

白皮书：

基于Amazon Bedrock与飞书的高效企业智能助手解决方案

发布方：成都易定云科技有限公司

日期：2026年3月

版本：1.0

摘要

随着企业数字化转型的深入，如何将生成式AI能力无缝集成到日常办公协作平台（如飞书）中，已成为提升组织效率的关键。成都易定云科技有限公司（以下简称“易定云”）作为AWS高级合作伙伴，结合自身在云原生架构与企业服务集成领域的深厚积累，推出了基于Amazon Bedrock、AWS Lambda、飞书开放平台的全托管智能助手解决方案。

本白皮书将深入解析该方案的技术架构、部署实践、核心优势以及易定云在落地过程中的独特见解，为企业快速构建内部AI知识库与智能应用提供可复用的参考路径。

一、背景与挑战

在企业日常运营中，大量信息沉淀于文档、会议记录、历史工单中，员工往往面临“找不到、看不懂、用不上”的知识困境。传统搜索方式难以理解语义，而通用大模型又缺乏企业内部知识的支撑。

主要挑战

- 1.知识分散： 文档存放于不同系统，缺乏统一检索入口。
- 2.集成复杂： AI能力与飞书等办公软件的集成涉及多平台对接，开发周期长。
- 3.运维负担： 缺乏基础设施即代码（IaC）管理，环境一致性难保障。

二、解决方案架构与易定云实践

易定云基于AWS CDK（Cloud Development Kit）全代码化构建了“飞书智能助手”解决方案。我们坚持“基础设施即代码”的原则，确保从开发到生产的环境一致性，并实现一键部署。

2.1整体架构

本方案采用事件驱动与托管服务架构，主要包括以下模块：

接入层

飞书开放平台作为交互入口，通过事件订阅与机器人消息与AWS侧交互。

API层：

使用Amazon API Gateway接收飞书回调，确保高可用与安全鉴权。

逻辑层：

AWS Lambda负责处理消息路由、调用大模型、解析意图。

知识层：

Amazon Bedrock知识库（Knowledge Base）负责RAG（检索增强生成），S3桶存储源文档。

易定云见解：

我们认为，在企业级AI应用中，RAG（检索增强生成）的准确率直接决定用户体验。因此，我们不仅使用Bedrock知识库的标准功能，还针对中文文档格式进行了预处理优化，并设计了“人机协同”的反馈链路，将用户纠错数据回传用于优化切片质量。

状态与凭证层：

DynamoDB存储对话状态，Secrets Manager安全存储飞书应用凭证。

2.2核心部署流程

我们利用AWS CDK实现了“一次定义，多地部署”，具体步骤如下：

环境初始化

飞书开放平台作为交互入口，通过事件订阅与机器人消息与AWS侧交互。

```
Bootstrapping environment aws://...
Trusted accounts for deployment: (none)
Trusted accounts for lookup: (none)
Using default execution policy of 'arn:aws:iam::aws:policy/AdministratorAccess'. Pass '--cloudformation-execution-policies' to customize.
CDKToolkit: creating CloudFormation changeset.
Environment aws://... bootstrapped.

NOTICES (What's this? https://github.com/aws/aws-cdk/wiki/CLI-Notices)
```

资源部署

执行 `cdk deploy`，自动创建API Gateway、Lambda、Bedrock知识库、DynamoDB表等所有资源。

```
Synthesis time: 10.92s
AwsFeishuAssistantCdkStack: start: Building IndexCreatorFn/Code
AwsFeishuAssistantCdkStack: success: Built IndexCreatorFn/Code
AwsFeishuAssistantCdkStack: start: Building SharedLayer/Code
AwsFeishuAssistantCdkStack: success: Built SharedLayer/Code
AwsFeishuAssistantCdkStack: start: Building McpLambda/Code
AwsFeishuAssistantCdkStack: success: Built McpLambda/Code
AwsFeishuAssistantCdkStack: start: Building AwsFeishuAssistantCdkStack Template
AwsFeishuAssistantCdkStack: success: Built AwsFeishuAssistantCdkStack Template
AwsFeishuAssistantCdkStack: start: Publishing IndexCreatorFn/Code (7a99d6c8)
AwsFeishuAssistantCdkStack: start: Publishing SharedLayer/Code (/1c40e71)
AwsFeishuAssistantCdkStack: start: Publishing AwsFeishuAssistantCdkStack Template (0101bafc)
AwsFeishuAssistantCdkStack: start: Publishing McpLambda/Code (a9f60202)
AwsFeishuAssistantCdkStack: success: Published AwsFeishuAssistantCdkStack Template (0101bafc)
AwsFeishuAssistantCdkStack: success: Published IndexCreatorFn/Code (7a99d6c8)
AwsFeishuAssistantCdkStack: success: Published McpLambda/Code (a9f60202)
AwsFeishuAssistantCdkStack: success: Published SharedLayer/Code (71c40e71)
Stack AwsFeishuAssistantCdkStack
IAM Statement Changes
```

Resource	Principal	Condition	Effect	Action
+	<code>\${AossCollection.Arn}</code>		Allow	<code>aoss:APIAccessAll</code>
	<code>AWS:\${KnowledgeBaseRole}</code>			

```
Synthesis time: 10.9s
AwsFeishuAssistantCdkStack: start: Building IndexCreatorFn/Code
AwsFeishuAssistantCdkStack: success: Built IndexCreatorFn/Code
AwsFeishuAssistantCdkStack: start: Building AwsFeishuAssistantCdkStack Template
AwsFeishuAssistantCdkStack: success: Built AwsFeishuAssistantCdkStack Template
AwsFeishuAssistantCdkStack: start: Publishing IndexCreatorFn/Code (172229444780-ap-northeast-2-7a99d6c8)
AwsFeishuAssistantCdkStack: start: Publishing AwsFeishuAssistantCdkStack Template (172229444780-ap-northeast-2-03e8b0e2)
AwsFeishuAssistantCdkStack: success: Published IndexCreatorFn/Code (172229444780-ap-northeast-2-7a99d6c8)
AwsFeishuAssistantCdkStack: success: Published AwsFeishuAssistantCdkStack Template (172229444780-ap-northeast-2-03e8b0e2)
Stack AwsFeishuAssistantCdkStack
IAM Statement Changes
```

Resource	Effect	Action	Principal	Condition
+	Allow	<code>aoss:APIAccessAll</code>	<code>AWS:\${IndexCreatorRole}</code>	
+	Allow	<code>sts:AssumeRole</code>	<code>Service:lambda.amazonaws.com</code>	

```
IAM Policy Changes
```

Resource	Managed Policy ARN
+	<code>arn:\${AWS::Partition}:iam::aws:policy/service-role/AWSLambdaBasicExecutionRole</code>

(NOTE: There may be security-related changes not in this list. See <https://github.com/aws/aws-cdk/issues/1299>)

"--require-approval" is enabled and stack includes security-sensitive updates: 'Do you wish to deploy these changes' (y/n)
y

配置输出

- o WebhookUrl: 飞书事件订阅地址。
- o SecretsArn: 用于存储飞书凭证的ARN。
- o KnowledgeBaseId / DataSourceId: 知识库及数据源ID。
- o KnowledgeBucketName: 用于上传知识文档的S3桶名称。

```
AwsFeishuAssistantCdkStack
Deployment time: 420.04s

Outputs:
AwsFeishuAssistantCdkStack.DataSourceId = 9ZAR7ZQRBZ
AwsFeishuAssistantCdkStack.FeishuWebhookApiEndpoint8BB4D767 = https://i[redacted].execute-api.[redacted].amazonaws.com/prod/
AwsFeishuAssistantCdkStack.KnowledgeBaseId = 44V[redacted]P
AwsFeishuAssistantCdkStack.KnowledgeBucketName = feishu-assistant-knowledge[redacted]
AwsFeishuAssistantCdkStack.SecretsArn = arn:aws:secretsmanager:[redacted]:secret:feishu-assistant/secrets-fdIIPs
AwsFeishuAssistantCdkStack.WebhookUrl = https://i[redacted].execute-api.[redacted].amazonaws.com/prod/webhook
Stack ARN:
arn:aws:cloudformation:[redacted]:stack/AwsFeishuAssistantCdkStack/ec7d4b80-281[redacted]:4e7081

Total time: 430.92s
```

数据同步

将企业知识文档上传至指定S3桶，触发Bedrock知识库数据源同步（Ingestion），完成知识库初始化。

应用配置

将飞书应用的App ID、Secret等凭证录入Secrets Manager，并在飞书开放平台配置事件回调地址为 WebhookUrl，完成最终对接。

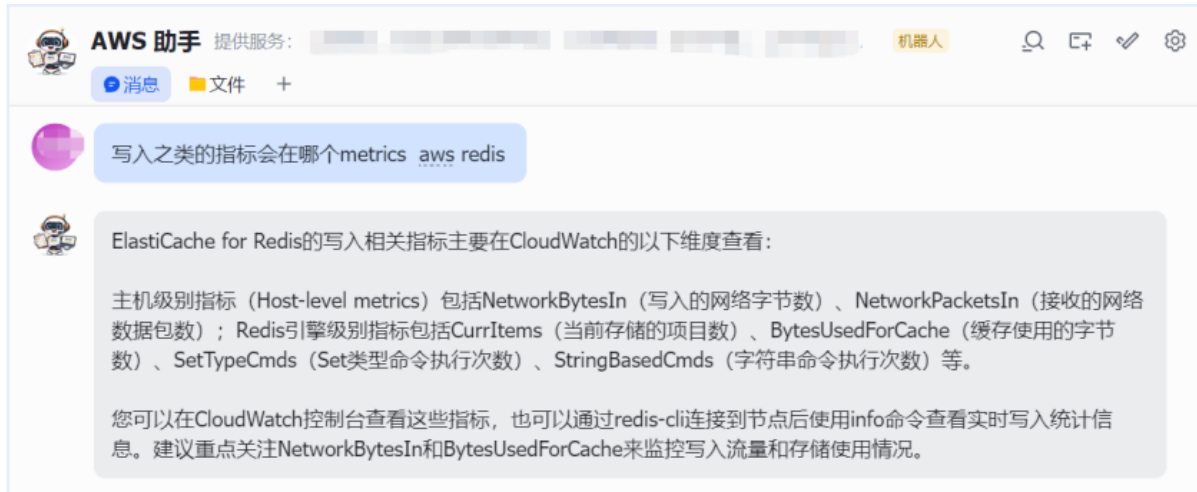
The screenshot displays the Feishu Open Platform console interface, divided into two main sections. The top section, titled '权限管理' (Permissions Management), lists various permissions for the application, all of which are marked as '已开通' (Enabled). The bottom section, titled '事件与回调' (Events and Callbacks), shows the configuration for sending events to a developer server. The '请求地址' (Request Address) field is populated with the WebhookUrl from the previous screenshot. Below this, a table lists the configured events.

事件名称	订阅类型	所需权限 (开通以下任一权限即可)	操作
接收消息 v2.0 im:message.receive_v1	应用身份	读取用户发给机器人的单聊消息 (已开通) 接收群聊中@机器人消息事件 (已开通)	删除事件

2.3使用效果展示

知识库检索效果

用户通过飞书提问，系统基于Bedrock知识库返回精准的企业内部知识，支持复杂的语义查询，显著提升信息获取效率。



AWS Knowledge Base MCP效果

我们深度集成了AWS Bedrock知识库的多模态能力 (MCP)，不仅支持文本，还能处理表格、流程图等复杂格式，实现跨文档的上下文连贯回答。



三、AWS资源清单与优势

易定云在选型时，严格遵循“无服务器优先”与“托管服务优先”的原则，以降低客户运维成本。

AWS服务	作用	选择理由
API Gateway	托管REST API入口，处理飞书回调	自动弹性伸缩，内置DDoS防护，免运维
Lambda	核心业务逻辑处理	按调用计费，毫秒级弹性伸缩，与CDK完美集成
Bedrock、Knowledge、Base	RAG核心能力	原生支持多种数据源，自动向量化，无需自建向量数据库
DynamoDB	存储用户会话上下文	高并发低延迟的NoSQL，适配Serverless架构
Secrets、Manager	安全存储API凭证	强制轮转策略，避免明文凭证泄露风险
S3	知识文档存储	高持久性，事件通知机制无缝触发数据同步

易定云见解：

在CDK部署过程中，我们预置了细粒度IAM权限，严格遵循最小权限原则，并为不同环境（开发、测试、生产）设计了独立的资源堆栈，有效避免了资源误操作与数据泄露风险。这是我们交付金融、政企客户的核心保障。

四、易定云的核心价值与差异化

作为AWS合作伙伴，易定云在此方案中不仅提供了技术实现，更注入了我们对企业级AI落地的深刻理解：

全生命周期管理

从文档上传、切片、向量化到检索生成，我们提供完整的监控与运维看板，帮助客户洞察知识库健康度与用户提问热点。

飞书深度集成优化

针对飞书卡片消息、交互式按钮等特性进行了定制开发，实现“一键反馈”、“答案溯源”等增强功能，提升用户交互体验。

成本可视与优化

基于AWS Budgets与自定义Dashboard，帮助客户精细化分析API调用、Bedrock token消耗、Lambda调用次数，实现成本可控的AI规模化应用。

安全合规

所有企业知识库数据存储于客户专属AWS账号，通过VPC私有链路与Secrets Manager，确保数据不离开客户控制平面，满足数据合规要求。

五、总结与展望

成都易定云科技有限公司基于AWS CDK与Bedrock构建的飞书智能助手，成功解决了企业内部知识检索难、AI集成门槛高、运维成本大的痛点。我们通过基础设施即代码、无服务器架构、安全优先的设计理念，为客户提供了**开箱即用、安全可控、弹性伸缩**的企业级AI解决方案。

未来

我们将继续探索Agent for Bedrock、多模态检索以及Fine-tuning等前沿技术进一步赋能企业在销售、研发、客服等场景下的智能化转型



FUTURE