当您要在已创建的现有 VPC 中部署解决方案时才需要此选项。
CognitoDomainPrefix	（可选输入）	仅当您想要在自己创建的现有 Amazon Cognito 用户池中部署解决方案时才需要这样做。如果您不提供值，则解决方案会生成该值。

参数	默认值	说明
ExistingCognitoUserPoolId	（可选输入）	仅当您想要在自己创建的现有 Amazon Cognito 用户池中部署解决方案时才需要这样做。
ExistingCognitoUserPoolClient	（可选输入）	仅当您想要在自己创建的现有 Amazon Cognito 用户池中部署解决方案时才需要这样做。如果您不提供值，则该解决方案会创建一个用户池客户端。只有在您提供ExistingCognitoUserPoolId值时才能提供此参数。

6. 选择 Next(下一步)。
7. 在配置堆栈选项页面上，请选择下一步。
8. 在审核并创建页面上，审核并确认设置。选中确认模板将创建 AWS Identity and Access Management (IAM) 资源的复选框。
9. 选择提交以部署堆栈。

您可以在 AWS CloudFormation 控制台的 “状态” 列中查看堆栈的状态。您将在大约 10 分钟后收到 CREATE_COMPLETE 状态。

步骤 2：部署用例

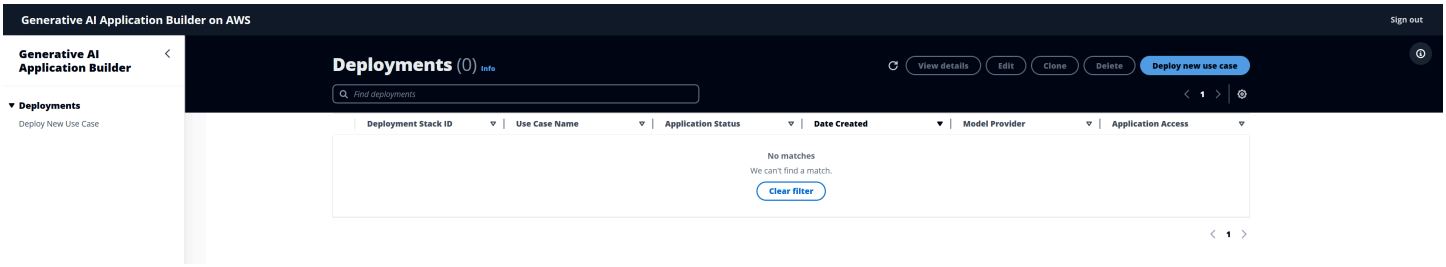
> **Important**
>
> 成功部署堆栈后，将向配置的管理员用户电子邮件发送一封注册电子邮件。使用这些凭据，管理员用户可以登录部署仪表板以使用 Web 应用程序。

> **Note**
>
> 堆栈完成后，有权访问 AWS 管理控制台的用户必须向管理员用户提供部署控制面板用户界面的 CloudFront URL。 DevOps URL 可以在 CloudFormation 堆栈的 “输出” 选项卡中找到。

1. 以管理员用户身份登录部署控制面板。
2. 在应用程序登录页面上，选择部署新用例。

   这将启动部署向导，该向导将引导您完成用例的构建。

描绘部署仪表板登录页面-全新部署



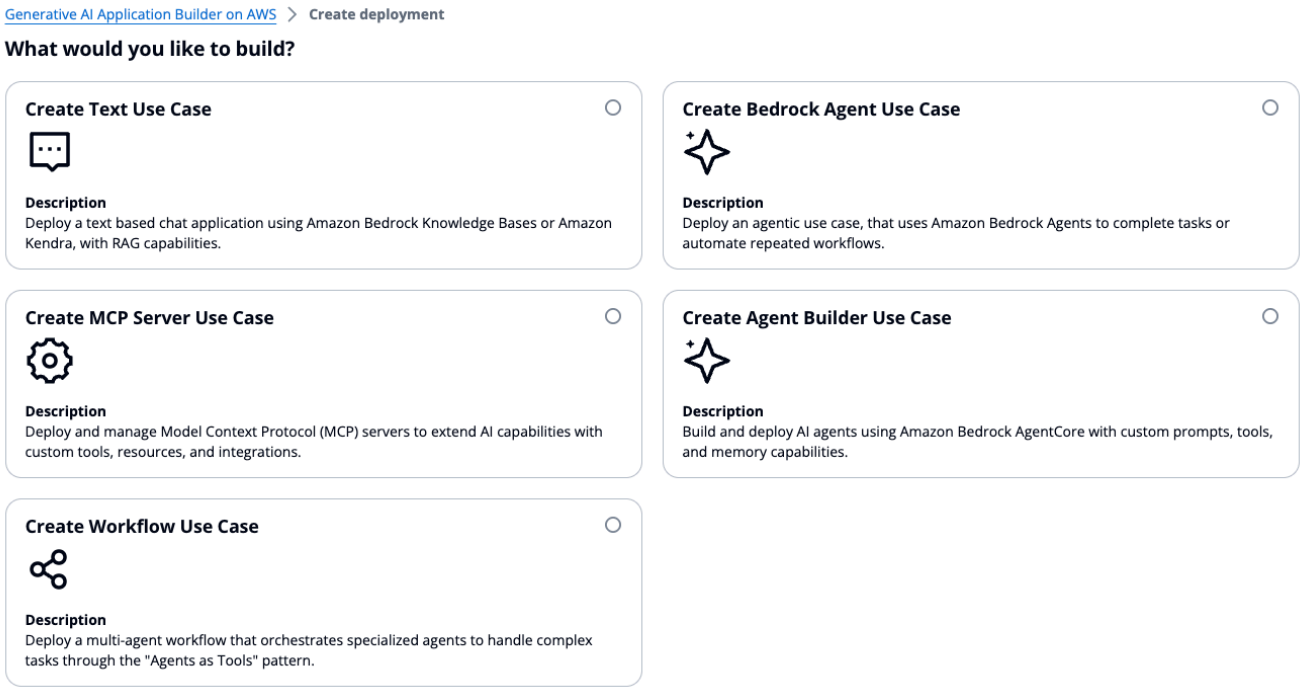
> **Note**
>
> 如果您需要在部署中添加其他用户，请参阅管理 Cognito 用户池了解更多详细信息。

步骤 3：使用部署仪表板向导部署用例

在部署仪表板向导中，您必须在以下选项之间进行选择：

- 文本用例-部署具有可选 RAG 功能的聊天应用程序
- 基岩代理用例-使用 Amazon Bedrock Agents 完成任务或自动执行重复的工作流程
- MCP 服务器-使用网关或运行时方法部署和管理 MCP 服务器
- Agent Builder- AgentCore 使用 MCP 集成和内存管理功能构建和部署自定义代理
- 工作流生成器-使用分层委托编排多个 Agent Builder 代理

显示五个选项：创建文本用例、创建 Bedrock Agent 用例、创建 MCP 服务器用例、创建 Agent Builder 用例或创建工作流程用例。



步骤 3a：部署文本用例

本节提供部署文本用例的说明。

选择用例

选择 “创建文本” 用例后，用户界面将打开 “选择用例” 屏幕。提供以下信息：

- 用例名称。
- 将该用例的默认用户添加到该用例的 Amazon Cognito 用户池并获得与其交互的权限的可选电子邮件地址。
- 是否要使用此用例部署 UI。如果您不想使用该用例部署 UI，则可以使用已部署的 API 端点与您的应用程序一起使用。

使用案例详细信息

用例详细信息步骤允许您为部署配置其他设置。

默认情况下，当解决方案部署部署控制面板时，文本用例会为您创建和配置 Amazon Cognito 用户池。该解决方案使用同一用户池中新创建的客户端对新的用例进行身份验证。但是，如果您想在用例中使用自己的 Amazon Cognito 用户池和客户端，则可以在此步骤中提供现有的用户池 ID 和客户端 ID。

Important

通过部署向导创建 Amazon Cognito 用户池后，管理员用户可以访问所有已部署的用例。如果您在部署期间提供了自己的用户池，则必须确保管理员有权访问已部署的用例。
您还需要在 Cognito 中更新应用程序客户端 URLs 中允许的回调 URLs 和允许的注销。要实现此目的，应按照以下步骤进行：

1. 导航到 Cognito 控制台
2. 选择用户池。
3. 选择您的用户池。
4. 在左侧菜单中选择 “应用程序客户端”。
5. 选择要修改的应用程序客户端。
6. 选择 “登录页面” 选项卡。
7. 选择编辑并添加您的 URLs.
8. 选择保存更改。

此外，如果您需要在用例中添加更多用户，请参阅管理 Cognito 用户池部分。

选择网络配置

此向导步骤允许您使用预先存在的或新的亚马逊虚拟私有云（Amazon VPC）部署用例。如果选择预先存在的 VPC，则需要提供一个 VPC ID、最多 16 个子网 ID 和最多 5 个安全组 IDs 才能与此 VPC 一起使用。如果您没有使用预先存在的 VPC，则将为您配置这些设置。

选择模型

在选择模型步骤中，您可以从下拉菜单中选择您的模型提供商。有两种选择：基岩和。SageMaker

如果选择 SageMaker，则可以在 SageMaker AI 控制台中创建 A SageMaker I 模型终端节点，并提供模型期望的输入架构和 LLM 响应 JSONPath 的输出。您可以参考解决方案 GitHub 存储库中提供的 “使用 SageMaker Amazon AI 作为 LLM 提供商” 部分和 SageMaker AI 负载示例。

如果您选择 Amazon Bedrock，您将看到四个选项：

- 快速入门模型-使用一系列具有不同 price/performance 特征的模型快速入门。推荐用于构建您的第一个应用程序。此选项允许您从提供的列表中选择型号名称。

- 其他基础模型-访问具有不同功能和专业的全系列基础模型。此选项允许您输入所需的 Bedrock 按需基础模型的模型 ID。
- 推理配置文件-推理配置文件利用 Bedrock 的跨区域推理，在利用率高峰期将请求路由到多个 AWS 区域，从而提高吞吐量并提高弹性。此选项允许您输入要使用的推理配置文件的 ID。
- 预配置模型-为需要一致性能的生产工作负载提供专用吞吐容量。此选项允许您输入要从 Amazon Bedrock 使用的 provisioned/custom 模型的 ARN。

模型选择步骤还允许您选择高级模型设置。有关配置 Amazon Bedrock Guardrails、Amazon Bedrock 的预配置吞吐量以及其他模型参数的详细信息，请参阅高级 LLM 设置。

跨区域推理

跨区域推理可帮助 Amazon Bedrock 用户通过使用跨不同 AWS 区域的计算来无缝管理计划外的流量爆发。要使用跨区域推理，您需要推理配置文件。推理配置文件是对一组已配置的 AWS 区域中的按需资源池的抽象。它可以将来自您的源区域的推理请求路由到该池中配置的另一个区域。这允许在多个 AWS 区域之间分配流量。这有助于在需求高峰期实现更高的吞吐量和增强的弹性。

推理配置文件以其支持的模型和区域命名。您必须从推理配置文件包含的其中一个区域调用推理配置文件。例如，如下表所示，推理配置文件 ID `us.anthropic.claude-3-haiku-20240307-v1:0` 允许在您选择的模型`us-east-1`的`us-west-2`区域上分配流量。某些模型仅在特定区域具有推理配置文件时才可用。

推理配置文件	推理配置文件 ID	包括的区域
US Anthropic Claude 3 Haiku	`us.anthropic.claude-3-haiku-20240307-v1:0`	美国东部（弗吉尼亚州北部）(`us-east-1`) 美国西部（俄勒冈州）(`us-west-2`)

如果要使用推理配置文件 ID 而不是模型 ID，则必须标识相应的推理配置文件 ID。有关更多信息，请参阅 Amazon Bedrock 用户指南中的推理配置文件支持的区域和模型。在 Amazon Bedrock 控制台中，左侧导航菜单中的跨区域推理选项提供了这些推理配置文件。 IDs

确定要使用的推理配置文件 ID 后，您可以通过执行以下步骤在 “选择模型” 阶段使用它：

1. 选择 Amazon Bedrock 作为模型提供商。

2. 选择 “推理配置文件” 单选按钮选项。
3. 在出现的文本框中输入您的推理配置文件 ID。

有关推理配置文件的更多详细信息，请参阅 A mazon Bedrock 用户指南中的通过跨区域推理提高弹性。

选择知识库

如果您想部署非检索增强生成 (RAG) 用例，则可以跳过此步骤。

但是，如果您希望在部署过程中启用 RAG，则现在可以提供预配置的 Amazon Kendra 索引 ID 或亚马逊 Bed rock 知识库 ID。您也可以创建新的 Amazon Kendra 索引以用于该解决方案。该解决方案目前支持 Amazon Kendra 和 Amazon Bedrock 知识库作为基于 RAG 的用例部署的知识库。

有关将数据提取到知识库以用于基于 RAG 的部署的指南，请参阅配置知识库部分。

高级 RAG 配置

该向导允许您选择用于RAG部署的高级选项，例如每次向知识库发送查询时要检索的文档数量，在知识库中找不到文档时LLM的静态文本响应，是否希望在LLM响应中显示文档源以进行健全性检查等。此外，您还可以为 Amazon Kendra 配置知识库的特定配置，例如基于角色的访问控制 (RBAC)，或者在使用 Amazon Serverless OpenSearch 和 Amazon B edrock 知识库时覆盖搜索类型。有关这些高级设置的更多详细信息，请参阅高级知识库设置部分。

> **Note**
>
> 您的知识库必须与部署的部署控制面板和用例堆栈位于相同的账户和区域。

选择提示和令牌限制

在此步骤中，您可以配置与 LLM 一起使用的提示。提示可能需要占位符`{input}`，例如、`{history}`和。`{context}`这些占位符指示 LLM 在哪里提取用户输入、对话历史记录以及从知识库中检索到的信息。

- 对于 Bedrock 模型提供商，必须提供系统提示，该提示符对非 RAG 用例没有限制。但是，Bedrock 模型提供者的消除歧义提示至少需要两个占位符——而且 `{input}` `{history}`
- 对于 SageMaker 模型提供者、系统和歧义消除提示，两者都需要至少两个占位符——和。`{input}` `{history}`

- 对于 RAG 用例，对于每个模型提供者，还需要`{context}`占位符。

有关更多信息，请参阅配置提示。您也可以参阅 “管理模型代币限制的提示” 部分，同时为提示选择代币限制大小。

启用多模态输入

此步骤允许您为用例启用多模式输入功能。启用后，用户可以上传和发送图像和文档以及文本查询。

支持的文件类型和限制：

- 图片：每封邮件最多 20 张图片。每张图片的大小不得超过 3.75 MB，高度和宽度不得超过 8,000 像素。支持的格式：png、jpeg、gif、webp
- 文档：每封邮件最多 5 个文档。每个文档的大小不得超过 4.5 MB。支持的格式：pdf、csv、doc、docx、xls、xlsx、html、txt、md

如何使用多模态输入：

1. 在用例部署期间启用该MultimodalEnabled参数
2. 在聊天界面中，用户可以通过两种方式上传文件：
   - 点击聊天输入框中的上传按钮，或
   - 将文件直接拖放到聊天界面中
3. 文件上传到 Amazon S3 并由所选模型进行处理
4. 上传的文件将在 48 小时后自动删除

文件状态跟踪：

DevOps 用户可以监控 DynamoDB 中的文件元数据，包括上传时间和处理状态。文件可以具有以下状态：

- 待处理-文件上传已开始但尚未完成。这是生成预签名 URL 时的初始状态。
- 已@@ 上传-文件已成功上传到 S3，可供模型处理。
- d eleted-文件已被用户删除，不应再访问该文件进行处理。
- in valid-文件未通过验证检查（例如，文件类型不匹配或安全验证失败）。

处于待处理状态且从未上传的文件将在其 TTL 到期时自动清除。模型只能处理状态为已上传的文件。

部署控制面板输出中提供了 S3 多模式存储桶和 DynamoDB 元数据表，其中分别包含密钥和。`MultimodalDataBucketName` `MultimodalDataMetadataTable`

> **Note**
>
> 并非所有型号都支持多模态输入。启用此功能之前，请确保所选型号支持图像和文档处理。请参阅 Amazon Bedrock 文档中的支持基础模型，以查看哪种型号支持图像作为输入模式。

> **Important**
>
> 用户上传的文件按照 48 小时生命周期策略存储在 Amazon S3 中。有关已上传文件的元数据存储在 Amazon DynamoDB 中，对话历史记录的 TTL 为 24 小时。

查看并部署

完成此步骤后，查看您选择的设置并选择 “部署用例”。然后，新的用例将部署并显示在您的部署仪表板视图中，以便进一步管理。

步骤 3b：部署 Bedrock Agent 用例

Bedrock Agent 用例提供了一种强大而安全的机制，用于在您的用例中调用 Amazon Bedrock Agent。该功能允许开发人员无缝集成人工智能驱动的自治代理的功能，这些代理可以在各种基础模型、数据源、软件应用程序和用户对话中协调和执行多步骤任务，同时保持强大的安全措施。

先决条件

在创建 Amazon Bedrock 代理之前，请确保您具备以下条件：

1. 在 AWS 上部署生成式 AI 应用程序生成器的 AWS 账户，可以访问 Amazon Bedrock 控制台。
2. 创建和管理 Amazon Bedrock 代理的适当 IAM 权限。

创建 Amazon 基岩代理

有关创建代理的详细说明，请参阅 Amazon Bedrock 用户指南中的手动创建和配置代理。您可以配置选项，例如：

- 给您的代理的说明（提示）
- 知识库，用于根据用户的输入查找其他信息

- 代理的内存，允许代理在多个会话中记住信息（最长 30 天）

成功创建 Amazon Bedrock 代理后，您可以继续前往 AWS Bedrock Agent 上的生成式 AI 应用程序生成器用例向导流程。为此，请在 “部署” 仪表板上选择 “部署新用例”，然后选择 “创建 Bedrock Agent 用例”。按照向导并使用以下步骤配置用例。

选择用例

此步骤与前面描述的文本用例相同。

选择网络配置

此步骤与前面描述的 “文本” 用例相同

选择代理

在此步骤中，您必须提供您创建的 Amazon Bedrock 代理的代理 ID 和别名 ID。

步骤 3c：部署 MCP 服务器用例

MCP（模型上下文协议）服务器用例使您能够部署和管理可与 AI 模型和代理集成的 MCP 服务器。MCP 服务器提供了一种向 AI 应用程序公开工具、资源和功能的标准化方式。您可以利用现有 Lambda 函数创建 MCP 服务器，也可以使用容器 APIs映像托管自定义 MCP 服务器。

先决条件

在部署 MCP 服务器用例之前，请确保具备以下条件：

1. 在 AWS 上部署生成式 AI 应用程序生成器的 AWS 账户。
2. 创建和管理 Amazon Bedrock AgentCore 资源的适当 IAM 权限。
3. 根据您选择的创建方法：
   - 对于网关方法（Lambda/API/MCP服务器）：Lambda 函数、API 终端节点及其相应的架构文件（Lambda 的 JSON 格式 APIs）或 MCP 服务器 OpenAPI/Smithy URL 终端节点
   - 对于运行时方法 (ECR)：推送到亚马逊 ECR 的 Docker 容器镜像，其中包含你的 MCP 服务器实现

MCP 服务器的创建方法

该解决方案支持两种创建 MCP 服务器的方法：

从 Lambda、API 或 MCP 服务器创建（网关方法）

此方法创建一个 MCP 网关，该网关封装了现有 Lambda 函数 APIs、REST 或外部 MCP 服务器，使它们可以作为 MCP 工具进行访问。网关处理 MCP 和现有服务之间的协议转换。

- Lambda 目标：通过提供函数 ARN 和描述函数格式的 JSON 架构文件来整合现有的 Lambda 函数 input/output
- OpenAPI 目标：使用 O APIs penAPI 规范（JSON 或 YAML 格式）集成 REST，支持 2.0 或 API 密钥身份验证 OAuth
- Smithy 目标：使用 Smithy 模型文件（.smithy 或.json 格式）进行集成 APIs
- MCP 服务器目标：通过 URL 端点直接连接到外部 MCP 服务器，无需部署新基础架构即可集成现有 MCP 服务器

您可以在单个 MCP 网关中配置多个目标（最多 10 个），每个目标代表不同的工具或功能。

从 ECR 映像托管（运行时方法）

此方法通过 Amazon ECR 映像部署容器化 MCP 服务器。当您的自定义 MCP 服务器实现需要作为独立服务运行时，请使用此方法。

- 提供 ECR 映像 URI（必须包含标签，例如`:latest`或`:v1.0.0`）
- （可选）配置环境变量以将配置传递给您的容器
- 容器必须实现 MCP 协议并公开所需的端点

部署 MCP 服务器

要部署 MCP 服务器用例，请在 “部署” 仪表板上选择 “部署新用例”，然后选择 “创建 MCP 服务器用例”。按照向导并使用以下步骤配置用例。

选择用例

此步骤与前面描述的文本用例相同。

选择网络配置

当前，仅启用公共访问，不支持 VPC 进行网络配置。

创建 MCP 服务器

在此步骤中，您将配置 MCP 服务器部署：

MCP 服务器的创建方法

在两种创建方法之间进行选择：

- 从 Lambda、API 或 MCP 服务器创建：利用现有 Lambda 函数、API 规范或外部 MCP 服务器终端节点创建 MCP 网关
- 从 ECR 镜像托管：从容器映像部署自定义 MCP 服务器

> **Note**
>
> 部署后无法更改创建方法。如果需要切换方法，则必须部署新的 MCP 服务器用例。

网关配置（适用于Lambda/API/MCP服务器方法）

如果您选择了网关方法，请配置一个或多个目标：

1. 目标名称（必填）：用于标识此目标配置的友好名称
2. 目标描述（可选）：对该目标的作用进行简要描述
3. 目标类型：选择要配置的目标类型：
   - Lambda：适用于 AWS Lambda 函数
   - OpenAPI：适用于带有 Op APIs enAPI 规范的 REST
   - Smithy：For with S APIs mithy 模型定义
   - MCP 服务器：用于通过 URL 端点直接连接到外部 MCP 服务器
4. 架构文件（必填）：上传描述目标的架构文件：
   - 对于 Lambda：描述 input/output 格式的 JSON 架构文件。有关创建 Lambda 工具架构的详细信息，请参阅 Amazon Bedrock 开发者指南中的 Lambda 工具架构。 AgentCore
   - 对于 OpenAPI：OpenAPI 规范文件（JSON 或 YAML）。有关 OpenAPI 架构要求的详细信息，请参阅《亚马逊 Bedrock 开发者指南》中的 OpenAPI 架构。 AgentCore
   - 对于 Smithy：Smithy 模型文件（.smithy 或.json）。有关构建 Smithy 目标的详细信息，请参阅 Amazon Bedrock AgentCore 开发者指南中的构建 Smithy 目标。
5. Lambda 函数 ARN（Lambda 目标所必需的）：要集成的 Lambda 函数的 ARN
6. MCP 服务器 URL（MCP 服务器目标所必需的）：要连接的外部 MCP 服务器的 URL 端点。网址必须经过正确编码，MCP 服务器必须支持 MCP 协议版本为 2025-06-18 的工具功能。有关更多信息，请参阅 Amazon Bedrock AgentCore 开发者指南中的 MCP 服务器目标。
7. 出站身份验证（OpenAPI 目标需要进行身份验证）：为 REST API 调用配置身份验证：

- 身份验证类型：选择 OAuth 2.0 或 API 密钥
- 出站身份验证提供商 ARN：亚马逊 Bedrock 令牌库中凭证提供商的 ARN AgentCore
- 其他配置：视身份验证类型而定：
  - 对于 OAuth 2.0：配置作用域和自定义参数
  - 对于 API 密钥：指定位置（标题或查询参数）、参数名称和可选前缀

您可以通过选择添加其他目标来添加多个目标（最多 10 个）。每个目标代表您的 MCP 服务器公开的单独工具或功能。

ECR 配置（用于 ECR 映像方法）

如果您选择了运行时方法，请提供：

1. ECR 镜像 URI（必填）：亚马逊 ECR 中 Docker 镜像的完整 URI
   - 格式：`account-id.dkr.ecr.region.amazonaws.com/repository-name:tag`
   - 映像必须与您的部署位于同一 AWS 区域
   - 需要一个标签（例如`:latest`，`:v1.0.0`）
2. 环境变量（可选）：配置键值对以在运行时传递到容器
   - 使用它们来提供配置、凭证或自定义标志
   - 您最多可以添加 10 个环境变量

查看并部署

配置 MCP 服务器后，查看所选设置并选择 “部署用例”。然后，新的 MCP Server 用例将部署并显示在您的部署仪表板视图中，以便进一步管理。

> **Note**
>
> MCP 服务器部署在 Amazon Bedrock 中创建资源 AgentCore，包括网关、运行时和工作负载身份。这些资源由解决方案自动管理，当您删除用例时，这些资源将被清除。

步骤 3d：部署代理生成器用例

代理生成器使您能够在 Amazon Bedrock 上创建、配置和部署可用于生产的 AI 代理。 AgentCore此功能可通过系统提示、型号选择、MCP 服务器集成和内存管理来完全控制代理行为。

部署过程与文本用例基本相同，但有一些明显的区别。

选择用例

此步骤与前面描述的文本用例相同。

使用案例详细信息

此步骤与前面描述的文本用例相同。

配置代理

在此步骤中，您将配置核心代理设置，包括系统提示符、可用的 MCP servers/Strands 工具和内存。

系统提示符

系统提示符定义了代理的行为、个性和能力。你可以：

- 编辑默认系统提示模板
- 使用 “重置为默认值” 按钮恢复原始模板
- 包括工具使用说明和响应格式说明

MCP 服务器集成（可选）

配置模型上下文协议服务器，让您的代理能够访问企业工具和数据：

1. 在下拉列表中从可用的 MCP 服务器中进行选择
2. 查看代理可以访问的开箱即用工具

> **Note**
>
> 在部署之前，必须对 MCP 服务器进行配置和访问。有关服务器设置说明，请参阅 MCP 文档。

内存配置

配置代理如何维护上下文和知识：

- 短期内存：默认情况下，所有代理都处于启用状态。维护会话中的对话背景。
- 长期记忆：切换以启用跨会话提取和存储见解。使用带有语义 AgentCore 记忆策略的内存。

查看并部署

完成此步骤后，查看您选择的设置并选择 “部署用例”。代理生成器部署通常会在 10-15 分钟内完成。然后，新的用例将显示在您的部署仪表板视图中，以便进一步管理。

步骤 3e：部署工作流程用例

通过 Workflow Builder，您可以使用 “代理即工具” 委派模式创建主管代理，以协调多个代理生成器代理。此功能允许您通过重复使用现有的 Agent Builder 部署来构建复杂的多代理工作流。

部署过程遵循与 Agent Builder 类似的模式，包括代理发现和选择的其他步骤。

选择用例

此步骤与前面描述的文本用例相同。

使用案例详细信息

此步骤与前面描述的文本用例相同。

配置主管代理

在此步骤中，您将配置将协调专门的 Agent Builder 代理的 Supervisor 代理。

系统提示符

系统提示符定义主管代理如何将工作委托给专业代理。你可以：

- 编辑默认系统提示模板
- 包括代理选择和委派说明
- 定义如何汇总来自多个代理的结果
- 使用 “重置为默认值” 按钮恢复原始模板

> **Note**
>
> 系统提示应清楚地描述何时以及如何使用每个专业代理。代理描述对于正确委派至关重要。

模型选择

为主管代理选择基础模型。主管代理使用此模型来：

- 了解用户请求
- 选择合适的专业代理
- 协调代理执行
- 汇总和格式化回复

选择专业代理

在此步骤中，您可以选择主管可以将工作委托给哪些 Agent Builder 代理。

添加代理

1. 单击 “添加代理” 打开代理选择对话框
2. 从列表中选择一个或多个 Agent Builder 代理
3. 查看将提供给主管的代理描述
4. 确认选择

> **Note**
>
> - 作为专业代理，工作流至少需要 1 个 Agent Builder 用例
> - 在创建工作流之前，必须成功部署所有专业代理

查看并部署

查看工作流程配置，包括：

- 主管代理系统提示和型号
- 专业代理商名单
- 内存设置

选择 “部署用例”。工作流程部署通常会在 15-20 分钟内完成。新的工作流程将显示在您的部署仪表板视图中，以便进一步管理。

步骤 4：部署后配置

本节为部署后配置解决方案提供了建议。

Amazon S3 存储桶版本控制、生命周期策略和跨区域复制

此解决方案不对其创建的存储桶强制执行生命周期配置。我们建议执行下列操作：

- 为生产部署设置生命周期配置。有关详细信息，请参阅《Amazon 简单存储服务用户指南》中的设置存储桶的生命周期配置。
- 根据部署解决方案的用例为 Amazon S3 存储桶启用版本控制和跨区域复制。

亚马逊 DynamoDB 备份

此解决方案将 DynamoDB 用于多种用途（请参阅本解决方案中的 AWS 服务）。该解决方案不为其创建的表启用备份。我们建议为生产部署创建此功能的备份。有关详细信息，请参阅备份 DynamoDB 表和使用 AWS Backup for DynamoD B。

Amazon CloudWatch 控制面板和警报

该解决方案在中部署了自定义控制面板， CloudWatch 以根据已发布的自定义指标和 AWS 服务指标呈现图表。我们建议根据部署解决方案的用例创建 CloudWatch 警报并添加通知。

亚马逊 CloudWatch 日志

Lambda 日志配置为永不过期，API Gateway 日志配置为有效期为 10 年。您可以更新相应日志组的到期时间，使其与企业的记录保留政策保持一致。

带有 TLS v1.2 或更高版本证书的自定义网域

该解决方案使用 CloudFront部署网页用户界面和边缘优化 API Gateway。 CloudFront的域不强制执行 TLS v1.2 或更高版本的证书。我们建议使用 A mazon Route 53 创建自定义域，使用 AWS Certificate Manager 创建证书，或者如果您的组织拥有现有证书，则使用现有证书。

有关更多详情，请参阅亚马逊 Route 53 开发者指南和在 API Gateway 中为自定义域选择最低的 TLS 版本。

使用 Amazon Kendra 进行扩展

该解决方案允许使用 Amazon Kendra 对摄取的文档执行 NLP 支持的智能搜索。对于较大的工作负载，您可以使用以下 CloudFormation 参数增加 Amazon Kendra 的容量：

参数	默认值	说明
Amazon Kendra 的额外查询容量	0	索引的额外查询容量和GetQuerySuggestions容量。索引的额外容量单位每天可提供大约 8,000 个查询。
亚马逊 Kendra 额外存储容量	0	索引的额外存储容量。单个容量单位可提供 30 GB 的存储空间或 100,000 个文档，以先到者为准。
亚马逊 Kendra 版	Developer	Amazon Kendra 提供开发者版和企业版来创建索引。有关亚马逊 Kendra 版本之间差异的更多信息，请参阅亚马逊 Kendra 定价。

要修改这些 CloudFormation 参数的值，请在部署堆栈时选择相应的值。有关查询和存储容量单位的更多信息，请参阅调整容量。

> **Note**
>
> 如果未在启用 RAG 的情况下部署文本用例，则不会使用或创建 Amazon Kendra 索引。

使用 Idp 联盟设置 SSO

此解决方案允许与支持基于 SAML 或 OIDC 的身份联合的外部身份提供商集成。部署解决方案时，它会为部署控制面板和个人用例创建 Amazon Cognito 用户池和个人应用程序客户端集成。根据外部 Idp，按照《Amazon Cognito 开发者指南》的 “为您的用户池配置身份提供商” 部分中提供的步骤进行操作，然后为部署控制面板或您想要设置 SSO 的用例选择应用程序客户端集成。

要将用户组信息传递到基于 RAG 的架构中的知识库或矢量存储，您需要将用户组从外部 Idp 映射到 Amazon Cognito 用户组。该解决方案提供了一个初始脚手架 Lambda 函数触发器，该触发器要与令牌生成前阶段进行映射。Lambda 函数有 group_mapping.json 文件，必须更新该文件才能提供组映射。请参阅使用 Amazon Cognito 支持的 Lambda 触发器自定义用户池工作流程。

手动配置用户池

如果您选择在部署期间不发送管理员或默认用户电子邮件，则必须在 Amazon Cognito 中手动创建相应的用户组以确保权限正确：

1. 对于部署控制面板，`Admin`在您的 Cognito 用户池中创建一个名为的群组。
2. 对于每个用例，`${UseCaseName}-Users`在您的 Cognito 用户池中创建一个名为的群组，其中`${UseCaseName}`是您部署的用例的名称。

授权机制需要这些组才能正常工作。您要向其授予访问权限的任何用户都必须添加到相应的群组中。

如果通过，`placeholder@example.com`则将创建 Cognito 群组，但您仍必须创建关联用户并将其分配到群组。

自定义登录屏幕

此解决方案使用 Amazon Cognito 托管的用户界面来呈现登录页面。要自定义内置登录页面，请参阅 Amazon Cognito 开发者指南中的自定义内置登录和注册网页。

其它安全注意事项

根据您部署解决方案的用例，请查看以下安全建议：

- 客户托管的 AWS KMS 加密密钥-该解决方案默认使用 AWS 托管的 AWS KMS 密钥，因为这些密钥无需额外付费即可获得。查看您的使用案例，确定是否应更新解决方案以使用客户托管的 AWS KMS 密钥。
- API Gateway 限制规则-该解决方案在 API Gateway 上使用默认的限制规则进行部署。根据您的用例和预期的交易量，我们建议您为配置限制。 APIs有关详细信息，请参阅 Amazon API Gateway 开发者指南中的限制 API 请求以提高吞吐量。
- 启用 AWS CloudTrail-作为推荐的安全措施，可以考虑 CloudTrail在部署解决方案的 AWS 账户中启用 AWS，以便在 AWS 账户中记录 API 调用。有关详细信息，请参阅 A WS CloudTrail 用户指南。
- 漂移检测-我们建议在 CloudFormation 堆栈上配置偏差检测，以识别已部署的解决方案堆栈的无意或恶意更改，并收到通知。有关详细信息，请参阅实现警报以自动检测 AWS CloudFormation 堆栈中的偏差。
- Cognito JSON Web Tokens (JWTs)-该解决方案使用亚马逊 Cognito 发行的 REST API 终 JWTs 端节点进行身份验证。我们为解决方案配置了 ID 令牌和访问令牌的有效期为五分钟。当用户注销时，他们生成新令牌的能力将被撤销（刷新令牌被撤销）。但是，在当前令牌到期之前，任何向 API 端

点发出的请求都将成功通过身份验证，因为它们具有有效的令牌。查看您的用例的安全注意事项并调整令牌有效期。

自定义生命周期策略：

对于生产部署，请根据您的保留要求查看和调整生命周期策略。请参阅《Amazon 简单存储服务用户指南》中的设置存储桶的生命周期配置。

多模式文件存储和生命周期

如果您为用例启用了多模式输入功能（MultimodalEnabled设置为Yes），则该解决方案会创建一个 Amazon S3 存储桶来存储上传的文件，并创建一个 DynamoDB 表来跟踪文件元数据。

默认生命周期策略：

- S3 文件：48 小时后自动删除
- DynamoDB 元数据：记录将在 24 小时后过期（对话历史记录 TTL）

安全注意事项：

- 文件按用例 ID、用户 ID、会话 ID 和消息 ID 进行分区，而文件则以 UUID 名称存储。UUID 到文件名的映射可在 DynamoDB 元数据表中找到
- 用户只能访问他们在自己的对话中上传的文件
- 文件类型验证是使用幻数检测进行的
- 我们建议启用适用于 S3 的 Amazon GuardDuty 恶意软件防护，以扫描上传的文件中是否存在恶意内容

部署独立的文本用例

按照本节中的 step-by-step说明配置解决方案并将其部署到您的账户。

部署时间：大约 10-30 分钟

1. 登录 AWS 管理控制台并选择按钮启动要部署的 CloudFront 模板。

BedrockChat。模板	
SageMakerChat。模板	

2. 默认情况下，该模板在美国东部（弗吉尼亚州北部）区域启动。要在其他 AWS 区域启动解决方案，请使用控制台导航栏中的区域选择器。

   注意：此解决方案使用 Amazon Kendra 和 Amazon Bedrock，它们目前并非在所有 AWS 区域都可用。如果使用这些功能，则必须在提供这些服务的 AWS 地区启动此解决方案。有关各地区的最新可用性，请参阅 AWS 区域服务列表。

3. 在创建堆栈*页面上，确认*Amazon S3 网址*文本框中是否有正确的模板 URL，然后选择* 下一步。
4. 在*指定堆栈详细信息*页面上，为您的解决方案堆栈指定一个名称。有关命名字符限制的信息，请参阅 AWS Identity and A ccess Management 用户指南中的 IAM 和 STS 限制。
5. 在参数下，检查该解决方案模板的参数，并根据需要进行修改。该解决方案使用以下默认值。

UseCaseUUID	*<_Requires input_>*	长度 UUIDv4 为 36 个字符，用于标识应用程序中已部署的用例。
UseCaseConfigRecordKey	*<_Requires input_>*	与包含聊天提供商 Lambda 在运行时所需的配置的记录对应的密钥。表中的记录必须具有与该值匹配的关键属性，以及包含所需配置的配置属性。此记录将由部署平台填充（如果正在使用）。对于此用例的独立部署，需要在中定义的表中手动创建的UseCaseConfigTableName条目。
UseCaseConfigTableName	*<_Requires input_>*	堆栈将从表中读取配置，关键是这个名字 UseCaseConfigRecordKey

ExistingRestApiId	（可选输入）	要使用的现有 API Gateway REST API ID。如果未提供，则将创建一个新的 API Gateway REST API。通常在从 “部署” 仪表板部署时提供。 注意：使用现有 APIs 可以帮助减少资源重复，并简化 APIs 何时需要部署多个独立用例的管理。在 APIs 为独立用例提供现有路径时，您有责任确保 API 配置为具有预期模型的所需路由。必须配置所需的预先配置的 /details 路由（在聊天期间获取用例详细信息），也可以配置 /feedback 路由（如果设置FeedbackEnabledYes为启用收集 LLM 聊天响应的反馈）。此外 ExistingApiRootResourceIdExistingCognitoUserPoolId，还ExistingCognitoGroupPolicyTableName必须提供、。
ExistingApiRootResourceId	（可选输入）	要使用的现有 API Gateway REST API 根资源 ID。REST API 根资源 ID 可以从 AWS 控制台获取，方法是在 API 的 “资源” 部分中选择根资源 (/)。然后，资源 ID 将显示在资源详细信息面板中。或者，您也可以在 REST API 上运行描述 API 调用以查找根资源 ID。

FeedbackEnabled	No	如果设置为 “否”，则部署的用例堆栈将无法访问反馈功能。
ExistingModelInfoTableName	（可选输入）	包含模型信息和默认值的表的 DynamoDB 表名。由部署平台使用。如果省略，则将根据房屋模型的默认值创建一个新表。
DefaultUserEmail	`placeholder@example.com`	此用例的默认用户的电子邮件。已为该电子邮件创建一个 Amazon Cognito 用户来访问该用例。如果未提供，则不会创建 Cognito 群组和用户。您也可以使用`placeholder@example.com` 创建群组，但不能使用创建用户。有关设置用户池的信息，请参阅手动用户池配置。
ExistingCognitoUserPoolId	（可选输入）	UserPoolId此用例将使用该用户池进行身份验证的现有 Amazon Cognito 用户池。通常在从 Deployment 控制面板部署时提供，但在独立部署此用例堆栈时可以省略。
CognitoDomainPrefix	（可选输入）	如果要为 Cognito 用户池客户端提供域，请输入一个值。如果您不提供值，则部署将生成一个值。

ExistingCognitoUserPoolClient	（可选输入）	提供用户池客户端（应用程序客户端）以使用现有的客户端。如果您不提供用户池客户端，则将创建一个新的用户池客户端。只有在提供了现有的用户池 ID 时，才能提供此参数。
ExistingCognitoGroupPolicyTableName	（可选输入）	包含用户组策略的 DynamoDB 表的名称。这是由自定义授权方在用例的 API 上使用的。通常，您可以在从部署平台部署时提供输入，但在独立部署此用例堆栈时可以省略该用例堆栈。
RAGEnabled	`true`	如果设置为 true，则部署的用例堆栈将使用为提供 RAG 功能而创建的 Amazon Kendra 索引。如果设置为`false`，则用户直接与 LLM 交互。
KnowledgeBaseType	Bedrock	用于 RAG 的知识库类型。只有在为时 RAGEnabled 才设置`true`。可以是 Bedrock 或 Kendra。 注意：只有在RAGEnabled为真时才相关。

ExistingKendraIndexId	（可选输入）	用于该用例的现有 Kendra 索引的索引 ID。如果未提供任何索引，并且KnowledgeBaseType是 Kendra，则将为您创建一个新索引。 注意：只有在 “是” `true` 和 RAGEnabledKnowledgeBaseType“是” 时才相关`Kendra`。
NewKendraIndexName	（可选输入）	要为此用例创建的新 Kendra 索引的名称。仅在未提供ExistingKendraIndexId时适用。 注意：只有在RAGEnabled为真且KnowledgeBaseType是 Kendra 时才相关。
NewKendraQueryCapacityUnits	0	将为此用例创建的新 Amazon Kendra 索引的其他查询容量单位。仅在未提供ExistingKendraIndexId时才适用，请参阅CapacityUnitsConfiguration。 注意：只有在 “是” `true` 和 RAGEnabledKnowledgeBaseType“是” 时才相关`Kendra`。

NewKendraStorageCapacityUnits	0	将为此用例创建新的 Amazon Kendra 索引的额外存储容量单位。仅在未提供ExistingKendraIndexId时才适用，请参阅CapacityUnitsConfiguration。 注意：只有在 “是” true 和 RAGEnabledKnowledgeBaseType“是” 时才相关Kendra。
NewKendraIndexEdition	（可选输入）	用于为此用例创建新的亚马逊 Kendra 索引的 Amazon Kendra 版本。仅在未提供时ExistingKendraIndexId适用，请参阅 Amazon Kendra 版本。 注意：只有在 “是” true 和 RAGEnabledKnowledgeBaseType“是” 时才相关Kendra。
BedrockKnowledgeBaseId	（可选输入）	要在 RAG 用例中使用的基础知识库的 ID。如果提供了ExistingKendraIndexId或，NewKendraIndexName则无法提供。 注意：只有在 “是” true 和 RAGEnabledKnowledgeBaseType“是” 时才相关Bedrock。
VpcEnabled	No	堆栈资源是否应部署在 VPC 内。

CreateNewVpc	No	如果您希望解决方案为您创建新 VPC 并用于此用例，请选择Yes。 注意：只有在VpcEnabled是的情况下才相关Yes。
IPAMPoolId	（可选输入）	如果您想使用 Amazon VPC IP 地址管理器分配 CIDR 范围，请提供要使用的 IPAM 池 ID。 注意：只有在 “是” Yes 和 VpcEnabledCreateNew Vpc“是” 时才相关No。
ExistingVpcId	（可选输入）	用于该用例的现有 VPC 的 VPC ID。 注意：只有在 “是” Yes 和 VpcEnabledCreateNew Vpc“是” 时才相关No。
ExistingPrivateSubnetIds	（可选输入）	用于部署 Lambda 函数 IDs 的现有私有子网的子网列表，以逗号分隔。 注意：只有在 “是” Yes 和 VpcEnabledCreateNew Vpc“是” 时才相关No。
ExistingSecurityGroupIds	（可选输入）	用于配置 Lambda 函数的现有 VPC 的安全组列表，以逗号分隔。 注意：只有在 “是” Yes 和 VpcEnabledCreateNew Vpc“是” 时才相关No。

VpcAzs	（可选输入）	以逗号分隔的列表，列出了 AZs 在哪些子网中创建的 VPCs 注意：只有在 “是” Yes 和 VpcEnabledCreateNew Vpc“是” 时才相关No。
UseInferenceProfile	No	如果配置的模型是 Bedrock，则可以指示是否使用基岩推理配置文件。这将确保在堆栈部署期间配置所需的 IAM 策略。有关更多详细信息，请参阅以下 https://docs.aws.amazon.com/bedrock/ latest/userguide/cross-region-inference.html
部署用户界面	是	选择用于部署此部署的前端 UI 的选项。选择 “否”，将仅创建用于托管的基础架构 APIs、身份验证和后端处理。 APIs

6. 选择 Next(下一步)。
7. 在配置堆栈选项页面上，请选择下一步。
8. 在审核页面上，审核并确认设置。选中确认模板将创建 AWS Identity and Access Management (IAM) 资源的复选框。
9. 选择 Create stack（创建堆栈）以部署堆栈。

您可以在 AWS CloudFormation 控制台的 “状态” 列中查看堆栈的状态。您将在大约 10-30 分钟后收到 “创建完成” 状态。

部署独立的 Bedrock Agent 用例

按照本节中的 step-by-step说明配置解决方案并将其部署到您的账户。

部署时间：大约 10-30 分钟

1. 登录 A WS 管理控制台并选择启动 CloudFront 模板的按钮。

BedrockAgent。模板	Launch solution

2. 默认情况下，该模板在美国东部（弗吉尼亚州北部）区域启动。要在其他 AWS 区域启动解决方案，请使用控制台导航栏中的区域选择器。

> **Note**
>
> 此解决方案使用 Amazon Bedrock，但目前并非在所有 AWS 区域都可用。如果您正在使用这些功能，则必须在提供这些服务的 AWS 地区启动此解决方案。有关各地区的最新可用性，请参阅 AWS 区域服务列表。

3. 在创建堆栈页面上，确认 Amazon S3 URL 文本框中已有正确的模板 URL，然后选择下一步。
4. 在指定堆栈详细信息页面上，为您的解决方案堆栈分配一个名称。有关命名字符限制的信息，请参阅 AWS https---docs-aws-amazon-com Identity and Access Man agement 用户指南中的 {https---docs-aws-amazon-comUserGuide--iam-latest--reference-iam-limits-html} [IAM 和 AWS STS 配额]。
5. 在参数下，检查该解决方案模板的参数，并根据需要进行修改。该解决方案使用以下默认值。

参数	默认条目	说明
UseCaseUUID	*<_Requires input_>*	长度 UUIDv4 为 36 个字符，用于标识应用程序中已部署的用例。
UseCaseConfigRecordKey	*<Requires input>*	与包含聊天提供商 Lambda 函数在运行时所需的配置的记录对应的密钥。 表中的记录必须具有与该值匹配的关键属性，以及包含所需配置的配置属性。 如果部署平台正在使用此记录，则该记录将由部署平台

参数	默认条目	说明
		填充。对于此用例的独立部署，需要在中定义的表中手动创建的UseCaseConfigTableName条目。
UseCaseConfigTableName	*<Requires input>`*	堆栈将使用中定义的记录密钥从此处提供的表中读取用例配置UseCaseConfigRecordKey。
DefaultUserEmail	`placeholder@example.com`	此用例的默认用户的电子邮件。该解决方案为该电子邮件创建了一个 Amazon Cognito 用户来访问该用例。

参数	默认条目	说明
ExistingRestApiId	（可选输入）	要使用的现有 API Gateway REST API ID。如果未提供，则将创建一个新的 API Gateway REST API。通常在从 “部署” 仪表板部署时提供。 注意：使用现有 APIs 可以帮助减少资源重复，并简化 APIs 何时需要部署多个独立用例的管理。在 APIs 为独立用例提供现有路径时，您有责任确保 API 配置为具有预期模型的所需路由。必须配置所需的预先配置的 /details 路由（在聊天期间获取用例详细信息），也可以配置 /feedback 路由（如果设置FeedbackEnabledYes为启用收集 LLM 聊天响应的反馈）。此外 ExistingApiRootResourceIdExistingCognitoUserPoolId，还ExistingCognitoGroupPolicyTableName必须提供、。
ExistingApiRootResourceId	（可选输入）	要使用的现有 API Gateway REST API 根资源 ID。REST API 根资源 ID 可以从 AWS 控制台获取，方法是在 API 的 “资源” 部分选择根资源 (/)。然后，资源 ID 将显示在资源详细信息面板中。或者，您也可以在 REST API 上运行描述 API 调用以查找根资源 ID。

参数	默认条目	说明
FeedbackEnabled	No	如果设置为 “否”，则部署的用例堆栈将无法访问反馈功能。
CognitoDomainPrefix	（可选输入）	如果您想为 Amazon Cognito 用户池客户端提供域名，请输入一个值。如果您不提供值，则解决方案会生成一个值。
ExistingCognitoUserPoolId	（可选输入）	UserPoolId您想要验证此用例的现有 Amazon Cognito 用户池。注意：从部署仪表板部署时，通常会提供此 ID，但在独立部署此用例堆栈时可以省略它。
ExistingCognitoUserPoolClient	（可选输入）	提供用户池客户端（应用程序客户端）以使用现有客户端。如果您不提供用户池客户端，则解决方案会创建一个用户池客户端。只有在您提供时，您才能提供此参数ExistingCognitoUserPoolId。
ExistingCognitoGroupPolicyTableName	（可选输入）	包含用户组策略的 DynamoDB 表的名称。这是由自定义授权方在用例的 API 上使用的。注意：从 Deployment 控制面板部署时通常会提供此名称，但是在独立部署此用例堆栈时，可以省略该名称。
VpcEnabled	No	堆栈资源是否部署在 VPC 内。

参数	默认条目	说明
CreateNewVpc	No	Yes如果您希望解决方案为您创建新 VPC 并将其用于此用例，请选择此选项。注意：此参数仅在相关时才VpcEnabled相关Yes。
IPAMPoolId	（可选输入）	如果要使用 IPAM 分配 CIDR 范围，请提供要使用的 IPAM 池 ID。注意：此参数仅在 “是” Yes 和 CreateNew Vpc“VpcEnabled是” 时才相关No。
ExistingVpcId	（可选输入）	用于该用例的现有 VPC 的 VPC ID。注意：此参数仅在 “是” Yes 和 CreateNew Vpc“VpcEnabled是” 时才相关No。
ExistingPrivateSubnetIds	（可选输入）	用于部署 Lambda 函数 IDs 的现有私有子网的子网列表，以逗号分隔。注意：此参数仅在 “是” Yes 和 CreateNew Vpc“VpcEnabled是” 时才相关No。
ExistingSecurityGroupIds	（可选输入）	用于配置 Lambda 函数的现有 VPC 的安全组列表，以逗号分隔。注意：此参数仅在 “是” Yes 和 CreateNew Vpc“VpcEnabled是” 时才相关No。

参数	默认条目	说明
VpcAzs	（可选输入）	以逗号分隔的列表，列出了 AZs 在哪些子网中创建的 VPCs 注意：只有在 “是” Yes 和 VpcEnabledCreateNew Vpc“是” 时才相关No。
BedrockAgentId	*<Requires input>*	要使用的亚马逊 Bedrock Agent 的 ID。
BedrockAgentAliasId	*<Requires input>*	要使用的亚马逊 Bedrock Agent 的别名 ID。
部署用户界面	Yes	选择用于部署此部署的前端聊天用户界面的选项。选择No后会创建用于托管的基础架构 APIs、身份验证以及不使用聊天界面的后端处理。 APIs

6. 选择 Next(下一步)。
7. 在配置堆栈选项页面上，请选择下一步。
8. 在审核页面上，审核并确认设置。选中确认模板将创建 IAM 资源的复选框。
9. 选择 Create stack（创建堆栈）以部署堆栈。

您可以在 AWS CloudFormation 控制台的 “状态” 列中查看堆栈的状态。您将在大约 10-30 分钟后收到 “创建完成” 状态。

提供 DynamoDB 聊天配置

部署用例时，UseCaseConfigRecordKey和UseCaseConfigTableName是必填 CloudFormation 参数，通常由部署仪表板填充。部署仪表板堆栈处理此表的创建和配置，而对部署 API 的调用则会触发参数的填充。

执行独立部署时，必须执行以下操作：

1. 使用哈希键为密钥创建一个 DynamoDB 表。
2. 在包含用例配置的表中创建一条记录作为格式记录：`{key: some_use_case_key, config: {your_configuration}.`
3. 部署时，将选定的UseCaseConfigTableName和 UseCaseConfigRecordKey（在本示例`some_use_case_key`中）参数传递给用例堆栈。

要为独立部署创建合适的配置，您可以从 Deployment 控制面板创建所需的用例，然后从配置表中复制记录。否则，您可以根据以下 Bedrock 部署示例制作自己的配置：

```
{
 "UseCaseName": "SampleUseCase",
 "ConversationMemoryParams": {
 "ConversationMemoryType": "DynamoDB",
 "HumanPrefix": "H",
 "AiPrefix": "A",
 "ChatHistoryLength": 20
 },
 "KnowledgeBaseParams": {
 "KnowledgeBaseType": "Bedrock",
 "NumberOfDocs": 2,
 "ScoreThreshold": 0,
 "ReturnSourceDocs": false,
 "BedrockKnowledgeBaseParams": {
 "BedrockKnowledgeBaseId": "SOME_ID",
 "OverrideSearchType": null
 }
 },
 "LlmParams": {
 "ModelProvider": "Bedrock",
 "BedrockLlmParams": { "ModelId": "anthropic.claude-v2" },
 "PromptParams": {
 "PromptTemplate": "some prompt",
 "MaxPromptTemplateLength": 187500,
 "MaxInputTextLength": 187500,
 "UserPromptEditingEnabled": true,
 "DisambiguationEnabled": true,
 "DisambiguationPromptTemplate": "some prompt"
 },
 "ModelParams": {},
 "Temperature": 1,
 "RAGEnabled": true,
```

```
 "Streaming": true,
 "Verbose": false
 }
}
```

使用 Service Catalog 监控解决方案 AppRegistry

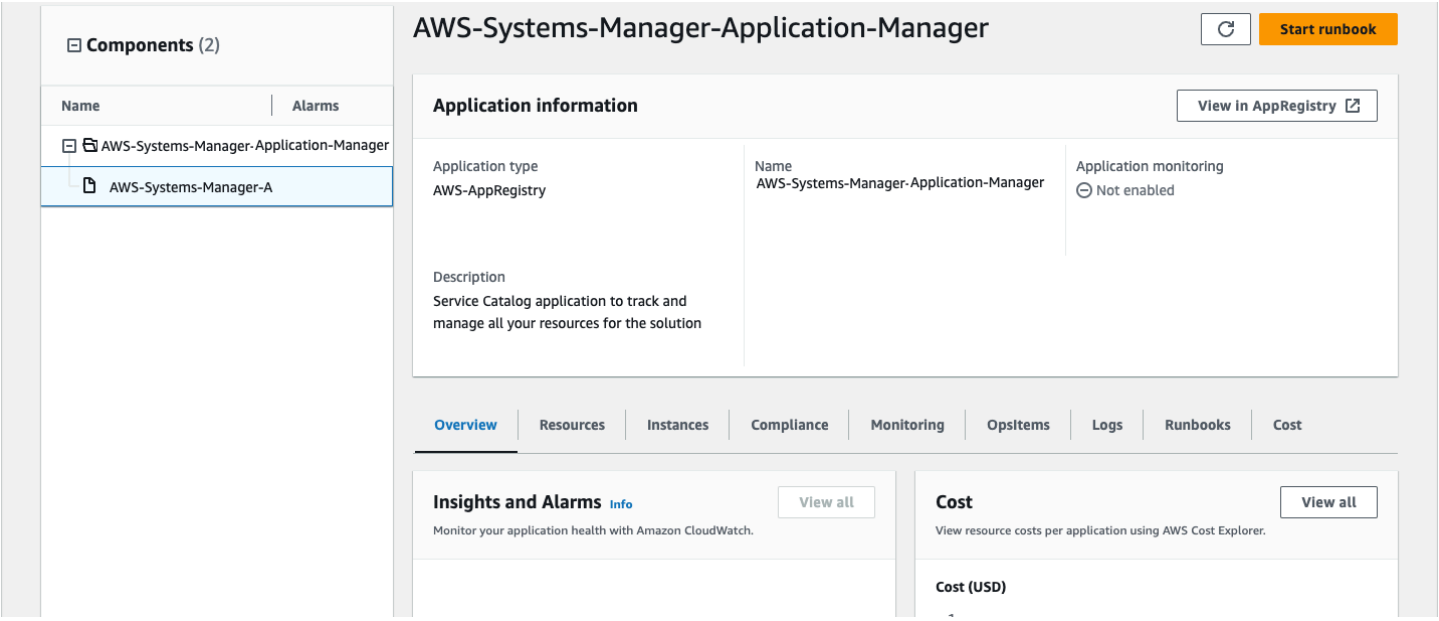
该解决方案包括服务目录 AppRegistry 资源，用于在 Service Catalog 和 Systems Manager Application Manager Application Manager 中将 CloudFormation 模板 AppRegistry 和底层资源注册为应用程序。

Systems Manager Application Manager 为您提供了此解决方案及其资源的应用程序级视图，因此您可以：

- 从中心位置监控其资源、跨堆栈和 AWS 账户部署的资源的成本，以及与此解决方案相关的日志。
- 在应用程序的上下文中查看此解决方案资源的操作数据。例如，部署状态、 CloudWatch 警报、资源配置和操作问题。

下图描述了 Application Manager 中解决方案堆栈的应用程序视图示例。

描绘应用程序管理器中的解决方案堆栈



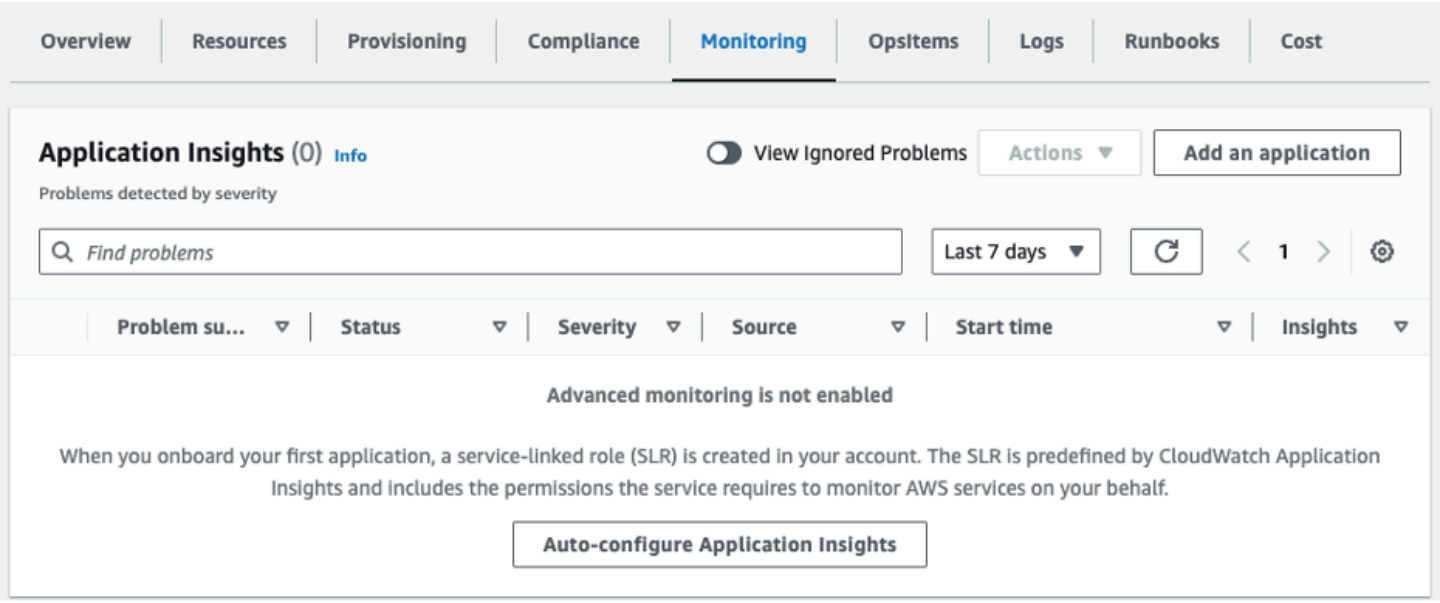
激活 CloudWatch 应用程序见解

1. 登录 Systems Manager 控制台。
2. 在导航窗格中，选择 Application Manager。
3. 在 “应用程序” 中，搜索此解决方案的应用程序名称并将其选中。

应用程序名称的 “应用程序来源” 列中将包含 App Re gistry，并将包含解决方案名称、区域、账户 ID 或堆栈名称的组合。

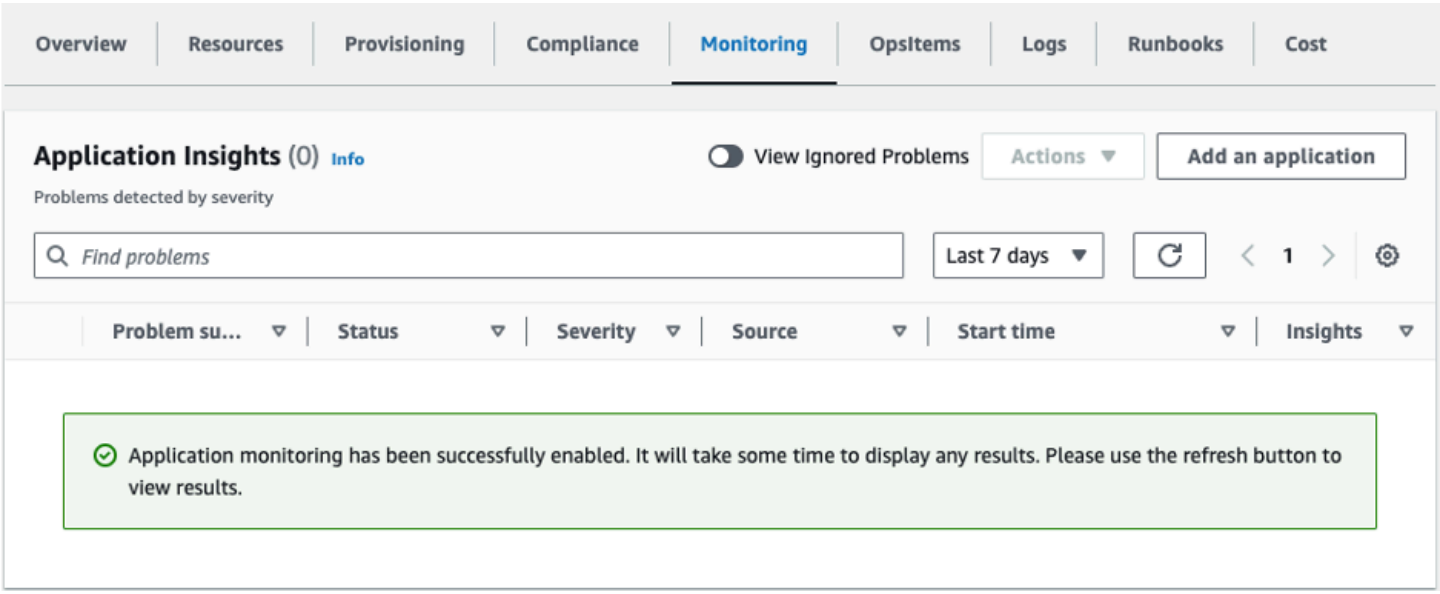
4. 在组件树中，选择要激活的应用程序堆栈。
5. 在 “监控” 选项卡的 “应用程序见解” 中，选择 “自动配置应用程序见解”。

Application Insights 仪表板显示未检测到的问题和自动配置选项。



监控应用程序现已激活，系统显示以下状态框：

Application Insights 仪表板显示成功监控



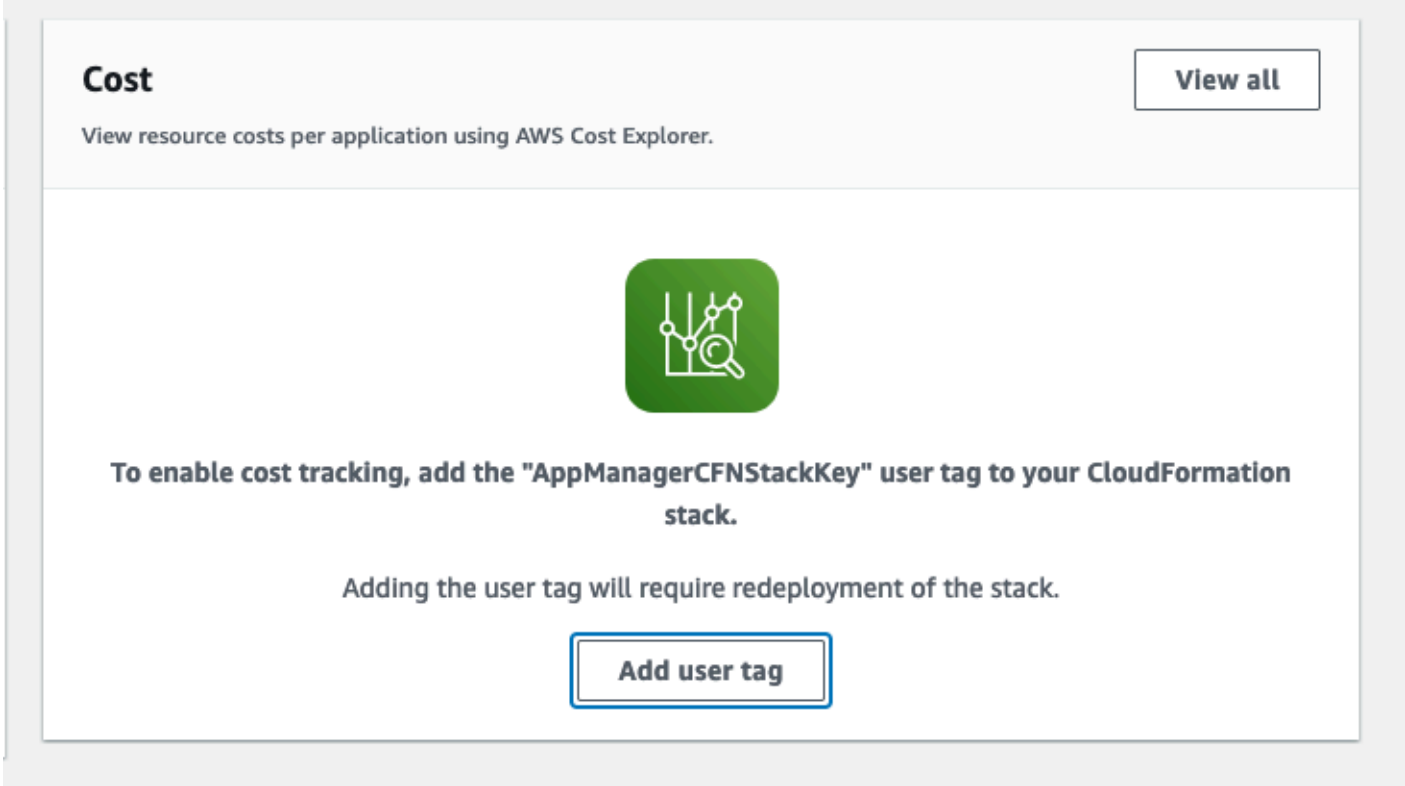
确认与此解决方案关联的成本标签

激活与此解决方案关联的成本分配标签后，您必须确认成本分配标签才能查看此解决方案的成本。要确认成本分配标签，请按以下步骤操作：

1. 登录 Systems Manager 控制台。
2. 在导航窗格中，选择 Application Manager。
3. 在应用程序中，选择此解决方案的应用程序名称并将其选中。

   应用程序名称的 “应用程序来源” 列中将包含 App Re gistry，并将包含解决方案名称、区域、账户 ID 或堆栈名称的组合。
4. 在概览选项卡的成本中，选择添加用户标签。

   屏幕截图描绘了应用程序成本添加用户标签屏幕



5. 在添加用户标签页面上，输入 `confirm`，然后选择添加用户标签。

激活过程可能需要长达 24 小时才能完成，显示标签数据。

激活与此解决方案关联的成本分配标签

激活 Cost Explorer 成本管理服务后，您必须激活与此解决方案关联的成本分配标签才能查看此解决方案的成本。成本分配标签只能在组织的管理账户中激活。要激活成本分配标签，请按以下步骤操作：

1. 登录 AWS Billing and Cost Management 以及成本管理控制台。
2. 在导航窗格中，选择成本分配标签。
3. 在成本分配标签页面上，筛选 AppManagerCFNStack密钥标签，然后从显示的结果中选择标签。
4. 选择激活。

AWS Cost Explorer 成本管理服务

通过与必须先激活的 AWS Cost Explorer 集成，您可以在应用程序管理器控制台中查看与应用程序和应用程序组件相关的成本概览。Cost Explorer 通过提供一段时间内您的 AWS 资源成本和使用情况的视图，帮助您管理成本。要为此解决方案激活 Cost Explorer 成本管理服务，请按以下步骤操作：

1. 登录 A WS 成本管理控制台。
2. 在导航窗格中，选择 Cos t Explorer 以查看解决方案在一段时间内的成本和使用情况。

更新此解决方案

如果您之前部署过该解决方案，请按照以下步骤更新解决方案 CloudFormation 堆栈以获取最新的功能和增强功能。升级过程分为三个部分：

- 步骤 1：更新部署控制面板
- 步骤 2：迁移用例配置
- 第 3 步：更新用例

> **Note**
>
> 1. 在 v2.0.0 中，不推荐与 Anthropic 和 Hugging Face 集成，转而使用亚马逊 Bedrock 和亚马逊 AI。 SageMaker 你可以通过以下方式部署 Hugging Fac SageMaker JumpStart e 提供的模型。有关更多详细信息，请参阅将 Hugging Face 与 A SageMaker mazon AI 配合使用。
> 2. 在运行这些步骤之前，请确保在非生产环境中测试更新过程。

步骤 1：更新部署控制面板

1. 登录CloudFormation 控制台，选择您的现有 CloudFormation 堆栈，然后选择更新。
2. 选择替换当前模板。
3. 在指定模板下：
   a. 选择 Amazon S3 URL。
   b. 复制最新的CloudFormation 模板链接。
   c. 将链接粘贴到 Amazon S3 URL 框中。
   d. 验证 Amazon S3 URL 文本框中显示了正确的模板 URL，然后选择下一步。再次选择下一步。
4. 在参数下，检查模板的参数，并根据需要进行修改。有关参数的详细信息，请参阅步骤 1：启动部署控制面板堆栈。
5. 选择 Next(下一步)。
6. 在配置堆栈选项页面上，请选择下一步。
7. 在审核页面上，审核并确认设置。选中用于确认模板将创建 IAM 资源的框。
8. 选择查看更改集并验证更改。

9. 选择更新堆栈以部署堆栈。

您可以在 AWS CloudFormation 控制台的 “状态” 列中查看堆栈的状态。您将在大约 10 分钟后收到 “更新完成” 状态。

如果现有解决方案版本低于 v2.0.0，则更新将创建一个 Web UI 堆栈（使用 Cognito 托管用户界面取代登录屏幕的`amplify-ui`实现）和一个新 CloudFront URL，堆栈状态为 UPDATE_COMPLETE 后，即可从 CloudFormation 控制台的 “输出” 部分获取该堆栈。

> **Note**
>
> 在您完成以下步骤之前，不会显示使用 v2.0.0 之前的版本创建的现有用例。

步骤 2：迁移用例配置（仅限 2.0.0 以下版本的更新）

在 2.0.0 版本中，存储架构和用于存储用例的 AWS 服务配置已更改。使用 gaab_v2_migration.py 脚本按照 GAAB v2 迁移用户指南中描述的步骤进行操作。运行脚本后，您可以访问部署仪表板以查看已部署的用例。

> **Note**
>
> 您必须按照以下步骤完成用例的迁移。

第 3 步：更新用例

您可以使用最新版本的 GAAB 中提供的新功能来编辑已部署的用例。有关如何使用此解决方案中的功能的信息，请参阅使用解决方案。

要将用例更新到最新版本，您必须完成部署控制面板中的 “编辑” 用例步骤（尽管您可能不会进行任何更改）。此操作会触发使用最新模板版本的 CloudFormation 堆栈更新。

> **Note**
>
> 使用 1.x 或 2.x 版本的解决方案创建的用例可能不适用于更高版本。因此，我们建议通过 Deployment 控制面板克隆使用 v3.0.0 之前的版本创建的现有用例。然后，逐步迁移并替换为使用 v3.0.0 或更高版本创建的新用例。

问题排查

此部分提供有关部署和使用解决方案的问题排查说明。

如果这些说明不能解决您的问题，联系支持会提供有关如何为此解决方案开立支持工单的说明。

问题：使用 “为我创建 VPC” 部署支持 VPC 的配置失败

部署控制面板堆栈或用例堆栈部署失败， CloudFormation 因为无法配置 VPC 网络资源。

解决方案

查看您账户中的 VPCs、和 Elastic IPs 的配额限制。Elastic IPs 和每个 AWS 账户 VPCs 每个 AWS 区域的默认限制各为 5 个。

> **Note**
>
> 当解决方案创建 VPC 时，单个启用 VPC 的部署（部署控制面板或用例）是 2 可用区部署，每个可用区中有 1 个公有子网和 1 个私有子网，每个公有子网部署 1 个 NAT 网关。如果有 2 个 NAT 网关，则部署会消耗配额限制中的 2 个公有 IP 地址。

需要注意的一些限制（每个账户、每个区域）：

- 数量 VPCs -5
- 公有 IP 地址的数量-5
- 网关 VPC 终端节点数量-20
- 接口 VPC 终端节点数量-20

问题：删除部署仪表板堆栈 CloudFormation 后，无法在中删除用例堆栈

如果在删除所有用例堆栈 CloudFormation 之前删除了 Deployment 仪表板堆栈，则用例可能最终处于锁定（不可用）状态。这是因为 Deployment 控制面板堆栈创建的 IAM 角色已不存在，无法修改用例堆栈。

解决方案

> **Warning**
>
> 确保在使用后立即清理所有手动创建的角色。这些是提升的权限，用户可以利用这些权限进行角色提升。

重新创建已删除的 IAM 角色以允许删除 CloudFormation 堆栈：

1. 打开 CloudFormation 控制台并确定与您的锁定堆栈关联的角色。
   a. 角色 ARN 可以在标有 IAM 角色的堆栈信息部分中找到。
   b. 角色名称是 IAM 角色 ARN 中:role/ 之后的名称（例如，arn: aws: iam::: role/ ）<account-id><role-name>
2. 在 IAM 中创建一个与已删除角色同名的新角色。
   a. 选择 AWS 服务作为可信实体，然后CloudFormation从下拉列表中选择。
   b. 添加必要的权限。如果您不确定所需的权限，可以使用 AWS 托管AdministratorAccess策略。
   c. 输入与步骤 1 中获得的完全相同的角色名称。
3. 返回 CloudFormation 控制台并删除锁定的堆栈。
4. 成功删除所有锁定的堆栈后，返回 IAM 并删除步骤 2 中创建的所有角色。

问题：用例用户界面无法反映设置中的更改

更新用例后，用户界面将部署到 CloudFront。但是，由于 CloudFront 缓存部署以及指示如何向用户显示某些设置的配置文件，因此这些更改可能不会立即反映出来。

解决方案

可以使 CloudFront 分发失效，以强制将新配置传播给前端用户。

1. 打开 CloudFormation 控制台并确定与您的用例堆栈关联的 CloudFront 发行版。
   a. 用例堆栈的开头应与部署用例时使用的名称相同。
   b. 找到与 UI 对应的嵌套堆栈。嵌套堆栈名称应以 WebAppS3 UINested Stack UINested StackResource S3 开头。

c. 在资源选项卡下，找到资源类型 AWS::CloudFront::Distribution，然后选择物理 ID。这将在 CloudFront 控制台中打开发行版。

2. 导航至 “失效” 选项卡，然后选择 “创建失效”，然后输入路径 /*。这将使所有路径失效。
3. 在您自己的浏览器中，删除与用例相关的所有 Cookie 和缓存文件。

联系 AWS Support

如果您有 AWS Business Support+、AWS Enterpris e Support 或统一运营，则可以使用 AWS 支持中心获取有关此解决方案的专家帮助。以下部分提供了说明。

创建工单

1. 登录支持中心。
2. 选择创建工单。

我们可提供哪些帮助？

1. 选择技术。
2. 对于服务，选择解决方案。
3. 在 “类别” 中，选择 “其他解决方案”。
4. 对于严重性，选择与您的使用案例最匹配的选项。
5. 当您填写完服务、类别和严重性信息后，界面中会填入常见故障排除问题的链接。如果您无法通过这些链接解决问题，请选择下一步: 附加信息。

附加信息

1. 对于主题，输入可概括您的问题的文本。
2. 有关描述，请详细描述问题，包括此解决方案的名称：AWS 上的生成式 AI 应用程序生成器。
3. 选择附加文件。
4. 附上 AWS Support 处理请求所需的信息。

帮助我们更快地处理您的工单

1. 输入请求的信息。
2. 选择下一步：立即解决或联系我们。

立即解决或联系我们

1. 查看立即解决中的解决方案。
2. 如果您无法使用这些解决方案解决问题，请选择联系我们，输入请求的信息，然后选择提交。

卸载此解决方案

> **Note**
>
> 通过部署仪表板创建的部署不打算在解决方案之外进行管理。在中删除堆栈之前，请务必从部署仪表板中删除并清理所有部署 CloudFormation。

您可以从 AWS 管理控制台或使用 AWS 命令行界面卸载 AWS 上的生成式 AI 应用程序生成器解决方案。您必须手动删除此解决方案创建的 Amazon S3 存储桶、Amazon Kendra 索引 CloudWatch 或日志。如果您存储了需要保留的数据，AWS 解决方案不会自动删除 Amazon S3 存储桶、Amazon Kendra 索引 CloudWatch 或日志。

使用 AWS 管理控制台

1. 登录 A WS CloudFormation 控制台。
2. 在堆栈页面上，选择此解决方案的安装堆栈。
3. 选择删除。

使用 AWS 命令行界面

确定 AWS 命令行界面（AWS CLI）在您的环境中是否可用。有关安装说明，请参阅 AWS CLI 用户指南中的 AWS 命令行界面是什么。确认 AWS CLI 可用后，请运行以下命令。

```
$ aws cloudformation delete-stack --stack-name <installation-stack-name>
```

手动卸载步骤

删除 Amazon S3 存储桶

如果您决定删除 AWS CloudFormation 堆栈以防止意外丢失数据，则此解决方案配置为保留解决方案创建的 Amazon S3 存储桶。卸载解决方案后，如果您不需要保留数据，则可以手动删除此 Amazon S3 存储桶。按照以下步骤删除 Amazon S3 存储桶。

1. 登录 Amazon S3 控制台。

2. 在导航窗格中，选择 Buckets。
3. 找到 S <stack-name>3 存储桶。
4. 选择 S3 存储桶，然后选择删除。

要使用 AWS CLI 删除 S3 存储桶，请运行以下命令。使用--force 选项时，无需先清空存储桶。

```
$ aws s3 rb s3://<bucket-name> --force
```

删除亚马逊 Kendra 索引

为防止数据意外丢失，此解决方案配置为在删除 AW CloudFormation S 堆栈时保留解决方案创建的 Amazon Kendra 索引。卸载解决方案后，您可以手动删除不再需要为其保留数据的 Amazon Kendra 索引。按照以下步骤删除亚马逊 Kendra 索引。

1. 登录亚马逊 Kendra 控制台。
2. 在导航窗格中，选择索引。
3. 找到并选择要删除的索引。
4. 选择删除，以便删除所选索引。

要使用 AWS CLI 删除 Amazon Kendra 索引，请运行以下命令：

```
$ aws kendra delete-index --id<index-id>
```

删除日 CloudWatch 志

为防止意外丢失数据，我们将此解决方案配置为在您决定删除 CloudFormation 堆栈时保留 CloudWatch 日志。卸载解决方案后，如果您不需要保留数据，则可以手动删除日志。按照以下步骤删除日 CloudWatch 志。

1. 登录 Amazon CloudWatch 控制台。
2. 在导航窗格中，选择日志组。
3. 找到解决方案创建的日志组。
4. 选择其中一个日志组。
5. 选择操作，然后选择删除。

重复这些步骤，直到删除所有解决方案日志组。

使用解决方案

访问用户界面

在堆栈部署过程中（对于部署仪表板和用例），将向配置的电子邮件地址发送一封电子邮件。该电子邮件包含用户的临时证书，他们可以用来注册和访问 Web 界面。

> **Note**
>
> 堆栈完成后，有权访问 AWS 管理控制台的用户必须向管理员用户提供部署控制面板用户界面的 CloudFront URL。 DevOps

对于用例，部署完成后，有权访问部署仪表板用户界面的管理员用户必须向业务用户提供用例用户界面的 CloudFront URL。

登录后，用户可以与解决方案进行交互 UIs，管理员可以通过 Deployment 控制面板进行交互，对于业务用户，则可以与用例进行交互。

如何更新部署

在部署仪表板主页（或部署的详细信息页面）上，您可以编辑部署使用的配置。您只能编辑处于 CREATE_COMPLETE 或 UPDATE_COMPLETE 状态的部署。

除用例名称外，所有其他选项均可在部署中编辑。只需更改要编辑的值并重新部署即可。

根据所做的编辑范围，重新部署的时间会有所不同。如果简单设置（例如，模型参数）已更改，则可能需要几秒钟；如果与基础设施相关的较大选项已更改（例如，请求为文本用例 RAG 创建 Amazon Kendra 索引），则可能需要超过 30 分钟。

成功完成编辑后，应用程序状态将报告为 UPDATE_COMPLETE 状态。此时，您可以通过 CloudFront URL 访问已部署的用户界面并与修改后的部署进行交互。

> **Note**
>
> 如果您想比较不同的设置或， side-by-side则运行多个部署可能会更容易 LLMs。使用克隆功能快速使用现有配置启动新部署。

如何克隆部署

在部署仪表板主页（或部署的详细信息页面）上，您可以克隆部署使用的配置。克隆部署会启动 “部署新用例” 向导，但大多数字段都预先填充了相同的值。

这是一项便捷操作，可帮助您快速复制已更改设置的部署、恢复已删除的部署，或者比较原本相同的部署 LLMs 中的多个部署。

如何删除部署

在部署仪表板主页（或部署的详细信息页面）上，可以在不再需要部署后将其删除。删除部署会调用 CloudFormation 堆栈删除操作并取消部署资源。

默认情况下，已删除的部署仍保留在仪表板上，以启用克隆功能。要将部署从仪表板中完全移除，以便停止在用户界面中对其进行跟踪，请在删除确认窗口中选择永久删除。

> **Important**
>
> 删除堆栈时会留下一些资源，必须手动删除。有关保留哪些资源以及如何清理这些资源的详细信息，请参阅手动卸载部分。

配置大型语言模型 (LLM)

哪种 LLM 适合您的用例取决于您的需求和您想要策划的客户体验类型所特有的大量因素。该解决方案看起来不是规范性的，而是旨在为您提供必要的工具，以评估哪种方法最适合您的应用程序。

人工智能生成的领域正在迅速发展，因此您有责任及时了解最新的模型、优化技术和最佳实践，以确保为客户打造合适的体验。

> **Note**
>
> 如果您正在处理非公开或敏感数据，请务必使用 AWS 服务（例如 Amazon Bedrock 或 Amazon AI）选择 LLM 选项。 SageMaker 与使用第三方提供商托管的 LLM 相比，这样可以将数据保留在您所在的地区和 AWS 网络上，从而改善部署的整体安全状况。

使用 Amazon SageMaker AI 作为 LLM 提供商

从 v1.3.0 开始，A mazon SageMaker AI 可用作文本用例的模型提供者。此功能允许您在解决方案中使用 AWS 账户中已存在的 A SageMaker I 推理终端节点。以下是一些入门方法。

> **Important**
>
> 该解决方案不管理您的 SageMaker AI 终端节点的生命周期。当不再需要 SageMaker AI 终端节点时，您有责任将其删除，以免产生额外费用。

创建 A SageMaker I 终端节点

您可以使用 Amazon SageMaker AI JumpStart 快速部署终端节点。

您还可以使用基于文本生成的 SageMaker AI 端点，并使用基 SageMaker 本 AI 服务进行部署。有关如何部署模型进行推理的分步指南，请参阅 SageMaker AI JumpStart 文档。

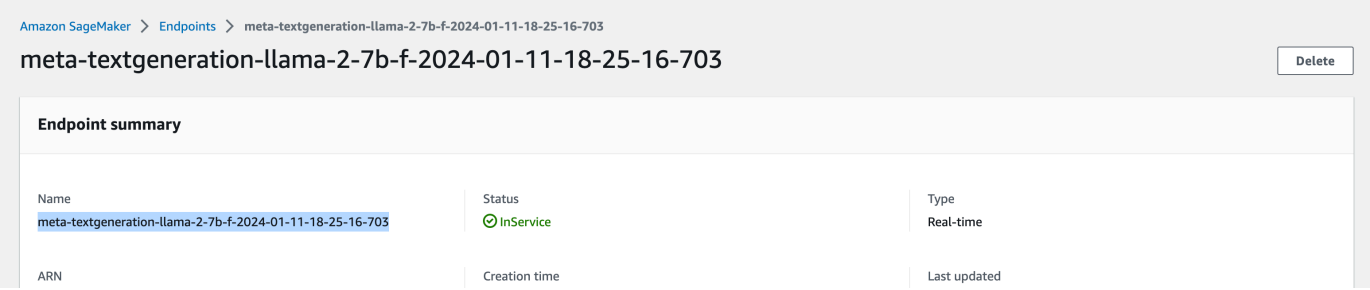
> **Note**
>
> 基础 models/LLMs 通常相当大，通常需要使用大型加速计算实例。默认情况下，其中许多较大的实例在您的 AWS 账户中可能不可用。请参阅默认 SageMaker AI 配额，并确保在部署前申请增加配额，以避免常见的部署失败。

使用 SageMaker AI 端点创建文本用例部署

要使用 SageMaker AI 端点进行推理部署新的文本用例，请执行以下操作：

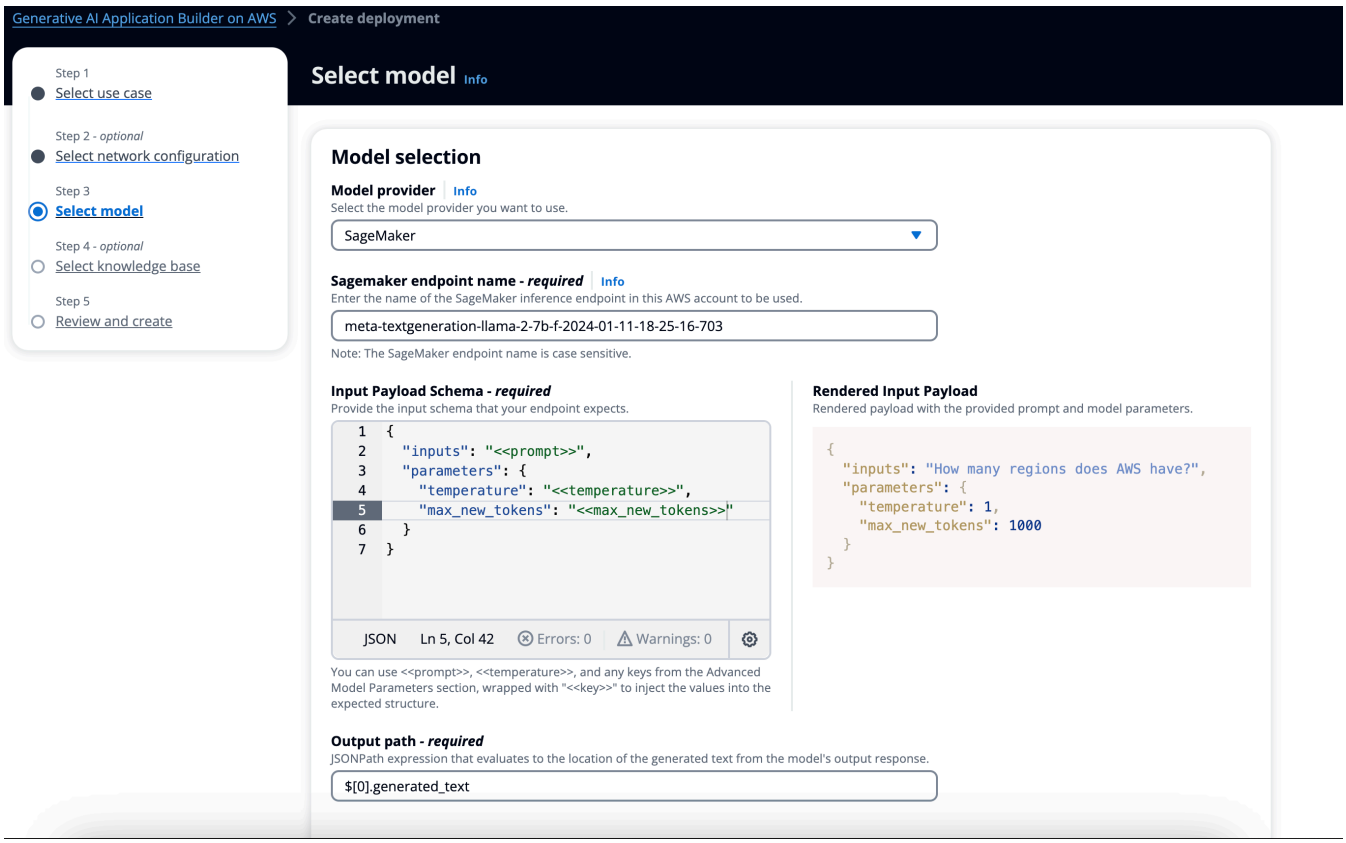
1. 通过部署仪表板向导@@ 创建新的用例并填写表单，直到进入模型选择页面。
2. 在模型页面上，选择 SageMaker AI 作为模型提供者。这将生成一个需要用户输入三个关键部分的自定义表单：
   - 您要使用的 SageMaker AI 终端节点的名称。 DevOps 用户可以从 AWS 控制台获取此信息。请注意，终端节点必须与解决方案部署在同一个账户和区域中。

     终端节点名称在 AWS 控制台上的位置



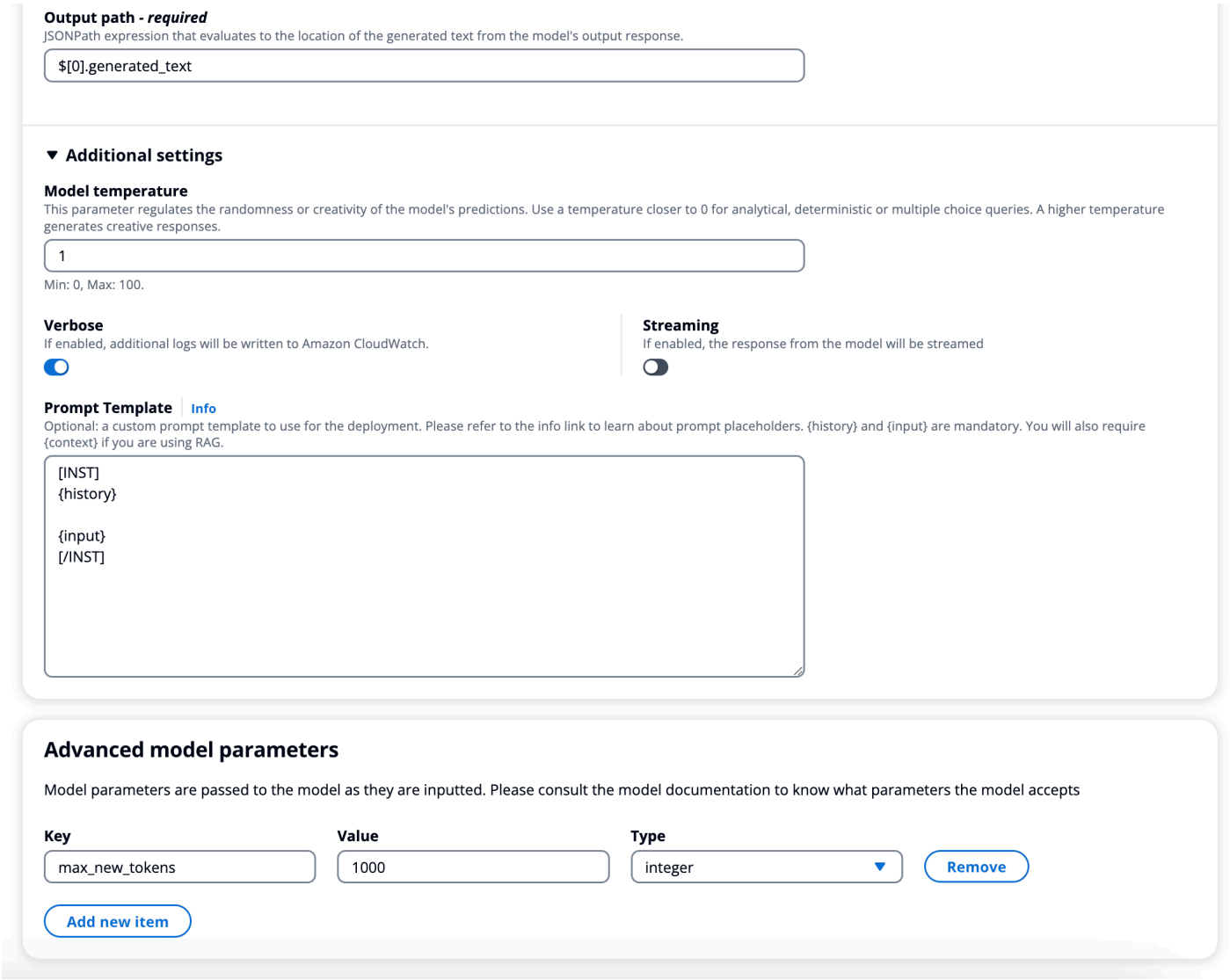
- 端点期望的输入有效载荷架构。为了支持最广泛的端点，管理员用户需要告诉解决方案他们的端点期望如何格式化输入。在模型选择向导中，提供要发送到端点的解决方案的 JSON 架构。您可以添加占位符以将静态和动态值注入请求有效负载。可用选项如下：
  - 强制占位符：\ <\ <prompt\ >\ > 将被动态替换为要在运行时发送到 SageMaker AI 端点的完整输入（例如，历史记录、上下文和提示模板中的用户输入）。
  - 可选占位符：<temperature\ >\ <\\ > *、\ * 以及高级模型参数中定义的任何参数都可以提供给端点。任何包含以\ <\ < and\ >\ > 括起的占位符的字符串（例如 <max_new_tokens\ >\ <\\ > ）都将被同名的高级模型参数的值替换。

    输入架构示例-设置必填字段、提示和温度，以及自定义高级参数 max_new_tokens。输出路径必须作为有效 JSONPath 字符串提供



3. LLMs 生成的字符串响应在输出负载中的位置。必须将其作为 JSONPath 表达式提供，以指示应从端点的返回对象和响应中访问显示给用户的最终文本响应的位置。

   添加要在 SageMaker AI 输入架构中使用的高级模型参数的示例（有关之前的选项/设置，请参阅图 2）



> **Note**
>
> SageMaker AI 现在支持在同一个端点后面托管多个模型，这是在当前版本的 SageMaker AI Studio（不是 Studio Classic）中部署端点时的默认配置。
> 如果您的终端节点是以这种方式配置的，则需要InferenceComponentName向高级模型参数部分添加一个与您要使用的模型名称相对应的值。

高级法学硕士设置

在使用 Amazon Bedrock 时，您可以为模型配置一些高级设置，例如 Amazon Bedrock Guardrails、Amazon Bedrock 的预配置吞吐量以及其他模型参数。

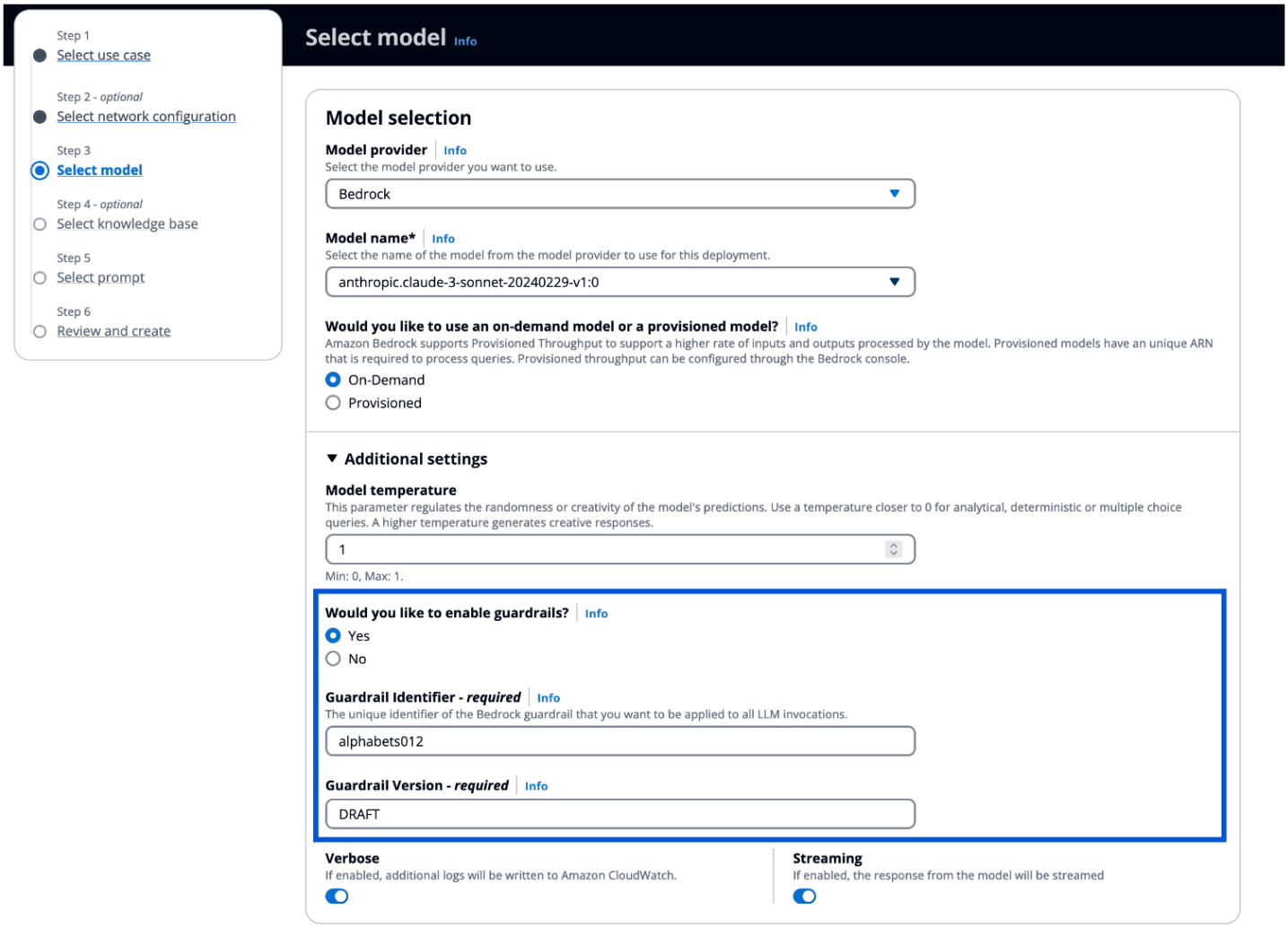
Amazon Bedrock 护栏

Amazon Bedrock Guardrails 是 Amazon Bedrock 的一项功能，它根据用户配置的策略评估用户输入和 LLM 响应，并提供额外的保护措施，无论用户为用例选择哪种底层 LLM。Guardrail 由 2 项策略组成，用于避免内容属于不良或有害类别：

1. 拒绝的主题，用于定义一组在用户应用程序环境中不受欢迎的话题，例如，金融应用程序中的投资建议，以及
2. 内容过滤器****它允许过滤包含有害内容的输入用户提示或模型响应。

要在生成式 AI 应用程序生成器解决方案中使用，必须使用创建护栏向导在 A mazon Bedrock 控制台中配置护栏。创建后，您可以通过提供您的护栏标识符和护栏版本，将此 Guardrail 添加到通过生成式 AI App Builder 解决方案向导在 “模型选择” 步骤中的其他设置中创建的聊天用例中。

描述部署向导——启用 Amazon Bedrock Guardrails

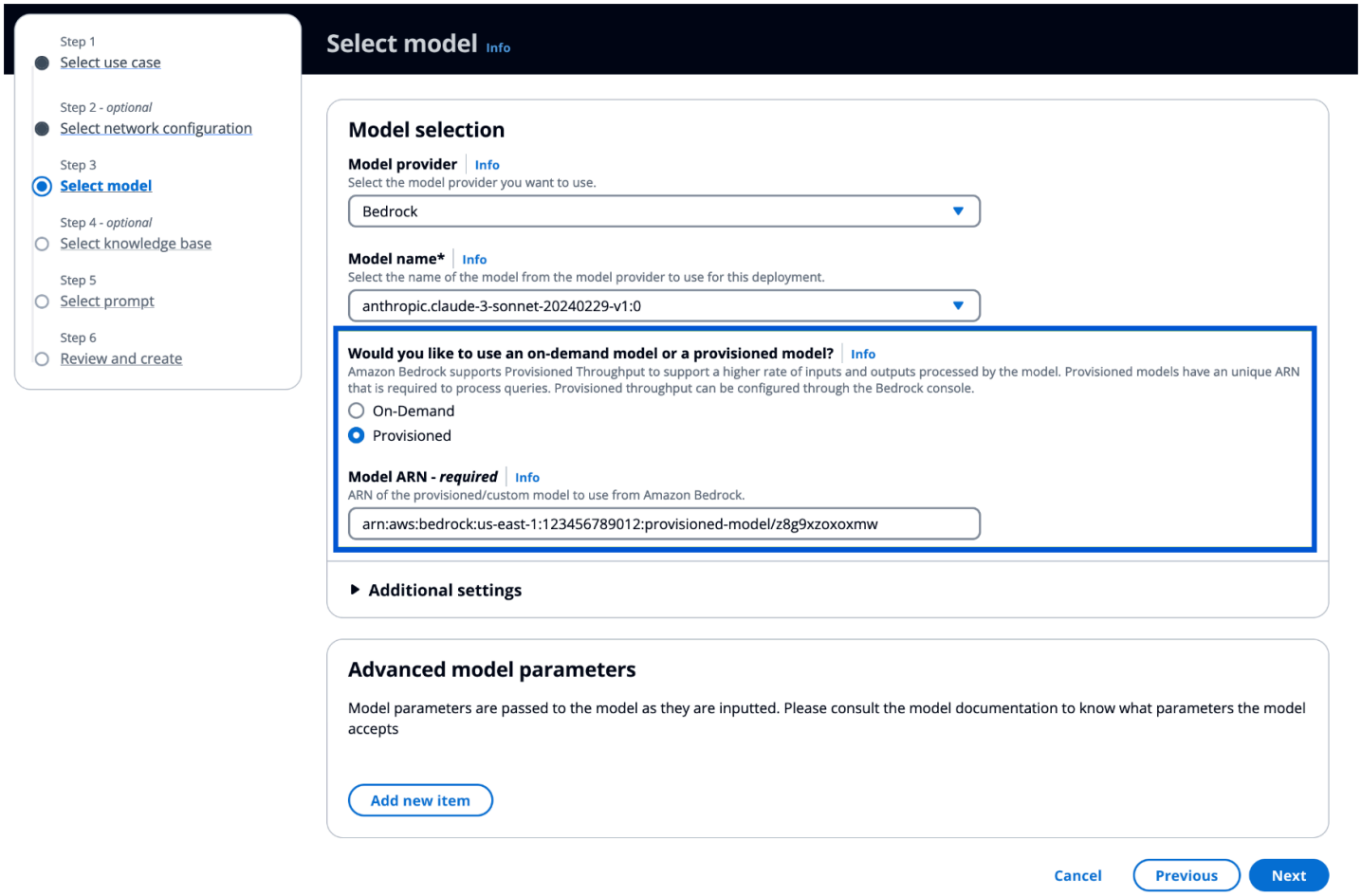


Amazon Bedrock 的预配置吞吐量

每个按需 Amazon Bedrock 模型都遵循特定区域的账户配额限制，用于模型推理。例如，Bedrock 上的 Anthropic Claude 2.x 目前允许在 us-east-1 和 us-west-2 地区每分钟处理 500 个请求和 500,000 个令牌。您可能还想将该解决方案与经过微调或持续的预训练模型一起使用。对于此类实例，Amazon Bedrock 允许预配置吞吐量，从而允许为您的基础运行大型一致的推理工作负载，并允许在生产级应用程序中使用经过微调或持续的预训练模型。

在 Amazon Bedrock 控制台中购买预配置吞吐量后，系统会生成一个模型 ARN 供使用。现在，您可以在生成式 AI 应用程序开发器向导的模型选择步骤中提供此模型 ARN。为此，请选择 Bedrock 作为模型提供者，并选择用于在 Amazon Bedrock 控制台中生成此预配置模型 ARN 的基本模型名称。然后，在按需模型和预配置模型之间进行选择时选择 “预配置模型”，并提供您的模型 ARN。

描述部署向导-为 Amazon Bedrock 启用预配置吞吐量



> **Note**
>
> 您的防护栏和预配置吞吐量必须与部署的部署控制面板和用例堆栈位于同一区域。

模型参数

LLMs 通常接受与其实现相关的各种参数。模型提供者通常会提供文档，概述支持的参数集及其用途。

该解决方案将模型参数直接传递给基础模型，因此确保参数设置正确非常重要。有关支持参数的最新信息，请参阅模型提供商的文档。

配置代理生成器

Agent Builder 为创建可用于生产的 AI 代理提供了全面的配置选项。本节介绍如何配置和管理 Agent Builder 部署。

系统提示符配置

系统提示符定义了您的代理的行为、个性和能力。要配置系统提示符，请执行以下操作：

1. 在代理生成器向导中，导航到配置代理步骤。
2. 在文本编辑器中编辑系统提示模板。
3. 包括以下内容的明确说明：
   - 代理人的角色和目的
   - 如何使用可用工具（MCP 服务器）
   - 响应格式首选项
   - 行为指南
4. 如果需要，使用重置为默认值按钮恢复原始模板。

代理提示的最佳实践：

- 具体说明代理的能力和局限性
- 提供所需行为的清晰示例
- 包括工具使用说明以及何时调用这些工具
- 定义回复格式期望
- 为代理行为设定界限

MCP 服务器集成

模型上下文协议 (MCP) 服务器为代理提供对企业工具和数据源的访问权限。要配置 MCP 服务器，请执行以下操作：

1. 在 “配置代理” 步骤中，找到 “MCP 服务器” 部分。
2. 在下拉菜单中从可用的 MCP 服务器中进行选择。

> **Note**
>
> 在部署代理之前，必须对 MCP 服务器进行配置和访问。代理将自动发现和使用已配置的 MCP 服务器公开的工具。有关服务器设置和工具配置，请参阅 MCP 文档。

内存设置

Agent Builder 为维护上下文和知识提供了两种类型的内存：

短期记忆

默认情况下，所有代理都处于启用状态：

- 维护会话中的对话背景
- 自动捕获用户消息和代理回复
- 由 actorID 和 sessionID 整理以实现适当的隔离
- 无需配置

长期记忆

用于跨会话存储见解的可选功能：

1. 在 “配置代理” 步骤中，找到 “内存配置” 部分。
2. 切换启用长期记忆以激活。
3. 启用后，代理可以：
   - 提取和存储对话中的重要信息
   - 从之前的会话中检索相关上下文
   - 积累有关用户偏好和历史记录的知识

> **Note**
>
> 长期记 AgentCore 忆使用具有语义记忆策略和默认保留设置的内存。

监视代理生成器部署

Agent Builder 通过 CloudWatch 仪表板和指标提供全面监控。

访问 CloudWatch 仪表板

1. 导航到您的 AWS 账户中的 CloudWatch 控制台。

2. 从左侧导航栏中选择 “仪表板”。
3. 找到名为的仪表板AgentBuilder-<UseCaseId>。
4. 查看实时指标和历史性能数据。

日志访问和分析

代理日志可在 CloudWatch 日志中找到：

1. 在 AWS 控制台中导航到 “ CloudWatch 日志”。
2. 查找前缀为的日志组。/aws/bedrock-agentcore/runtimes/
3. 使用 CloudWatch Insights 查询和分析日志。
4. 搜索特定的请求 IDs 或错误模式。

配置工作流生成器

Workflow Builder 通过将工作委托给专门的 Agent Builder 代理的主管代理来实现多代理编排。

创建工作流

1. 导航到部署控制面板
2. 选择 “创建工作流程用例”
3. 配置主管代理：
   - 名称：工作流程的描述性名称
   - 描述：目的和能力
   - 系统提示：代理委派和协调说明
   - 模型：主管代理的基础模型

主管提示的最佳实践：

- 清楚描述何时使用每个专业代理
- 包括汇总来自多个代理的结果的说明
- 定义回复格式预期
- 为委托行为设定界限

代理选择

选择要包含为专业代理的 Agent Builder 代理：

1. 在工作流配置中单击 “添加代理”
2. 浏览或搜索可用的代理生成器代理
3. 查看代理描述
4. 选择要包含在工作流程中的代理

代理描述

主管代理使用代理描述来决定委托给哪个代理。确保描述清楚地解释：

- 代理的专业领域或能力
- 代理处理的任务类型
- 投入/产出预期

测试工作流程

部署后：

1. 通过部署仪表板访问工作流程
2. 使用需要多个代理的查询进行测试
3. 在 CloudWatch 日志中监控代理委派
4. 审查回复质量和授权模式
5. 如果委派不理想，请调整主管的提示

管理模型代币限制的提示

注意：该解决方案不会直接尝试管理各种人施加的代币限制 LLMs。测试并确保您的提示保持在模型提供者强制执行的可用限制范围内。

要帮助控制提示的大小，请尝试以下操作：

1. 熟悉要使用的模型所施加的限制。这些值可能因型号而异，因此在开始之前了解可用预算是多少，这一点很重要。

2. 在制作初始提示时要考虑预算，并考虑要为提示的任何动态元素节省多少钱。例如，用户输入、聊天记录、文档摘录等。
3. 在提示配置页面中，设置尾随历史记录大小限制，以限制提示中包含的对话回合数。
4. 在知识库配置向导中设置文档返回限制。您需要尝试在为LLM提供足够的上下文来执行任务之间取得适当的平衡，但不要超过代币限制或对延迟产生负面影响。
5. 留点缓冲区。不要为典型案例做预算，要考虑和尝试边缘案例，例如长输入查询、大型文档摘录或长时间的对话。

构建 MCP 服务器 Docker 镜像的步骤

要在 AWS 上将 MCP（模型上下文协议）服务器与生成式 AI 应用程序生成器一起使用，您需要首先构建并存储在私有 Amazon ECR 存储库中的 Docker 映像。

> **Note**
>
> 到目前为止，在 Amazon Bedrock AgentCore 运行时中部署的现有的 MCP 服务器无法导出到 GAAB 中。要将 MCP 服务器连接到通过 GAAB 创建的代理，则需要通过 GAAB 创建这些服务器。

步骤 1：创建您的 MCP 服务器

首先，你需要准备好你的 MCP 服务器实现。有关创建 MCP 服务器的详细说明，请参阅 Amazon Bedrock AgentCore 开发者指南-创建 MCP 服务器。

我们建议采用以下项目结构：

```
.
### __init__.py
### extras/
#   ### extra_dependencies.py
#   ### Dockerfile
### requirements.txt
### server.py <-- Server Entry point
```

对于 Dockerfile 结构，我们建议使用类似于以下示例的格式：

```
FROM ghcr.io/astral-sh/uv:python3.13-bookworm-slim
```

```
WORKDIR /app

# All environment variables in one layer
ENV UV_SYSTEM_PYTHON=1 \
    UV_COMPILE_BYTECODE=1 \
    UV_NO_PROGRESS=1 \
    PYTHONUNBUFFERED=1 \
    DOCKER_CONTAINER=1 \
    AWS_REGION=us-east-1 \
    AWS_DEFAULT_REGION=us-east-1

COPY requirements.txt requirements.txt
# Install from requirements file
RUN uv pip install -r requirements.txt

RUN uv pip install aws-opentelemetry-distro>=0.10.1

# Signal that this is running in Docker for host binding logic
ENV DOCKER_CONTAINER=1

# Create non-root user
RUN useradd -m -u 1000 bedrock_agentcore
USER bedrock_agentcore

EXPOSE 9000
EXPOSE 8000
EXPOSE 8080

# Copy entire project (respecting .dockerignore)
COPY . .

# Use the full module path
CMD ["opentelemetry-instrument", "python", "-m", "server"]
```

第 2 步：在本地测试您的 MCP 服务器

在部署到 AWS 之前，务必在本地测试您的 MCP 服务器，以确保其按预期运行。有关本地测试的详细说明，请参阅 Amazon Bedrock AgentCore 开发者指南-在本地测试您的 MCP 服务器。

第 3 步：部署到 Amazon ECR

在本地创建并测试 MCP 服务器后，请按照以下步骤将其部署到 Amazon ECR：

1. 确保您安装了最新版本的 AWS CLI 和 Docker。有关更多信息，请参阅 Amazon ECR 入门。
2. 检索身份验证令牌并向注册表对 Docker 客户端进行身份验证。使用 AWS CLI：

```
aws ecr get-login-password --region us-east-1 | docker login --username AWS --
password-stdin <account-id>.dkr.ecr.us-east-1.amazonaws.com
```

3. 使用以下命令构建 Docker 镜像。有关从头开始构建 Docker 文件的信息，请参阅 Docker 文档。如果您的镜像已经构建，则可以跳过此步骤：

```
docker build -t <repository-name> .
```

4. 构建完成后，标记您的映像，以便您可以将图像推送到此存储库：

```
docker tag <repository-name>:latest <account-id>.dkr.ecr.us-east-1.amazonaws.com/
<repository-name>:latest
```

5. 运行以下命令将此映像推送到您新创建的 AWS 存储库：

```
docker push <account-id>.dkr.ecr.us-east-1.amazonaws.com/<repository-name>:latest
```

有关完整的部署说明，请参阅 Amazon Bedrock AgentCore 开发人员指南-将您的 MCP 服务器部署到 AWS。

第 4 步：在 GAAB 中使用 ECR URI

成功将 Docker 镜像推送到 Amazon ECR 后，从 ECR 控制台复制镜像 URI。通过 AWS 上的生成式 AI 应用程序生成器部署向导部署 MCP 服务器时，您将使用此 URI。

创建不同 MCP 网关目标的步骤

Amazon Bedrock AgentCore Gateway 允许您将现有的 AWS 服务转换为可供代理使用的 MCP 工具。APIs 网关支持多种目标类型，使您能够无缝集成各种后端服务。

支持以下目标类型：

- Lambda 目标：将 AWS Lambda 函数转换为 MCP 工具。有关详细说明，请参阅 Amazon Bedrock AgentCore 开发者指南-添加 Lambda 目标。
- OpenAPI 目标：使用 OpenAPI 规范将 RES APIs T 定义为 MCP 工具并将其公开。有关详细说明，请参阅亚马逊 Bedrock AgentCore 开发者指南-OpenAPI 架构。

- Smithy 目标：使用 Smithy 模型定义构建 MCP 工具，实现类型安全的 API 集成。有关详细说明，请参阅 Amazon Bedrock AgentCore 开发者指南——构建 Smithy 目标。
- MCP 服务器目标：通过 URL 端点直接连接到外部 MCP 服务器，允许您集成现有的 MCP 服务器。有关详细说明，请参阅 Amazon Bedrock AgentCore 开发者指南-MCP 服务器目标。

有关创建 MCP 网关目标的其他示例和教程，请访问 Amazon Bedrock AgentCore 示例存储库。

配置知识库

本节介绍如何将数据提取到您为解决方案选择的知识库中。该解决方案目前支持 Amazon Kendra 和 Amazon Bedrock 知识库作为基于 RAG 的用例部署的知识库。

Amazon Kendra

如果您使用 Amazon Kendra 作为知识库，请参阅 Amazon Kendra 开发者指南，了解如何使用各种数据源连接器来帮助您从多种来源提取数据。

重要：为防止数据意外丢失，在删除部署或堆栈时，该解决方案不会自动删除 Kendra 索引（无论是由解决方案创建的还是其他方式创建的）。如果您想删除知识库并停止产生成本，请参阅 “手动卸载” 部分，详细了解保留哪些资源以及如何清理这些资源。

Amazon 基岩知识库

Amazon Bedrock 知识库可以由各种不同的矢量存储作为后盾，每个矢量库都能够为您的数据编制索引。要设置和填充您的知识库，请查阅 Amazon Bedrock 用户指南。具体而言，你会希望：

- 首先设置您的数据源
- 然后在支持的矢量存储中为您的知识库设置向量索引。请注意，如果您在创建知识库时使用 Bedrock 控制台中的 “快速创建新的矢量存储” 选项，则可以跳过此选项。
- 最后，您可以创建知识库并同步配置的数据源。

高级知识库设置

高级知识库设置（例如知识库筛选和带有基于角色的访问控制的 RAG）可用于该解决方案。知识库筛选可以应用于任一知识库，而带有基于角色的访问控制的 RAG 专门适用于 Amazon Kendra。

知识库筛选

该解决方案允许您在向导知识库步骤的 “高级 RAG 配置” 部分部署用例时指定 Amazon Kendra 属性筛选器或 Bedrock 知识库检索筛选器。这些筛选器定义了如何查询知识库中的数据源，例如搜索策略、要查询的底层文档的语言等。

在这两种情况下，都使用 JSON 对象根据每个服务文档（如上面的链接）中指定的格式指定过滤器设置。

示例 1：Kendra AttributeFilter

```
{
 "EqualsTo": {
 "Key": "_language_code",
 "Value": {
 "StringValue": "es"
 }
 }
}
```

示例 2：基岩 RetrievalFilter

```
{
 "equals": {
 "key": "language",
 "value": "es"
 }
}
```

使用 Amazon Kendra 实现基于角色的访问控制的 RAG

基于角色的访问控制 (RBAC) 允许控制哪些用户或群组可以访问您的 Amazon Kendra 索引中的某些文档或在搜索结果中查看某些文档。要使用 AWS 上的生成式 AI 应用程序生成器 (GAAB) 用例为您的 Amazon Kendra 索引 ID 配置 RBAC，请按照以下步骤操作：

1。配置亚马逊 Kendra 索引

1. 确保您已创建一个 Amazon Kendra 索引，并向其中添加了至少一个数据源。
2. 根据用户组为数据源配置访问控制。对于 S3 数据源，请按照文档中的说明使用在 Amazon Cognito 用户池中创建的相同组名来设置访问控制列表 (ACLs)。这样可以确保用户只能根据其群组成员资格访问他们有权查看的文档和搜索结果。

> **Note**
>
> 在您创建的 Kendra 索引的 “用户访问控制” 下，将基于令牌的用户访问控制保留为 “否”。 当您在步骤 2 中启用基于角色的访问控制时，AWS 上的生成式 AI 应用程序生成器会从用户身份验证令牌中提取相应的声明并创建属性筛选器。

2。使用 GAAB 部署向导部署 RAG 用例

1. 按照 GAAB 部署向导中的屏幕向导说明进行操作，直到进入向导的步骤 4 来配置 RAG。
2. 在部署向导的 “选择知识库” 步骤中，选择 Amazon Kendra 作为知识库类型。
3. 指定您是否已有 Amazon Kendra 索引，或者是否要创建新的索引。如果您已有索引，请提供您的 Amazon Kendra 索引的 ID，该索引已根据用户组配置了访问控制列表 (ACLs)。
4. 启用 “基于角色的访问控制” 选项。此选项可确保根据用户的角色和群组权限筛选从 Amazon Kendra 索引返回的搜索结果。
5. 查看并部署用例。

3。配置 Amazon Cognito

1. 找到您的 GAAB 部署所使用的 Amazon Cognito 用户池。此 Amazon Cognito 用户池通常由主部署控制面板 CloudFormation 堆栈创建。
2. 在 Amazon Cognito 用户池中创建新用户。创建用户时，选择 “发送电子邮件邀请” 选项，这样用户就可以通过电子邮件收到临时登录凭证。这样，新用户就可以注册并访问GAAB应用程序。
3. 在 Amazon Cognito 用户池中创建用户组。确保群组名称与您的 Amazon Kendra 索引中配置的群组完全匹配。 ACLs这对于启用 RBAC 至关重要，因为用户的群组成员资格将决定他们可以访问的搜索结果。
4. 根据用户的角色和访问权限将用户分配到相应的群组。必须将用户添加到 Amazon Kendra 索引 ACL 所需的群组以及在 GAAB 部署期间创建的特定用例群组。这可确保用户拥有访问特定用例和相关搜索结果的必要权限。

通过执行这些步骤，您将为 GAAB 部署配置基于角色的访问控制 (RBAC)，确保用户只能根据分配的用户组和权限访问他们获得授权的信息和功能并与之交互。

> **Note**
>
> 截至目前，只有 Amazon Kendra 支持 RBAC 在 AWS 上的生成人工智能应用程序生成器中创建知识库。对于 Amazon Bedrock 知识库，不支持 RBAC，但您可以使用元数据筛选器来实现某种程度的筛选。有关更多信息，请参阅 Amazon Bedrock 用户指南。

配置您的提示

部署仪表板向导具有提示配置步骤，允许您自定义提示体验和模板，以指导用户与 AI 模型之间的交互。正确配置这些设置对于从 AI 助手获得准确和相关的响应至关重要。

本节控制 AI 提示的整体体验和行为。

- 最大提示模板长度：此设置决定了提示模板的最大长度（以字符为单位）。值越高，可以为 AI 模型提供更多的背景信息，从而可能获得更准确的响应。但是，过长的提示也可能带来噪音并对性能产生负面影响。对于 Amazon Bedrock 模型，最大提示模板长度（以字符为单位）的默认值是使用基础模型令牌限制计算得出的。如果您在 Bedrock 中编辑和更改模型名称，“重置为默认值” 按钮会突出显示，该按钮可用于采用新选择的模型的默认值。对于 SageMaker Amazon AI 模型，提供了合理的默认值，但建议您检查底层模型并相应地选择这些最大提示模板长度和输入文本长度。有关更多信息，请参阅 “管理模型代币限制的提示” 部分。
- 最大输入文本长度：此设置限制了用户输入文本的最大长度（以字符为单位）。较长的输入可能包含不相关的信息，从而增加了从 AI 模型中获得不相关或不准确响应的风险。
- 用户提示编辑：此选项允许您启用或禁用用户通过聊天界面修改提示模板的功能。禁用此功能可以帮助保持一致性并防止对提示进行意外更改。

提示模板

本节允许您定义 AI 模型将使用的实际提示模板。提示模板通常遵循一种结构，该结构包含各种组件的占位符，例如用户的输入、参考段落和聊天记录。

- 提示模板：这是主文本区域，您可以在其中编写或粘贴所需的提示模板。模板的制作应为人工智能模型提供必要的背景和说明。它通常包含以下占位符：
  - `{input}`：此占位符是 Sagemaker AI 部署的必填项，将替换为用户的输入或查询。
  - `{history}`：此占位符是 Sagemaker AI 部署的必填项，将替换为当前对话的聊天记录。
  - `{context}`：此占位符对于 RAG 部署是必需的，将替换为从配置的知识库中获得的文档摘录。

- 改写问题？ ：此选项（仅适用于 RAG 部署）决定在将用户的原始输入查询传递给 AI 模型之前，应改写还是消除歧义。改写查询的措辞有时可以帮助模型更好地理解用户的意图，从而可能获得更准确的响应。

在配置提示模板和体验时，必须在为 AI 模型提供足够的背景和说明与避免可能导致噪音或性能问题的过长或不相关的信息之间取得平衡。

高级提示设置

此部分允许您控制对话历史记录如何呈现给 AI 模型。

- 尾随历史记录的大小：此设置决定了最终提示中应包含的先前消息的数量。将此值设置为零将导致不会向提示模板或消除歧义提示模板中注入任何历史记录。请注意：即使设置为零，提示模板中仍需要有 {history} 占位符。在运行时，它将被替换为空字符串。
  - 注意：建议为此值提供偶数。提供奇数将导致仅返回配对交互的 AI 响应。
- Hum an Prefix：这是用于识别用户在对话历史记录中发送的消息的前缀。
- AI 前缀：这是用于识别 AI 模型在对话历史记录中返回的消息的前缀。

消歧提示配置

本节允许您在将用户输入发送到已配置的知识库之前配置行为和模板，以消除用户输入的歧义。

- 启用消歧义：此选项决定在将用户输入发送到知识库之前是否应消除歧义。
- 消歧提示模板：这是用于在连接到知识库时消除用户输入的歧义的提示模板。此提示生成的输出将用作发送到知识库的查询。禁用歧义消除功能将导致用户的原始查询不变地发送到知识库。

例如，启用消歧义功能后，后续用户会查询 “费用是多少？” 可以消除 “续订我的牌照要花多少钱？” ，从而获得更好的搜索查询。

使用已部署的文本用例

文本用例的内置 UI 旨在让业务用户能够快速浏览和试验管理员用户创建的部署。业务用户所做的配置更改仅在其会话中生效。业务用户必须与管理员用户共享这些更改，管理员用户可以用这些更改更新基础部署以供所有人使用。

聊天界面由以下组件组成：

- 聊天窗口
- 聊天输入框
- 设置
- 清晰的对话

聊天窗口

保持对话的不同转折。从右边开始的消息来自业务用户，从左边开始的消息来自配置的 LLM。所有 LLM 响应上都有一个小剪贴板图标，便于复制响应。

聊天输入框

聊天输入框固定在聊天窗口的底部。企业用户可以在此处输入要发送给 LLM 的消息。输入框正上方是连接状态。如果连接中断（例如，由于处于非活动状态），则下次发送聊天消息时会自动创建一个新的连接。由于 WebSocket 连接时间较长，此请求预计会花费更长的时间。

根据特定的配置，可能会对输入强制规定最大长度。如果超过此限制，则用户会收到警报，并且不会发送消息。

注意：如果将 RAG 与 Amazon Kendra 配合使用，检索 API 会将查询截断为 30 个代币字。如果预计用户输入的时间会更长，请评估这可能会如何影响搜索性能。

设置

为了使业务用户能够快速尝试不同的配置，我们提供了一个设置面板，允许 on-the-fly编辑某些部署配置选项

（例如，提示模板）。这些更改只能在新会话开始时进行。对话开始后，清除对话即可重新启用配置设置的编辑。

注意：管理员用户可以选择锁定部署的设置。他们可以在部署时通过向导在提示步骤中阻止实时编辑。

清晰的对话

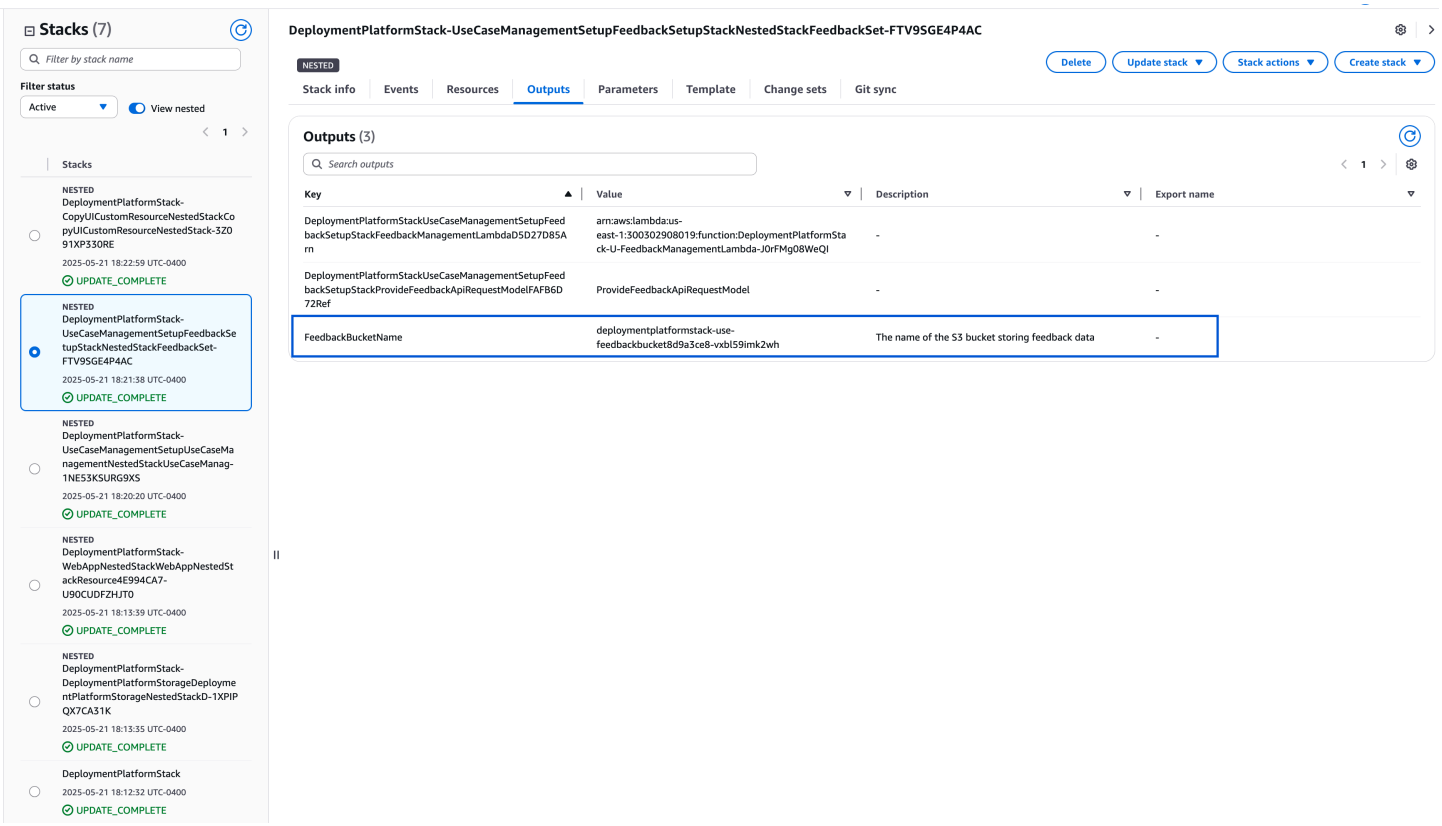
在对话过程中，该解决方案会保留聊天记录，从而实现对话体验。这样可以消除查询歧义和后续提问。要重置对话并删除该互动的所有聊天记录，请在聊天窗口顶部选择*清除对话*。清除对话后，将创建一个新的会话，该会话可以重新允许编辑设置。

访问和分析用户收集的反馈

从 v3.0.0 开始，部署仪表板部署了一个嵌套的反馈堆栈，该堆栈允许在仪表板上部署的 Text 和 Bedrock Agent 用例具有为其生成的响应收集反馈的功能。 LLM/Agent 特别是，用户可以提供正面或负面的反馈以及可选的评论。如果用户提供了负面反馈，他们可以进一步选择以下负面类别之一：“不准确”、“不完整或不足”、“有害”、“其他”。 and/or

用户提供反馈后，反馈将存储在 S3 存储桶中，按用例 ID、年份和月份进行分区。用例 ID 可以在部署控制面板中找到，Feedback S3 存储桶可以在部署控制面板堆栈的反馈嵌套堆栈的输出中找到：

描述部署堆栈-查找反馈存储桶名称



用户反馈以 API 请求的形式发送，其中包含最少的一组信息：

```
{
  "useCaseRecordKey": "a1b2c3d4-e5f6g7h8",
  "conversationId": "12345678-1234-1234-1234-123456789012",
  "messageId": "87654321-4321-4321-4321-210987654321",
  "rephrasedQuery": "What are the key features of the Generative AI Application Builder
 on AWS?",
  "sourceDocuments": [
```

```
    "s3://bucket-name/document1.pdf",
    "s3://bucket-name/document2.pdf"
  ],
  "feedback": "positive",
  "feedbackReason": [
    "Incomplete or insufficient"
  ],
  "comment": "The response was helpful but could include more details about important
 features."
}
```

然后，使用的 lambda 处理此有效负载`useCaseRecordKey`，用于识别部署时用例的正确配置。此配置用于获取反馈的具体细节，例如 ConversationTable姓名（包含所有对话以及人类和人工智能消息序列），这些信息进一步用于检索实际`userInput`和`llmResponse`。此反馈记录中还附有其他详细信息，例如 Bedrock Agent 用例的和 `modelProvider`bedrockModelId，以及使用此配置的文本用例等。`agentId` `agentAliasId`有关如何访问此配置的详细信息，请参阅下面的自定义反馈映射部分。每个传入的反馈请求都存储为 JSON 对象，对于文本用例，示例反馈记录可能如下所示：

```
{
   "useCaseId": "12345678-1234-1234-1234-123456789012",
   "useCaseRecordKey": "c07a2e3b-2f31b1e0",
   "userId": "22345678-1234-1234-1234-123456789012",
   "conversationId": "dd51de5d-5af1-4ec6-91d2-aadf14352109",
   "messageId": "32345678-1234-1234-1234-123456789012",
   "userInput": "What are its key features?",
   "rephrasedQuery": "What are the key features of the Generative AI Application
 Builder on AWS?",
   "llmResponse": "Generative AI Application Builder on AWS can help you build
 production ready enterprise chatbots rapidly.",
   "feedback": "negative",
   "feedbackReason": [
      "Incomplete or insufficient"
   ],
   "comment": "The response was helpful but could include more details about important
 features.",
   "timestamp": "2025-05-22T18:48:08.340Z",
   "feedbackId": "42345678-1234-1234-1234-123456789012",
   "useCaseType": "Text",
   "modelProvider": "Bedrock",
   "bedrockModelId": "amazon.nova-lite-v1:0",
   "ragEnabled": "false"
}
```

或者对于 Bedrock Agent 用例来说是这样的：

```
{
   "useCaseId": "12345678-1234-1234-1234-123456789012",
   "useCaseRecordKey": "c07a2e3b-2f31b1e0",
   "userId": "22345678-1234-1234-1234-123456789012",
   "conversationId": "dd51de5d-5af1-4ec6-91d2-aadf14352109",
   "messageId": "32345678-1234-1234-1234-123456789012",
   "userInput": "What are its key features?",
   "llmResponse": "Generative AI Application Builder on AWS can help you build
 production ready enterprise chatbots rapidly.",
   "feedback": "negative",
   "feedbackReason": [
      "Incomplete or insufficient"
   ],
   "comment": "The response was helpful but could include more details about important
 features.",
   "timestamp": "2025-05-22T18:48:08.340Z",
   "feedbackId": "42345678-1234-1234-1234-123456789012",
   "useCaseType": "Agent",
   "agentId": "AHFXUJCAK1",
   "agentAliasId": "KSEDKOS0BL"
}
```

然后，该反馈可用于进一步处理、分析和建模再训练/反馈回路。您还可以添加自定义映射以增强存储在反馈 lambda 中的反馈记录。

自定义反馈映射

部署仪表板包含一个LLMConfigTable，该密钥可以在部署仪表板堆栈的堆栈输出中找到LLMConfigTableName。 LLMConfigTable包含每个用例的配置，这些配置基于管理员在通过“部署控制面板” 向导部署用例时选择的设置。每个用例配置都由其标识。useCaseRecordKey以下是用例配置记录的示例：LLMConfigTable

```
{
   "key": "2dd76cfa-bc1a14da",
   "config": {
      "ConversationMemoryParams": {
        ...
      },
      "FeedbackParams": {
         "CustomMappings": {
```

```
            "NumberOfDocs": "$.KnowledgeBaseParams.NumberOfDocs",
            "ScoreThreshold": "$.KnowledgeBaseParams.ScoreThreshold"
         },
         "FeedbackEnabled": true
      },
      "IsInternalUser": "true",
      "KnowledgeBaseParams": {
         "KendraKnowledgeBaseParams": {
            "ExistingKendraIndexId": "d2831033-667f-4539-ab28-e6c7c7c5988b",
            "RoleBasedAccessControlEnabled": false
         },
         "KnowledgeBaseType": "Kendra",
         "NumberOfDocs": 5,
         "ReturnSourceDocs": false,
         "ScoreThreshold": 0.3
      },
      "LlmParams": {
         "BedrockLlmParams": {
            "BedrockInferenceType": "QUICK_START",
            "ModelId": "amazon.nova-lite-v1:0"
         },
         "ModelParams": {},
         "ModelProvider": "Bedrock",
         "PromptParams": {
            ...
         },
         "RAGEnabled": true,
         "Streaming": false,
         "Temperature": 0.1,
         "Verbose": false
      },
      "UseCaseName": "test-rag-usecase",
      "UseCaseType": "Text"
   }
}
```

如果为用例启用了反馈，则此配置将包含一个FeedbackParams对象，该对象允许其中的CustomMappings对象 JSONPaths 为所有其他字段指定要添加到存储在反馈 S3 存储桶中的反馈 JSON 记录中。例如，对于上面的示例用例配置，在以ScoreThreshold JSONPaths 根开头的CustomMappingsconfig对象中 CustomMappings 包含NumberOfDocs和。 JSONPath使用此配置，除了已经提供的字段外，存储在反馈 S3 存储桶中的每条 JSON 记录都将开始获取这 2 个额外值。

分析反馈数据

反馈数据作为 JSON 对象存储在 S3 中。以下是一些使反馈数据更易于访问和可操作的方法：

使用 AWS Glue 和亚马逊 Athena

AWS Glu e 和 Amazon Athena 提供了一种无服务器方式来对您的反馈数据进行分类、查询和分析。

AWS Glue 允许您创建 AWS Glue 爬虫来检查 S3 存储桶中的数据，推断其架构，并将所有相关元数据记录在目录中。之后，可以使用诸如Amazon Athena之类的服务来查询数据。

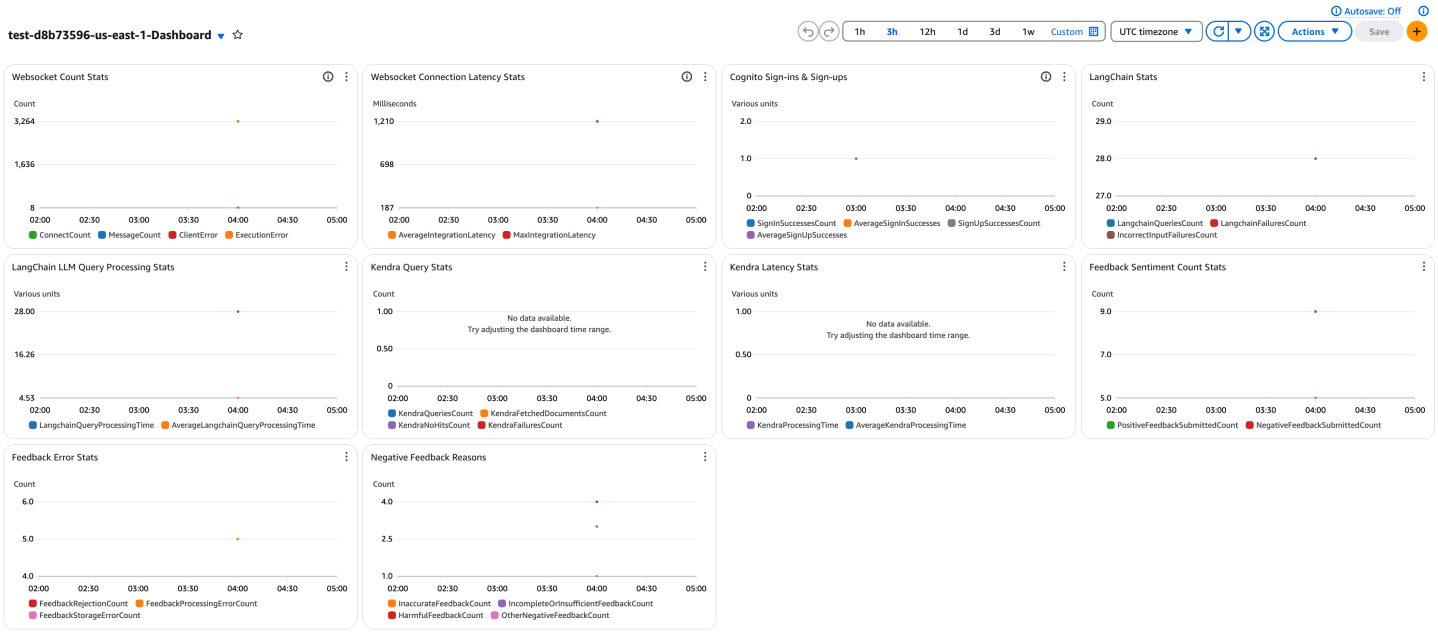
您可以参阅 AWS Athena 文档，了解使用 A WS Glue 数据目录将反馈 S3 存储桶与亚马逊 Athena 连接起来的步骤。您还可以使用 Glue 的一些更强大的功能对这些数据执行提取转换和加载 (ETL) 作业，并将其转换为适合您的分析或模型再训练用例的格式。使用 Glue，您可以执行诸如筛选具有特定反馈类型的记录、填写任何缺失信息之类的操作，还可以将这些数据加载到其他存储位置，例如另一个 S3 存储桶或其他 AWS 数据存储。

> **Note**
>
> 根据您的用例，可以考虑将 Glue 爬虫安排为定期（例如每周）运行，而不是每晚运行，以优化成本，因为反馈数据可能很少。

使用解决方案的 CloudWatch 仪表板

您还可以访问包含解决方案的控制面板，该CloudWatch 仪表板可以根据每个用例为您提供正面和负面反馈、负面反馈原因类别等的趋势。您可以在 AWS CloudWatch 控制台内的控制面板中使用您的用例名称找到此控制面板：

描绘用例仪表板 CloudWatch



您还可以在此控制面板中构建其他小组件或创建 Amazon Quick Sight 控制面板。

反馈数据分析的最佳实践

- 在 S3 存储桶上@@ 实施数据生命周期策略，将较旧的反馈数据存档到成本较低的存储层
- 为每个用例创建单独的分析，以确定特定于模型的改进机会
- 建立反馈阈值，以便在负面反馈超过可接受水平时触发警报
- 定期@@ 导出关键见解，以便与利益相关者和模型改进团队共享

查看部署的操作指标

部署仪表板和用例堆栈都带有自己的 CloudWatch 仪表板，用于跟踪解决方案的各种操作指标。您可以使用这些 CloudWatch 仪表板来帮助比较不同的部署。要访问仪表板，请执行以下操作：

1. 导航至 CloudWatch 控制台。
2. 通过查找堆栈名称或通用唯一标识符 (UUID) 来搜索预先构建的仪表板。

例如，文本用例附带了跟踪 WebSocket 连接数、用户登录和注册数量、LLM 处理完成操作所花费的时间等的图表。客户可以使用这些图表来比较部署的各种 _定量_ 指标。

Example

很难比较应用于不同用例的不同模型的定性结果。使用克隆功能可以快速启动多个部署，以便您可以并排比较输出。

访问 CloudWatch 日志见解

此解决方案记录了 Lambda 函数的错误、警告、信息和调试消息。要选择要记录的消息类型，请执行以下操作：

1. 在 AWS Lambda 控制台中找到适用的函数。
2. 添加 POWERTOOLS_LOG_LEV EL 环境变量
3. 将变量设置为适用的消息类型。

有关更多说明，请参阅 AWS Lambda 开发人员指南中的创建 Lambda 环境变量。

下表列出了您可以选择的日志级别类型。

级别	说明
ERROR (错误)	日志包含有关导致操作失败的任何内容的信息。
WARNING	日志包含任何可能导致函数不一致但不一定会导致操作失败的信息的信息。日志还包括错误消息。
信息	日志包含有关函数运行方式的高级信息。日志还包括错误和警告消息。
调试	日志包含在调试函数问题时可能有用的信息。日志还包括错误、警告和信息消息。

使用以下步骤向该解决方案添加 CloudWatch 日志见解。

1. 确定相关的日志组：
   a. 登录 A WS CloudFormation 控制台。
   b. 选择您的目标堆栈。

c. 选择资源选项卡并搜索您的目标 Lambda 函数。

d. 登录 AWS Lambda 控制台并选择您的每个目标 Lambda 函数。

e. 对于每个目标 Lambda 函数，选择监控选项卡，然后选择查看 CloudWatch 日志。

f. 复制要从中提取见解的日志组的名称。

2. 导航至 Amazon CloudWatch 控制台。
3. 在导航菜单上的 “日志” 下，选择 “日志见解”。
4. 在 “日志见解” 页面上，选择 “日志” 选项卡。
5. 搜索步骤 1 中的日志组名称。
6. 复制以下示例查询之一，然后将其粘贴到查询字段中：

a. 要识别所有客户机异常，请执行以下操作：

```
fields @message
|filter @message like /(?i)Exception/|stats count(*) as exceptionCount by @message
```

b. 要按函数名检索调用次数，请执行以下操作：

```
stats count(*) by function_name
```

c. 要检索五分钟间隔内的调用次数，请执行以下操作：

```
stats count(*) as invocations by bin(5m)
```

d. 要检索所有 AWS X-Ray 跟踪，请执行 IDs以下操作

```
filter @message like "XRAY TraceId"
|parse @message "XRAY TraceId: * " as traceId|stats count(*) by traceId
```

e. 要检索与特定 X-Ray Trace ID 相关的日志，请执行以下操作：

```
filter @message like "your-traceid-here"
```

f. 要检索未经授权的 WebSocket 错误：

```
fields
@ingestionTime,
@log,
@logStream,
@message
```

```
@requestId,
@timestamp,
errorMessage,
errorType
|filter @message like /Unauthorized/ and @message like /websocket/|sort @timestamp
 desc
```

g. 要检索已发布的指标数量，请执行以下操作：

```
filter @message like "CloudWatchMetrics"
|parse @message /"Metrics":\s*\[(?<metrics>.*?)\]/|stats count(*) as metric_count
 by metrics
```

开发人员指南

本节提供解决方案的源代码、集成指南、自定义指南和 API 参考。

源代码

访问我们的GitHub 存储库，下载此解决方案的源文件，并与其他人共享您的自定义设置。

AWS 上的生成式 AI 应用程序生成器模板是使用 AWS C loud Development Kit (AWS CDK) 生成的。有关其他信息，请参阅 README.md 文件。

集成指南

整个解决方案都设计为易于扩展。此解决方案的编排层是使用LangChain构建的。您可以将 LangChain（或为这些组件提供 LangChain 连接器的第三方）支持的任何模型提供者、知识库或对话存储器类型添加到此解决方案中。

支持扩展 LLMs

要添加其他模型提供者，例如自定义 LLM 提供程序，必须更新解决方案的以下三个组件：

1. 创建新的 `TextUseCase` CDK 堆栈，该堆栈将部署使用您的自定义 LLM 提供程序配置的聊天应用程序：
   a. 克隆此解决方案的GitHub 存储库，并按照 README.m d 文件中提供的说明设置您的构建环境。
   b. 复制（或新建）`source/infrastructure/lib/bedrock-chat-stack.ts`文件，将其粘贴到同一目录中，然后将其重命名为`custom-chat-stack.ts`。
   c. 将文件中的类重命名为合适的类，例如`CustomLLMChat`。
   d. 您可以选择将 Secrets Manager 密钥添加到此堆栈中，该堆栈用于存储您的自定义 LLM 的凭据。您可以在模型调用期间在下一段中讨论的聊天 Lambda 层中检索这些凭证。
2. 构建并附加一个 Lambda 层，其中包含待添加的模型提供者的 Python 库。对于亚马逊 Bedrock 用例聊天应用程序，`langchain-aws`Python 库在 LangChain 包顶部包含自定义连接器，用于连接到 AWS 模型提供商（亚马逊 Bedrock 和 A SageMaker I）、知识库（亚马逊 Kendra 和 Amazon Bedrock 知识库）和内存类型（例如 DynamoDB）。同样，其他模型提供商也有自己的连接器。该层可帮助您附加此模型提供者的 Python 库，以便您可以在调用 LLM 的聊天 Lambda 层中使用这些连接器（步骤 3）。在此解决方案中，使用自定义资产捆绑器来构建 Lambda 层，这些层使用 CDK 方面进行附加。要为自定义模型提供程序库创建新层，请执行以下操作：

a. 导航到source/infrastructure/lib/utils/lambda-aspects.ts文件中的LambdaAspects类。

b. 按照有关如何扩展文件中提供的 Lambda 方面类的功能（例如添加getOrCreateLangchainLayer方法）的说明进行操作。要使用这个新方法（例如，getOrCreateCustomLLMLayer），还需要更新source/infrastructure/lib/utils/constants.ts文件中的LLM_LIBRARY_LAYER_TYPES枚举。

3. 扩展 chat Lambda 函数以实现新提供程序的生成器、客户端和处理程序。

source/lambda/chat包含不同的 LangChain 连接 LLMs 以及用于构建这些连接的支持类 LLMs。这些支持类遵循生成器和面向对象的设计模式来创建 LLM。

每个处理程序（例如bedrock_handler.py）首先创建一个客户端，检查环境中是否存在所需的环境变量，然后调用一个get_model方法来获取 LangChain LLM 类。然后调用 generate 方法来调用 LLM 并获取其响应。 LangChain 目前支持 Amazon Bedrock 的直播功能，但不支持 SageMaker AI。基于流式传输或非流式传输功能，调用相应的 WebSocket 处理程序（WebsocketStreamingCallbackHandler或WebsocketHandler），使用post_to_connection方法将响应发送回 WebSocket 连接。

该clients/builder文件夹包含有助于使用生成器模式构建 LLM Builder 的类。首先，从 DynamoDB 配置存储中检索，该存储库存储了有关要构建的知识库类型、对话记忆和模型的详细信息。use_case_config它还包含相关的模型详细信息，例如模型参数和提示。然后，Builder 可以帮助执行以下步骤：创建知识库、创建对话记忆以维护 LLM 的对话上下文、为直播和非直播案例设置适当的 LangChain 回调，以及根据提供的模型配置创建 LLM 模型。DynamoDB 配置是在您从部署控制面板部署用例时存储的（或者当用户在没有部署控制面板的独立用例堆栈部署中提供用例时）。

该clients/factories子文件夹有助于根据 LLM 配置设置相应的对话记忆和知识库类。这样可以轻松扩展到您希望实现支持的任何其他知识库或内存类型。

shared子文件夹包含知识库和对话记忆的特定实现，这些实现由构建器在工厂内部实例化。它还包含 Amazon Kendra 和 Amazon Bedrock 知识库检索器，用于检索 RAG 用例的文档，以及 LLM 模型使用的回调。 LangChain LangChain

这些 LangChain 实现使用 LangChain 表达式语言 (LCEL) 将对话链组合在一起。RunnableWithMessageHistoryclass 用于维护与自定义 LCEL 链的对话历史记录，从而使诸如返回源文档和使用发送到知识库的改写（或消歧义）问题之类的功能也可以发送到 LLM。

要创建自己的自定义提供程序实现，您可以：

a. 复制bedrock_handler.py文件并创建您的自定义处理程序（例如，custom_handler.py），该处理程序将创建您的自定义客户端（例如，CustomProviderClient）（在以下步骤中指定。）

b. 复制bedrock_client.py到客户文件夹。将其重命名为custom_provider_client.py（或您的特定模型提供商名称，例如CustomProvider）。适当地命名其中的类，例如继承CustomProviderClientLLMChatClient的类。

   您可以使用提供的方法LLMChatClient或编写自己的实现来覆盖这些方法。

   该get_model方法构建CustomProviderBuilder（参见以下步骤），并使用构建器步骤调用构造聊天模型construct_chat_model的方法。此方法在生成器模式中充当导向器。

c. 将其复制clients/builders/bedrock_builder.py并重命名为，其中的类继承 LLMBuilder (llm_builder.py)。custom_provider_builder.py CustomProviderBuilder您可以使用提供的方法 LLMBuilder 或编写自己的实现来覆盖这些方法。生成器步骤在客户端的construct_chat_model方法中按顺序调用set_model_defaults，例如set_knowledge_base、和set_conversation_memory。

   该set_llm_model方法将使用使用之前调用的方法设置的所有值来创建实际的 LLM 模型。具体而言，您可以根据从 DynamoDB 中的 LLM 配置中检索到的 RAG (CustomProviderRetrievalLLMCustomProviderLLM) 或非 RAG () LLM。rag_enabled variable

   此配置是在LLMChatClient类的retrieve_use_case_config方法中获取的。

d. 根据你CustomProviderLLM需要 RAG 还是非 RAG 用例，在llm_models子文件夹中实现或CustomProviderRetrievalLLM实现。对于非 RAG 和 RAG 用例，实现这些模型的大多数功能分别在它们的BaseLangChainModel和RetrievalLLM类中提供。

   您可以复制llm_models/bedrock.py文件并进行必要的更改，以调用引用您的自定义提供程序的 LangChain 模型。例如，Amazon Bedrock 使用一个ChatBedrock类来创建聊天模型。LangChain

   生成方法使用 L LangChain CEL 链生成 LLM 响应。

   您也可以使用该get_clean_model_params方法根据您的模型要求对模型参数 LangChain 进行消毒。

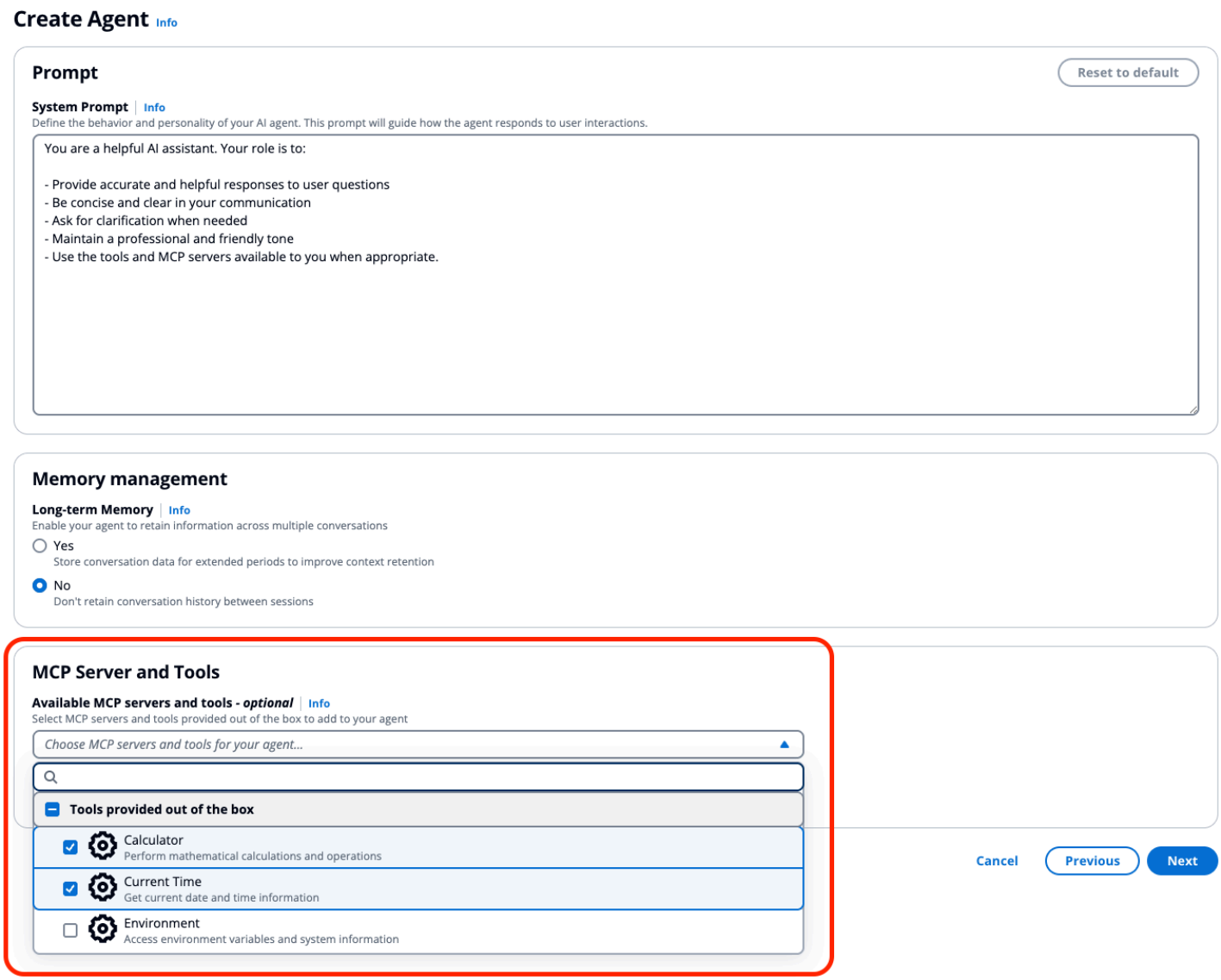
扩展支持的 Strands 工具

该解决方案使您能够构建和部署 MCP 服务器、AI 代理和多代理工作流程。在 Agent Builder 体验中，您可以连接 MCP 服务器，为代理提供更多功能。除了 MCP 服务器外，您还可以利用 Strands 提供的内置工具（解决方案使用的底层框架）。

该解决方案开箱即用，预先配置了以下 Strands 工具：

- 当前时间（默认启用）
- 计算器（默认启用）
- 环境

在 Agent Builder 向导中选择 MCP 服务器和工具，其中显示了内置 Strands 工具



要使用其他 Strands 工具扩展您的代理，请按照本节中概述的四步流程进行操作。

第 1 步：找到股线工具

浏览可用的 Strands 工具，确定要使用的工具。每种工具都有特定的功能和配置要求。

例如，要添加 Amazon Bedrock 知识库检索功能，您可以使用检索工具。

步骤 2：更新 SSM 参数

要在 Agent Builder 部署用户界面中提供工具，请更新定义支持哪些 Strands 工具的 AWS Systems Manager Parameter Store 参数。

1. 导航到您的 AWS 账户中的 AWS Systems Manager Parameter Store。

2. 找到参数：/gaab/<stack-name>/strands-tools
3. 使用以下 JSON 结构将您的工具配置添加到现有列表的末尾：

```
{
  "name": "Bedrock KB Retrieve",
  "description": "Retrieve information from Bedrock Knowledge Base",
  "value": "retrieve",
  "category": "AI",
  "isDefault": false
}
```

字段	说明
name	代理生成器用户界面中显示的显示名称
描述	该工具功能的简要描述
值	在 Strands 工具包中定义的确切工具名称
category	用于在 UI 中对工具进行分组的组织类别
是默认	默认情况下，是否应为新代理启用该工具

步骤 3：配置环境变量

许多 Strands 工具都需要环境变量进行配置。您可以通过两种方式设置这些变量：

选项 1：在 AgentCore 运行时直接配置

使用所需的环境变量直接在 Amazon Bedrock AgentCore Runtime 上更新已部署的代理。

选项 2：部署向导中的模型参数

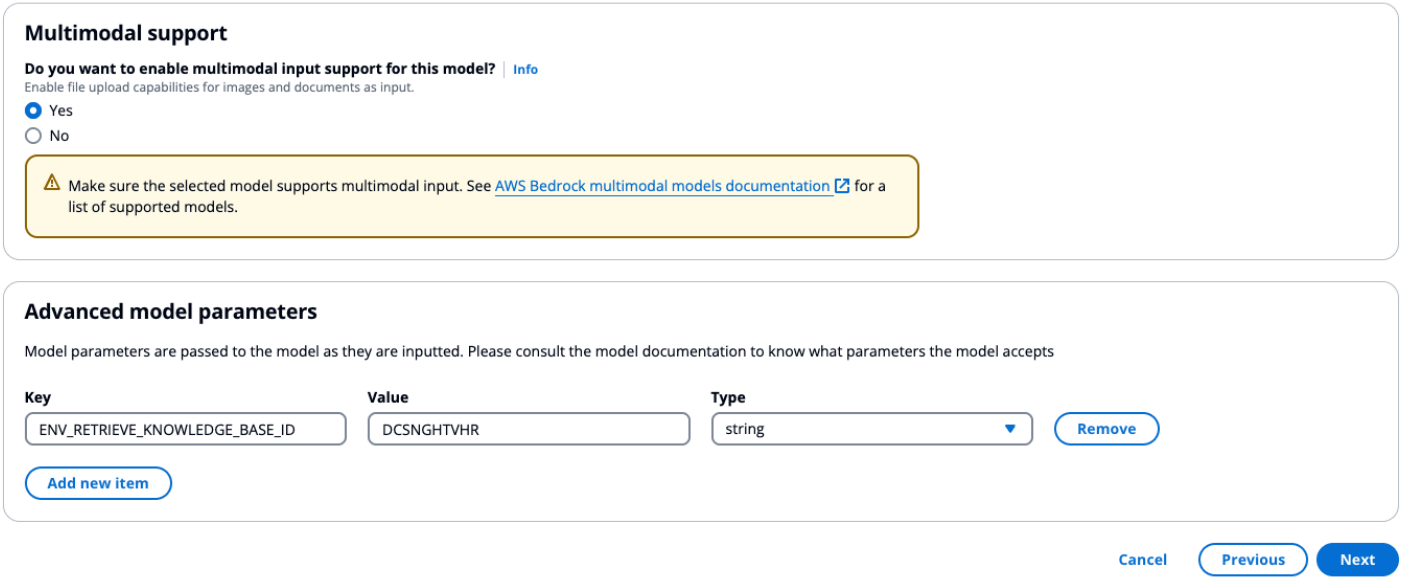
在 Agent Builder 向导的模型选择步骤中，使用模型参数部分添加环境变量。遵循命名约定的环境变量ENV_<ALL_CAPS_TOOL_NAME>_<env_variable_name>将在运行时自动加载到代理的执行环境中，即<env_variable_name>。

例如：

- ENV_RETRIEVE_KNOWLEDGE_BASE_ID 变为 KNOWLEDGE_BASE_ID

- ENV_RETRIEVE_MIN_SCORE 变为 MIN_SCORE

高级模型参数部分显示了 ENV_RETRIEVE_KNOWLEDGE_BASE_ID 配置



请参阅特定工具的文档或源代码以确定所需的环境变量。对于检索工具，您可以在源代码中找到配置选项。

步骤 4：添加 IAM 权限

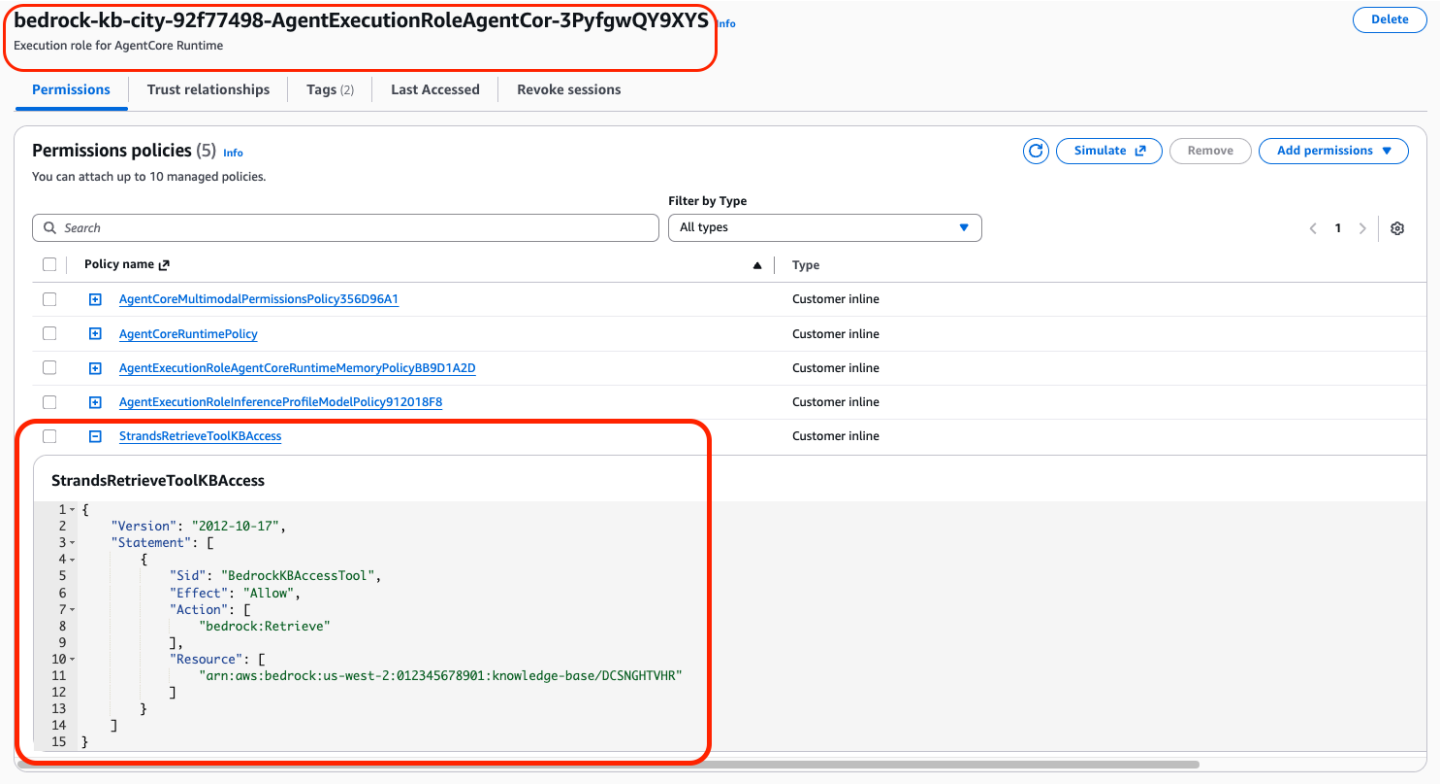
手动向您的 AgentCore 运行时执行角色添加任何必要的 IAM 权限，以允许代理使用该工具。

例如，要在 Amazon Bedrock 知识库中使用检索工具，请执行以下操作：

1. 导航到您的 AWS 账户中的 IAM 控制台。
2. 找到代理的 AgentCore 运行时执行角色。
3. 添加以下权限：

```
{
  "Effect": "Allow",
  "Action": "bedrock:Retrieve",
  "Resource": "arn:aws:bedrock:region:account-id:knowledge-base/knowledge-base-id"
}
```

显示附加到 AgentCore 运行时执行角色的 StrandsRetrieveToolKBAccess 策略的 IAM 控制台

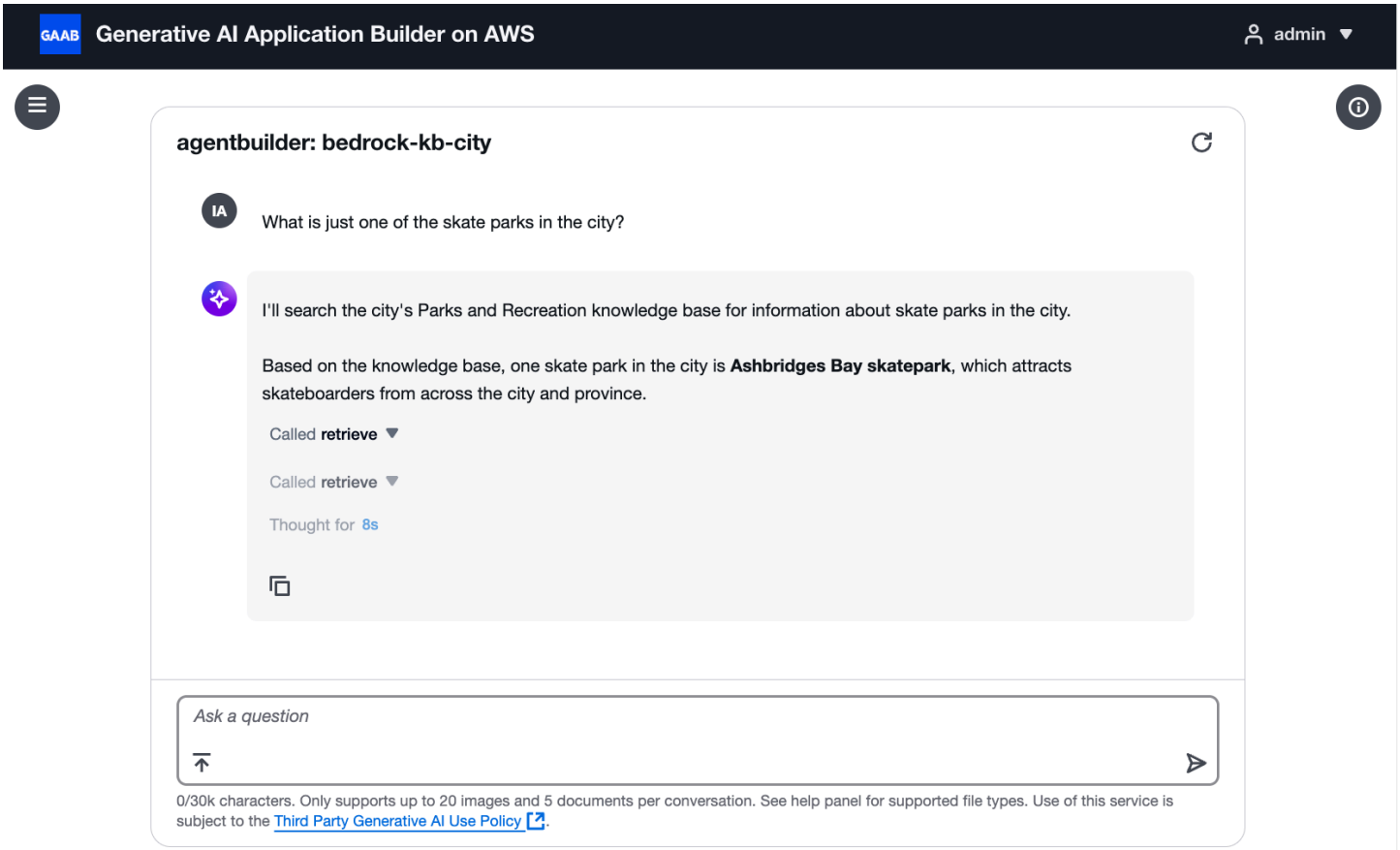


所需的特定权限将因工具而异。请查阅该工具的文档和 AWS 服务文档，以确定适当的 IAM 权限。

步骤 5：测试代理

完成配置步骤后，测试您的代理以验证该工具是否正常运行。您应该在代理的执行日志和响应中看到工具调用。

代理成功使用检索工具回答了有关滑板公园的问题



> **Note**
>
> 有关可用 Strands 工具及其功能的完整列表，请参阅 Strands 社区工具文档。

扩展支持的知识库和对话记忆类型

要添加对话记忆或知识库的实现，请在`shared`文件夹中添加所需的实现，然后编辑工厂和相应的枚举以创建这些类的实例。

当您提供存储在参数存储库中的 LLM 配置时，将为您的 LLM 创建相应的对话存储器和知识库。例如，当指定`ConversationMemoryType`为 DynamoDB 时，将创建一个（`shared_components/memory/ddb_enhanced_message_history.py`内部可用）`DynamoDBChatMessageHistory`的实例。如果指定`KnowledgeBaseType`为 Amazon Kendra，则会创建一个`KendraKnowledgeBase`（内部可用`shared_components/knowledge/kendra_knowledge_base.py`）的实例。

生成和部署代码变更

使用`npm run build`命令生成程序。解决所有错误后，运行`cdk synth`生成模板文件和所有 Lambda 资产。

1. 您可以使用该`0/stage-assets.sh`脚本将任何生成的资产手动暂存到账户中的暂存存储桶中。
2. 使用以下命令部署或更新平台：

```
cdk deploy DeploymentPlatformStack --parameters AdminUserEmail='admin-
email@amazon.com'
```

任何其他 AWS CloudFormation 参数也应与AdminUserEmail参数一起提供。

定制指南

管理 Cognito 用户池

部署控制面板后，将创建一个 Amazon Cognito 用户池和一个管理员用户，为应用程序提供身份验证。此用户池在 “部署” 控制面板和所有用例中共享。在部署仪表板时创建的管理员用户将自动获得访问使用控制板部署的所有用例的权限。此机制是通过 Amazon Cognito 用户池组提供的。

从控制面板部署用例时，如果提供了电子邮件，则将在共享用户池中创建一个用户以及一个针对特定用例命名的用户组。然后，新创建的用户将被添加到群组中，授予该用户访问该用例的权限。

如果您想在给定用例中添加其他用户，则可以通过在 Cognito 用户池中创建用户并将他们添加到与您希望用户访问的用例相对应的组来实现。有关 step-by-step指南，请参阅在 AWS 管理控制台中创建新用户。

同样，如果要创建其他管理员用户，则必须创建一个新用户并将其添加到用户池中的管理员组。

用户名是通过将提供的电子邮件中的部分放在之前并附加生成的用例 UUID（或者`-admin`对于管理员用户而言）来创建的。@

在 “群组” 选项卡中，您可以看到已使用用例名称（如向导中所提供）和用例 UUID 自动创建了一个管理员组和每个用例的群组。

API 参考

本节提供解决方案的 API 参考。

部署控制面板

REST API	HTTP method	功能	授权来电者
/deployments	GET	获取所有部署。	亚马逊 Cognito 经过身份验证的 JWT 令牌
/deployments	POST	创建新的用例部署。	亚马逊 Cognito 经过身份验证的 JWT 令牌
/deployments/{useCaseId}	GET	获取单个部署的部署详细信息。	亚马逊 Cognito 经过身份验证的 JWT 令牌
/deployments/{useCaseId}	PATCH	更新给定的部署。	亚马逊 Cognito 经过身份验证的 JWT 令牌
/deployments/{useCaseId}	DELETE	删除给定的部署。	亚马逊 Cognito 经过身份验证的 JWT 令牌
/model-info/use-case-types	GET	获取部署的可用用例类型	亚马逊 Cognito 经过身份验证的 JWT 令牌
/model-info/{useCaseType}/providers	GET	获取给定用例类型的可用模型提供者	亚马逊 Cognito 经过身份验证的 JWT 令牌
/model-info/{useCaseType}/{providerName}	GET	获取给 IDs 定提供者可用的模型和用例类型	亚马逊 Cognito 经过身份验证的 JWT 令牌
/model-info/{useCaseType}/{providerName}/{modelId}	GET	获取有关给定模型的信息，包括默认参数。	亚马逊 Cognito 经过身份验证的 JWT 令牌

> **Note**
>
> OpenAPI 和 Swagger 文件也可以从 API Gateway 导出，以便更轻松地与 API 集成。请参见从 API Gateway 导出 REST API。

POST 和 PATCH 有效负载

有关/deployments端点的 POST 有效负载示例，请参阅下文，这将创建一个新的用例。

```
{
 "UseCaseName": "usecase1",
 "UseCaseDescription": "Description of the use case to be deployed. For display
 purposes", // optional
 "DefaultUserEmail": "placeholder@example.com", // optional, if not provided, the
 Cognito Group and User will not be created
 "DeployUI": true, // optional
 "VpcParams": {
 "VpcEnabled": true,
 "CreateNewVpc": false,
 // provide these if not creating new vpc
 "ExistingVpcId": "vpc-id",
 "ExistingPrivateSubnetIds": ["subnet-1", "subnet-2"],
 "ExistingSecurityGroupIds": ["sg-1", "sg-2"]
 },
 "ConversationMemoryParams": {
 "ConversationMemoryType": "DynamoDB",
 "HumanPrefix": "user", // optional
 "AiPrefix": "ai", // optional
 "ChatHistoryLength": 10 // optional
 },
 "KnowledgeBaseParams": {
 "KnowledgeBaseType": "Bedrock",
 // one of the following based on selected provider
 "BedrockKnowledgeBaseParams": {
 "BedrockKnowledgeBaseId": "my-bedrock-kb",
 "RetrievalFilter": {}, // optional
 "OverrideSearchType": "HYBRID" // optional
 },
 "KendraKnowledgeBaseParams": {
 "AttributeFilter": {}, // optional
 "RoleBasedAccessControlEnabled": true, // optional
 "ExistingKendraIndexId": "12345678-abcd-1234-abcd-1234567890ab",
```

```
 // provide the following in place of ExistingKendraIndexId if you want the solution to
 deploy an index for you
 "KendraIndexName": "index",
 "QueryCapacityUnits": 1, // optional
 "StorageCapacityUnits": 1, // optional
 "KendraIndexEdition": "DEVELOPER" // optional
 },
 "NoDocsFoundResponse": "Sorry, I couldn't find any relevant information for your
 query.", // optional
 "NumberOfDocs": 3, // optional
 "ScoreThreshold": 0.7, // optional
 "ReturnSourceDocs": true // optional
 },
 "LlmParams": {
 "ModelProvider": "Bedrock | SAGEMAKER",
 // one of the following based on selected provider
 "BedrockLlmParams": {
 "ModelId": "model-id", // use this for on demand models. Can't use with ModelArn
 "ModelArn": "model-arn", // use this for provisioned/custom models. Can't use with
 ModelId,
 "InferenceProfileId": "profile-id"
 "GuardrailIdentifier": "arn:aws:bedrock:us-east-1:123456789012:guardrail/my-
guardrail", // optional
 "GuardrailVersion": "1" // optional. Required if GuardrailIdentifier provided.
 },
 "SageMakerLlmParams": {
 "EndpointName": "some-endpoint",
 "ModelInputPayloadSchema": {},
 "ModelOutputJSONPath": "$."
 },
 // optional. Passes on arbitrary params to the underlying LLM.
 "ModelParams": {
 "param1": {
 "Value": "value1",
 "Type": "string"
 },
 "param2": {
 "Value": 1,
 "Type": "integer"
 }
 },
 // optional
 "PromptParams": {
 "PromptTemplate": "some template",
```

```
 "UserPromptEditingEnabled": true,
 "MaxPromptTemplateLength": 1000,
 "MaxInputTextLength": 1000,
 "DisambiguationPromptTemplate": "some disambiguation template",
 "DisambiguationEnabled": true
 },
 "Temperature": 1.0, // optional
 "Streaming": true, // optional
 "RAGEnabled": true, // optional. Must be true if providing KnowledgeBaseParams above.
 "Verbose": false // optional
 },
 "AgentParams": {
 "AgentType": "Bedrock",
 "BedrockAgentParams": {
 "AgentId": "agent-id",
 "AgentAliasId": "alias-id",
 "EnableTrace": true
 }
 },
 // optional
 "AuthenticationParams": {
 "AuthenticationProvider": "Cognito",
 "CognitoParams": {
 "ExistingUserPoolId": "user-pool-id",
 "ExistingUserPoolClientId": "client-id" // optional. If not provided, the solution
 will create a client for you in the provided pool
 }
 }

}
```

对于更新，结构与上面相同，但有一些注意事项：

- 无法更改用例名称
- 只有在 VPC 中部署使用案例后，才能更改安全组和子网。VPC 本身无法更改。
- 如果 Kendra 索引是作为知识库创建的，则无法更改该索引的配置（例如， KendraIndexName ） QueryCapacityUnits

共享用例 APIs

以下 REST API 端点可用于文本代理和 Bedrock Agent 用例：

REST API	HTTP method	功能	授权来电者
`/details/{useCaseConfigKey}`	`GET`	获取特定用例的配置详细信息。	亚马逊 Cognito 经过身份验证的 JWT 令牌

WebSocket API	功能	授权来电者
`/$connect`	启动 WebSocket 连接并对用户进行身份验证。	亚马逊 Cognito 经过身份验证的 JWT 令牌
`/$disconnect`	WebSocket 连接断开时调用终端节点。	亚马逊 Cognito 经过身份验证的 JWT 令牌

用例详情 API

details API 端点检索有关特定用例的信息：

```
GET /details/{useCaseConfigKey}
```

此端点返回特定用例的配置详细信息，包括模型参数、知识库设置和其他部署信息。它需要经过 Amazon Cognito 身份验证的 JWT 令牌才能进行授权。

文本用例

WebSocket API	功能	授权来电者
`/sendMessage`	向发送用户的聊天消息，以便按照配置 WebSocket 的 LLM 体验进行处理。	亚马逊 Cognito 经过身份验证的 JWT 令牌

REST API	HTTP method	功能	授权来电者
/feedback/{useCaseId}	POST	针对特定用例提交用户反馈。	亚马逊 Cognito 经过身份验证的 JWT 令牌

发送消息有效负载

如果您直接与 /sendMessage API 集成，则必须遵守以下请求和响应负载格式。

请求有效负载

```
{
 "action": "sendMessage",
 "question": "the message to send to the api",
 "conversationId": "", // If not provided, a new conversation will be created, with the
 conversationId returned in the response. All subsequent messages in that conversation
 (where history is retained), should provide the conversationId there.
 "promptTemplate": "", // Optional. Overrides the configured prompt
 "authToken": "XXXX" // Optional. accessToken from cognito flow. Required for RAG with
 RBAC
}
```

参数名称	Type	说明
action	String	目前，我们仅支持 “SendMessage” 操作 WebSocket
问题	String	要发送给 LLM 的用户输入
会话 ID	String	用于标识对话的 UUID。如果未提供，则将创建一个新的对话，并在响应中返回 conversationID。该对话中的所有后续消息（您希望保留在哪里）都应在此处提供 conversationID。history/context
提示模板	String[可选]	覆盖此消息的提示模板。如果为空或未提供，则默认为部署

参数名称	Type	说明
		时设置的提示符。必须为给定配置（即非 Rag Sagemaker AI 部署的 {history} 和 {input}）指定正确的占位符，如果所有部署都使用 RAG，则必须添加 {context}。
验证令牌	`String`[可选]	AccessToken 是从认知身份验证流程中获得的。使用基于角色的访问控制 (RBAC) 调用为 RAG 配置的聊天 websocket 端点时，这是必需的。此 JWT 令牌中的 cognito: groups 声明列表用于控制对 Kendra 索引中文档的访问。非 RAG 用例不需要此参数。对于禁用 RBAC 的 RAG 用例，也不是必需的。

响应负载

问题回答

对于每个查询， WebSocket API 将使用 1 个（如果禁用了流式传输）或许多（如果启用了流式传输）JSON 对象进行响应，其结构如下。

```
{
 "data": "some data",
 "conversationId": "id",
}
```

参数名称	Type	说明
data	`String`	如果启用了流式传输，则是来自 LLM 的响应的一部分，或者是整个响应。如果使用

参数名称	Type	说明
		流式传输，则将发送此格式的响应，数据内容为 END_CONVERSATION，以表明对单个问题的回答已结束。
会话 ID	`String`	此 SourceDocument 响应所属的对话的 ID。

源文件回复

如果您已将 RAG 用例配置为返回源文档，那么对于用于创建响应的每个源文档，您还将在每个响应的末尾收到以下有效负载。

```
{
 "sourceDocument": {
 "excerpt": "some excerpt from the",
 "location": "s3://fake-bucket/test.txt",
 "score": 0.500,
 "document_title": null,
 "document_id": null,
 "additional_attributes": null
 },
 "conversationId": "some-id"
}
```

参数名称	Type	说明
摘录	`String`	源文档的摘录。
位置	`String`	源文档的位置。这将取决于使用的数据源和知识库的类型，但可能是 s3 URIs 或网站之类的东西。
得分	`Number` | `String`	对文档与所提问题相对应的信心。对于 Bedrock，这将是一个从 0 到 1 的浮点数，对于

参数名称	Type	说明
		Kendra，这将是一个字符串（例如 HIG、LOW 等）。
文档标题	String	返回的源文档的标题。仅在使用 Kendra 时可用。
文档_ID	String	返回的源文档的 ID。仅在使用 Kendra 时可用。
其他属性	String	该字段将包含文档上所有其他属性，这些属性是在摄取时在知识库中自定义的。
会话 ID	String	此 SourceDocument 响应所属的对话的 ID。

反馈 API 有效负载

以下是/feedback/{useCaseId}端点的 POST 负载示例，它将针对特定用例提交用户反馈：

```
{
  "useCaseRecordKey": "12345678-12345678",
  "conversationId": "12345678-1234-1234-1234-123456789012",
  "messageId": "12345678-1234-1234-1234-123456789012",
  "feedback": "positive",
  "feedbackReason": ["accurate", "helpful"],
  "comment": "This response was very helpful.",
  "rephrasedQuery": "What are the key features of Amazon Bedrock?",
  "sourceDocuments": [
    "s3://bucket-name/document1.pdf",
    "s3://bucket-name/document2.pdf"
  ]
}
```

基岩代理用例

WebSocket API	功能	授权来电者
`/invokeAgent`	向发送用户的消息，WebSocket 以便使用配置的代理进行处理。	亚马逊 Cognito 经过身份验证的 JWT 令牌

InvokeAgent 负载

如果您要直接与集成`/invokeAgent` API，则必须遵守以下请求和响应负载格式。

请求负载

```
{
"action": "invokeAgent",
"inputText": "User query to the agent",
"conversationId": "", // Optional. Empty conversationId implies a new conversation.
 When not provided, a new conversationId will be created and returned with the
 response. All subsequent messages in the same conversation should provide the same
 conversationId (i.e. chat memory/history is maintained).
"authToken": "XXXX" // Optional. accessToken from cognito flow. If provided, it needs
 to be a valid JWT token associated with the user
}
```

参数名称	Type	说明
action	`String`	我们只支持对... 的`invokeAgent` WebSocket行动
输入文本	`String`	要发送给 LLM 的用户输入。
会话 ID	`String[Optional]`	唯一标识对话的 UUID。如果您不提供此值，则解决方案会创建一个新的对话并在响应中返回 convers ationID。该对话中的所有后续消息（您想保留

参数名称	Type	说明
		历史和上下文）都在此处提供了 convers ationID。
验证令牌	`String[Optional]`	A@@ ccessToken 是从亚马逊 Cognito 身份验证流程中获得的。此参数不是必需的。如果您提供了 JWT 令牌，则将对其进行验证。这有助于更轻松地扩展此解决方案。

响应有效载荷

问题回答

对于每个查询， WebSocket API 将使用一个（如果禁用了流式传输）或多个（如果启用了流式传输）JSON 对象进行响应，其结构如下。

```
{
 "data" "some data",
 "conversationId": "id",
 }
```

参数名称	Type	说明
data	`String`	代理调用的响应。
会话 ID	`String`	对话的 ID。

参考

本节包括有关此解决方案的数据收集的信息、相关资源的指针以及为该解决方案做出贡献的构建者列表。

支持的法学硕士提供商

该解决方案可以与以下法学硕士提供商集成：

1. Amazon Bedrock
   - 文档： https://aws.amazon.com/bedrock/
   - 支持的型号：
     - Amazon
       - Nova Lite
       - Nova Micro
       - Nova Pro
     - AI21 实验室
       - Jamba 1.5 Mini
       - Jamba 1.5 Large
     - Anthropic
       - Claude v3 俳句
       - Claude v3.5 十四行诗
       - Claude v3.7 Sonnet（通过使用推理配置文件）
     - Cohere
       - Command R
       - Command R+
     - 深度搜寻
       - Deepseek-R1（通过使用推理配置文件）
     - Meta
       - Llama 3
       - Llama 3.2（通过使用推理配置文件）
     - Mistral AI

    - Mistral 7B Instruct
    - Mistral 8x7b Instruct
  - 跨区域推理
    - 能够使用在与部署仪表板相同的区域中定义的推理配置文件

2. 亚马逊 SageMaker AI
   - 文档： https://aws.amazon.com/sagemaker/
   - 支持的型号：文本转文本模型

有关最新的模型参数、最佳实践和推荐用途，请参阅模型提供商提供的文档。

数据收集

此解决方案向 AWS 发送有关该解决方案使用情况的运营指标（“数据”）。我们使用这些数据来更好地了解客户如何使用此解决方案以及相关服务和产品。AWS 对这些数据的收集受 AWS 隐私声明的约束。

贡献者

- 塔雷克·阿卜杜纳比
- Majd Arbash
- 乔治·比尔登
- Mukit Bin Momin
- 迈克尔·康纳
- Johny Duval
- Nihit Kasabwala
- Ahern Knox
- 西蒙·克罗尔
- 迈克尔·林
- Tim Mekari
- 易卜拉欣·穆罕默德
- 奥马尔·拉德万·穆森
- 詹姆斯·尼克松

- Dekshitha Ravikumar
- Jae Shim
- Ajay Swamy
- 穆罕默德·塔哈
- Reet Takkar
- 迪米特里·奇卡季洛夫
- Jason Wreath
- Kamyar Ziabari

修订

发布日期：2023 年 10 月（最后更新时间：2025 年 1 月）

查看 GitHub 存储库中的 Changelog.md 文件，查看该软件的所有重要更改和更新。更改日志会清晰记录每个版本的改进和修复。

版权声明

客户有责任对本文档中的信息进行单独评测。本文档：（a）仅供参考；（b）代表当前提供的 AWS 产品和实操，如有更改，恕不另行通知；并且（c）AWS 及其附属机构、供应商或许可方不做任何承诺或保证。AWS 产品或服务“按原样”提供，不提供任何形式的保证、陈述或条件，无论是明示还是暗示。AWS 对其客户承担的责任和义务受 AWS 协议制约，本文档不是 AWS 与客户直接协议的一部分，也不构成对该协议的修改。

AWS 上的生成式 AI 应用程序生成器根据 A pache 许可版本 2.0 的条款获得许可。

> **Important**
>
> AWS 上的生成式 AI 应用程序生成器允许您使用自己选择的生成式 AI 模型，包括您可以选择使用的 AWS 不拥有或无法控制的第三方生成人工智能模型（“第三方生成式 AI 模型”），在 AWS 上构建和部署生成式人工智能应用程序。
>
> 您对第三方生成人工智能模型的使用受第三方生成人工智能模型提供商在您获得使用许可时向您提供的条款的约束（例如，他们的服务条款、许可协议、可接受的使用政策和隐私政策）。您有责任确保您对第三方生成人工智能模型的使用符合管理这些模型的条款以及适用于您的任何法律、法规、法规、政策或标准。
>
> 您还负责对您使用的第三方生成人工智能模型进行自己的独立评估，包括其输出以及第三方生成人工智能模型提供商如何使用根据您的部署可能传输给他们的任何数据。AWS 不对第三方生成式 AI 模型做出任何陈述、担保或担保，根据您与 AWS 达成的协议，这些模型是 “第三方内容”。根据您与 AWS 达成的协议，AWS 上的生成式 AI 应用程序生成器作为 “AWS 内容” 提供给您。

本文属于机器翻译版本。若本译文内容与英语原文存在差异，则一律以英文原文为准。