



架构图

Server-Side Micro-Frontends 在 AWS 中渲染



Server-Side Micro-Frontends 在 AWS 中渲染: 架构图

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon 的商标和商业外观不得用于任何非 Amazon 的商品或服务，也不得以任何可能引起客户混淆、贬低或诋毁 Amazon 的方式使用。所有非 Amazon 拥有的其他商标均为各自所有者的财产，这些所有者可能附属于 Amazon、与 Amazon 有关联或由 Amazon 赞助，也可能不是如此。

Table of Contents

SSR Micro-Frontends i

Server-Side Micro-Frontends 在 AWS 中渲染 1

延伸阅读 2

图表历史 2

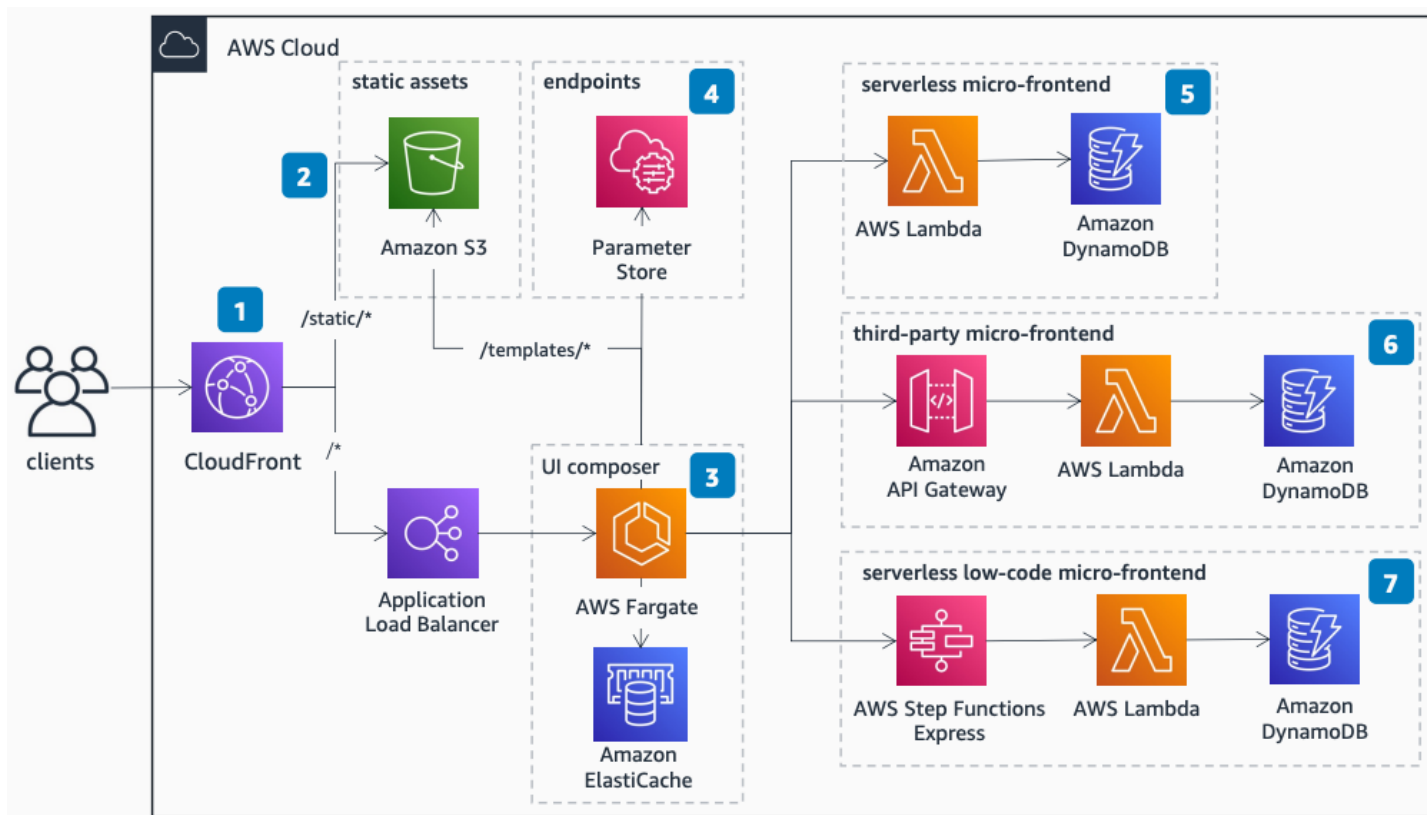
..... iii

Server-Side Micro-Frontends 在 AWS 中渲染

发布日期：2022年 9月9日 ([图表历史](#))

该架构展示了如何 AWS 使用无服务器方法实现服务器端渲染微前端。每个无服务器微前端都会返回一个 HTML 片段 (HTML-on-the-wire)，用户界面编辑器将这些独立的部分拼接在一起，为用户创造无缝体验。

Server-Side Micro-Frontends 在 AWS 中渲染



以下步骤描述了架构：

1. [亚马逊 CloudFront](#) 作为入口点有两个来源：[亚马逊简单存储服务存储桶](#)和公共应用程序负载均衡器。
2. Amazon S3 存储桶包含浏览器的所有静态文件，例如常见的微前端依赖关系、图像和 CSS 文件。它还包含用户界面编辑器用来将每个微前端放置在 HTML 页面上的模板。
3. 用户界面编辑器在 [AWS Fargate](#) 集群上运行，将不同的微前端拼接在一起，串流响应以提高性能。使用 [Amazon ElastiCache](#) 缓存微前端输出或整页以获得额外的性能提升。

4. 使用 [AWS 系统管理器](#) 参数存储收集所有微服务终端节点。它们可以是 HTTP 终端节点或亚马逊资源名称 (ARN)，解耦团队级别的依赖关系。
5. 无服务器微前端使用 [AWS Lambda](#) 和 [亚马逊 DynamoDB](#) 来存储数据并呈现 HTML 片段，准备嵌入到用户界面编辑器模板中。
6. 对于第三方微前端，使用 [Amazon API Gateway](#) 验证令牌或 API 密钥，确保只有您的应用程序可以访问该终端节点。
7. 使用 [AWS Step Functions](#) Express 作为生成微前端的低代码解决方案。Step Functions 与 200 多种服务集成，可减少计算，从本地从 DynamoDB 检索数据，并将渲染委托给 Lambda 函数。

延伸阅读

有关其他信息，请参阅以下资源：

- [AWS 架构图标](#)
- [AWS 建筑中心](#)
- [AWS Well-Architected](#)

图表历史

要获得有关此参考架构图更新的通知，请订阅 RSS 提要。

变更	说明	日期
初次发布	参考架构图首次发布。	2022 年 9 月 9 日

Note

要订阅 RSS 更新，必须为正在使用的浏览器启用 RSS 插件。

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。