



架构图

AWS DevOps 上的数据库



AWS DevOps 上的数据库: 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

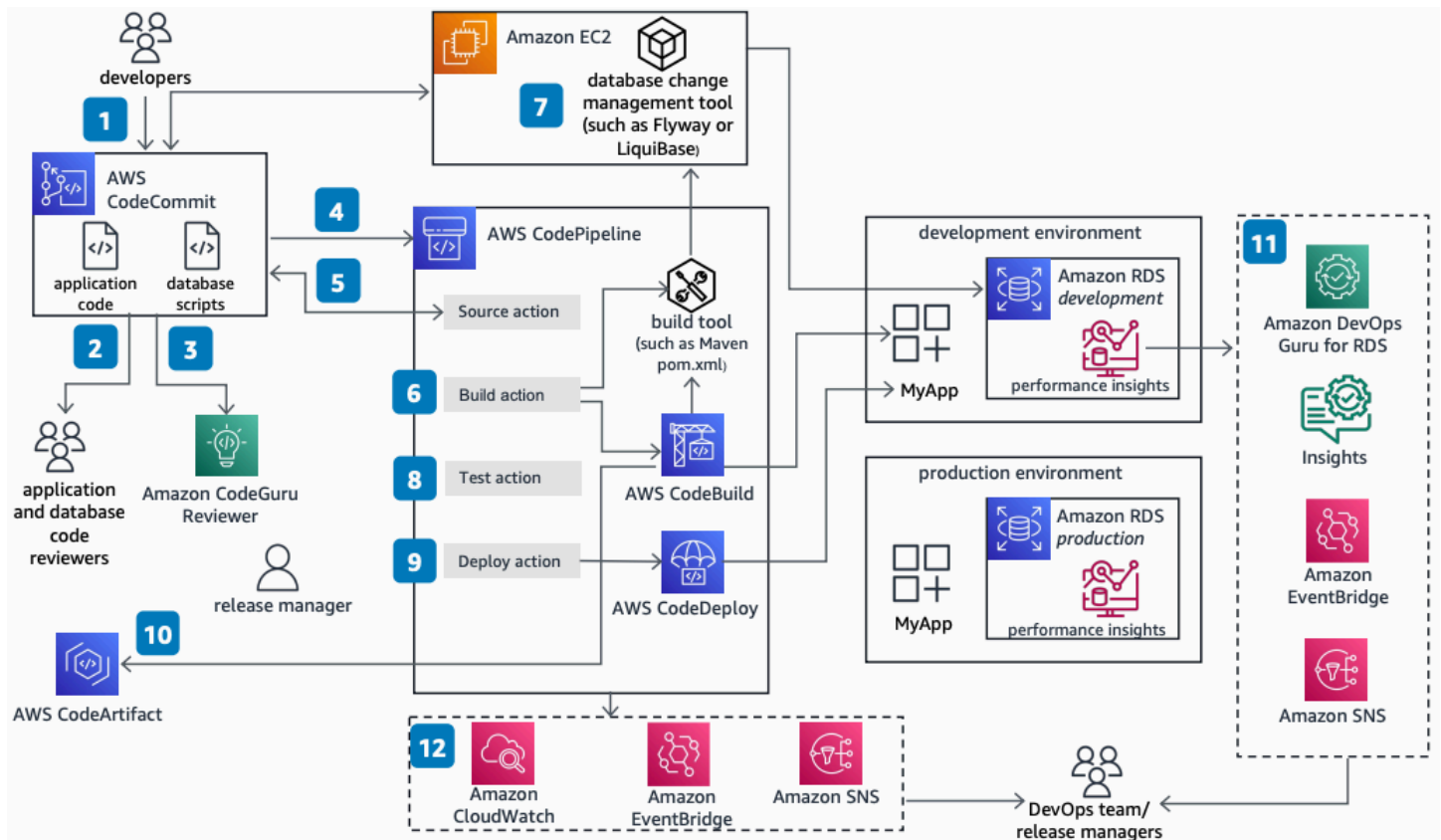
- 开发工作流程 1
 - AWS DevOps 上的数据库：开发工作流程 1
 - 延伸阅读 2
 - 图表历史 2
- 制作工作流程 3
 - AWS DevOps 上的数据库：生产工作流程 3
- v

AWS DevOps 上的数据库：开发工作流程

发布日期：2023 年 3 月 31 日 ([图表历史](#))

作为 DevOps 实践的一部分，在开发工作流程中使用此架构来自动进行数据库更改。建立编排，以与应用程序代码相同的速率部署数据库变更，强制执行数据库-应用程序-联合代码审查，整合数据库和应用程序代码操作，检测异常系统行为，并自动向指定团队发送通知。

AWS DevOps 上的数据库：开发工作流程



以下步骤描述了架构：

1. 开发人员将应用程序和数据库代码都签入 [AWS CodeCommit](#)。
2. 在每次推送请求时，都会使用批准规则 AWS CodeCommit 强制进行代码审查。
3. Amazon CodeGuruReviewer 会自动评估代码更改、检测错误并提供代码建议。
4. AWS CodeCommit 开始 [CodePipeline](#) 采取管道行动。
5. S CodePipeline ource 操作检查应用程序和数据库代码。

6. CodePipeline 生成操作调用生成管理工具来启动应用程序和数据库代码构建阶段。
7. 构建工具调用数据库更改工具。它下载数据库脚本并针对目标数据库运行它们。
8. 生成工具调用数据库测试脚本并验证输出。
9. “部署 CodePipeline 部署”操作将代码部署到目标环境。
10. 构建项目发布到 [AWS CodeArtifact](#) 供其他人使用。
11. (可选) 在 [Amazon RDS 上使用性能洞察](#) 并 Amazon DevOps Guru 自动应用机器学习技术来检测性能瓶颈和运营问题。
12. [EventBridge](#) 捕获事件并将其发送到 [Amazon SNS](#) 主题以接收用户通知。

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。	2023 年 3 月 31 日

Note

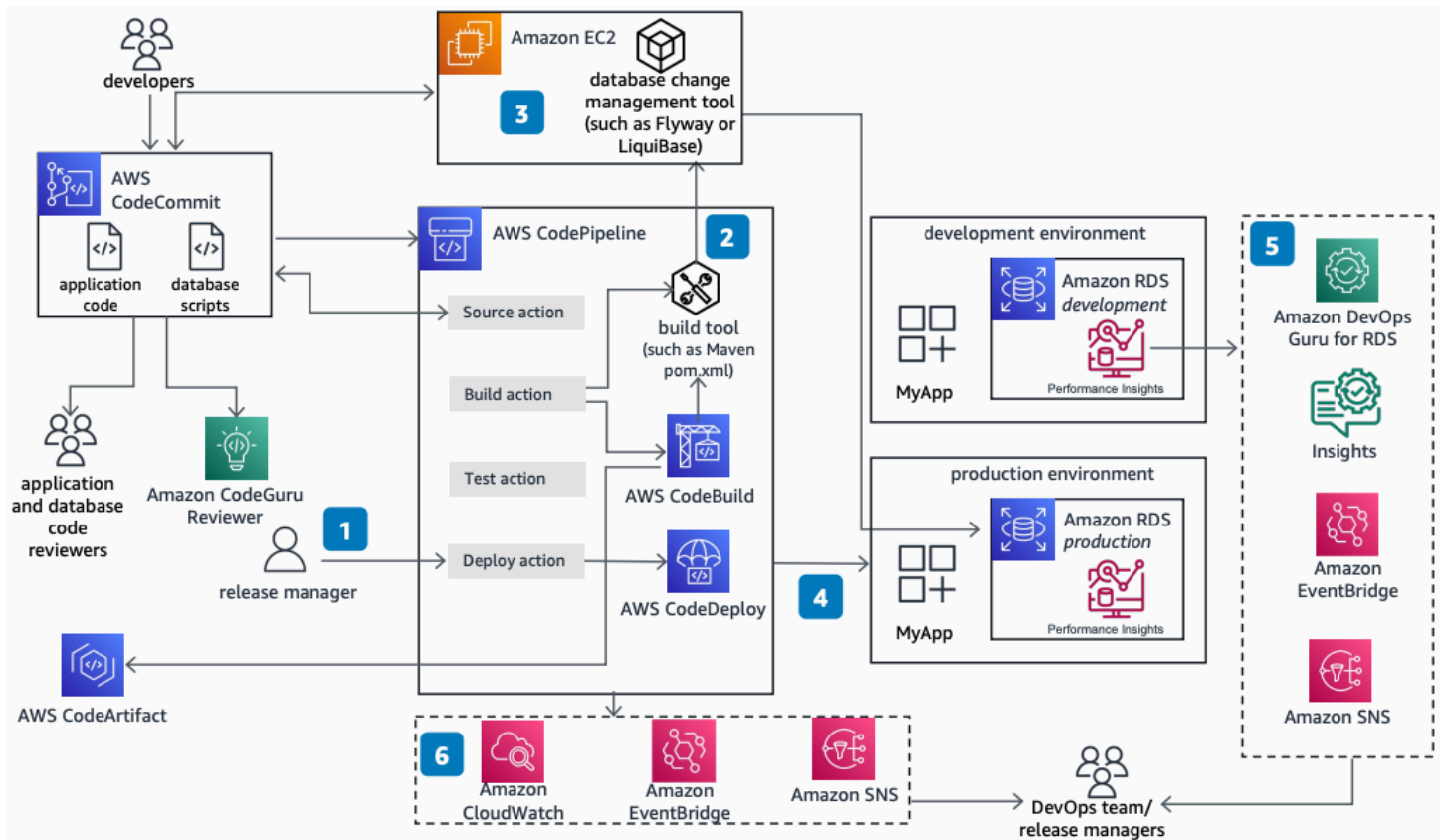
要订阅 RSS 更新，必须为正在使用的浏览器启用 RSS 插件。

AWS DevOps 上的数据库：生产工作流程

发布日期：2023 年 3 月 31 日 ([图表历史](#))

作为 DevOps 实践的一部分，在生产工作流程中使用此架构来自动进行数据库更改。建立编排，以与应用程序代码相同的速率部署数据库变更，检测异常的系统行为，并自动通知指定团队以采取纠正措施。

AWS DevOps 上的数据库：生产工作流程



以下步骤描述了架构：

1. 发布管理器发出生产部署请求。
2. [CodePipeline](#) 调用编译管理工具，该工具调用数据库变更管理工具。
3. 数据库更改工具下载数据库变更脚本并针对目标 [Amazon RDS](#) 生产数据库运行它们。
4. CodePipeline 启动“部署”操作以将代码部署到目标环境。
5. (可选) 在 Amazon RDS 上使用性能洞察并 Amazon DevOps Guru 自动应用机器学习技术来检测性能瓶颈和运营问题。

6. [EventBridge](#) 捕获事件并将其发送到 [Amazon SNS](#) 主题以接收用户通知。

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。