
aws
SUMMIT
ONLINE
JAPAN | MAY 11-12, 2021

DEMO ZONE

Amazon CodeGuru を使って AWS Lambda 用コードの品質を保つ

Shuhei Kimura

Solutions Architecture, Game Entertainment Solutions
Amazon Web Services Japan K.K.

aws

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

自己紹介

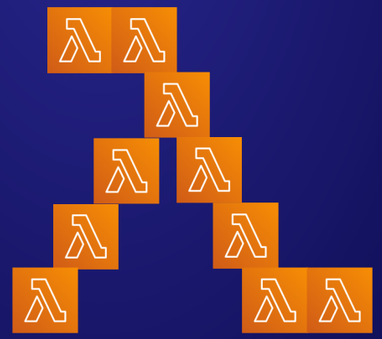


木村 秀平 (Shuhei Kimura)

ソリューションアーキテクト（SA）

好きなAWSサービス:

AWS Lambda



Amazon S3



© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

本セッションの流れ

- Amazon CodeGuru の紹介
- デモ
- 利用時の Tips

Lambda の紹介やベストプラクティス、git の利用方法などは省いて進めます

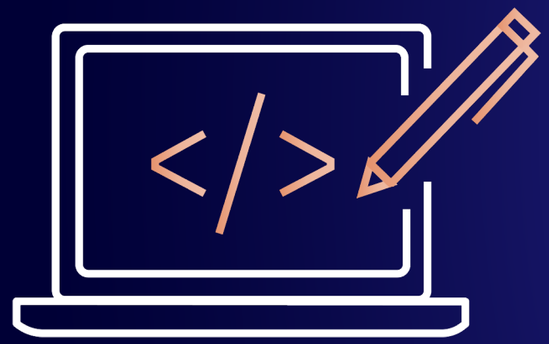
© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Amazon CodeGuru の紹介

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

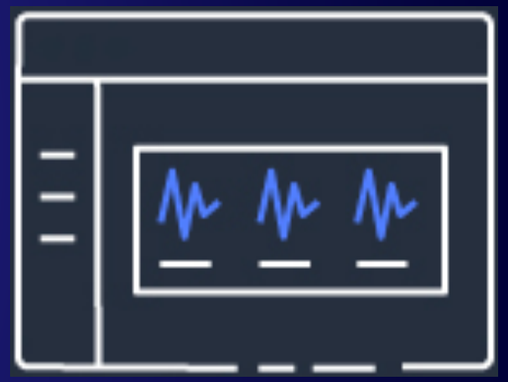
CodeGuru の機能

「ML を活用した推奨事項を使用して、
コードレビューを自動化し、アプリケーションのパフォーマンスを最適化」



Reviewer

静的解析



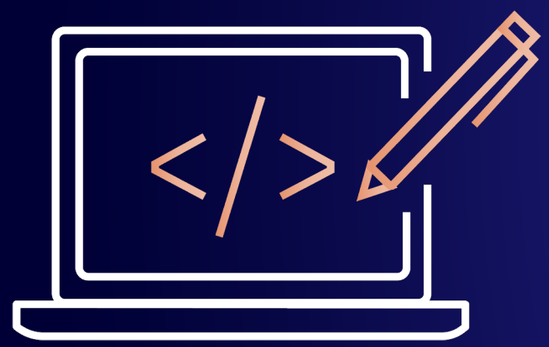
Profiler

動的解析

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

CodeGuru の機能

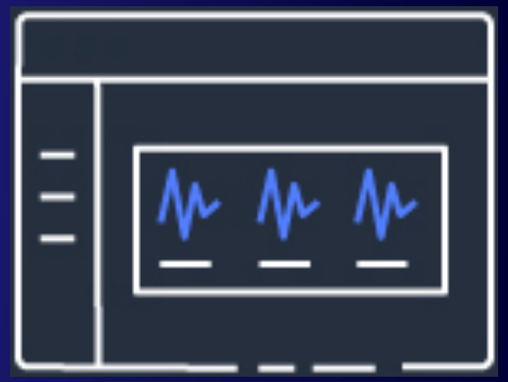
「ML を活用した推奨事項を使用して、
コードレビューを自動化し、アプリケーションのパフォーマンスを最適化」



Reviewer

静的解析

今回はこちらのみ紹介

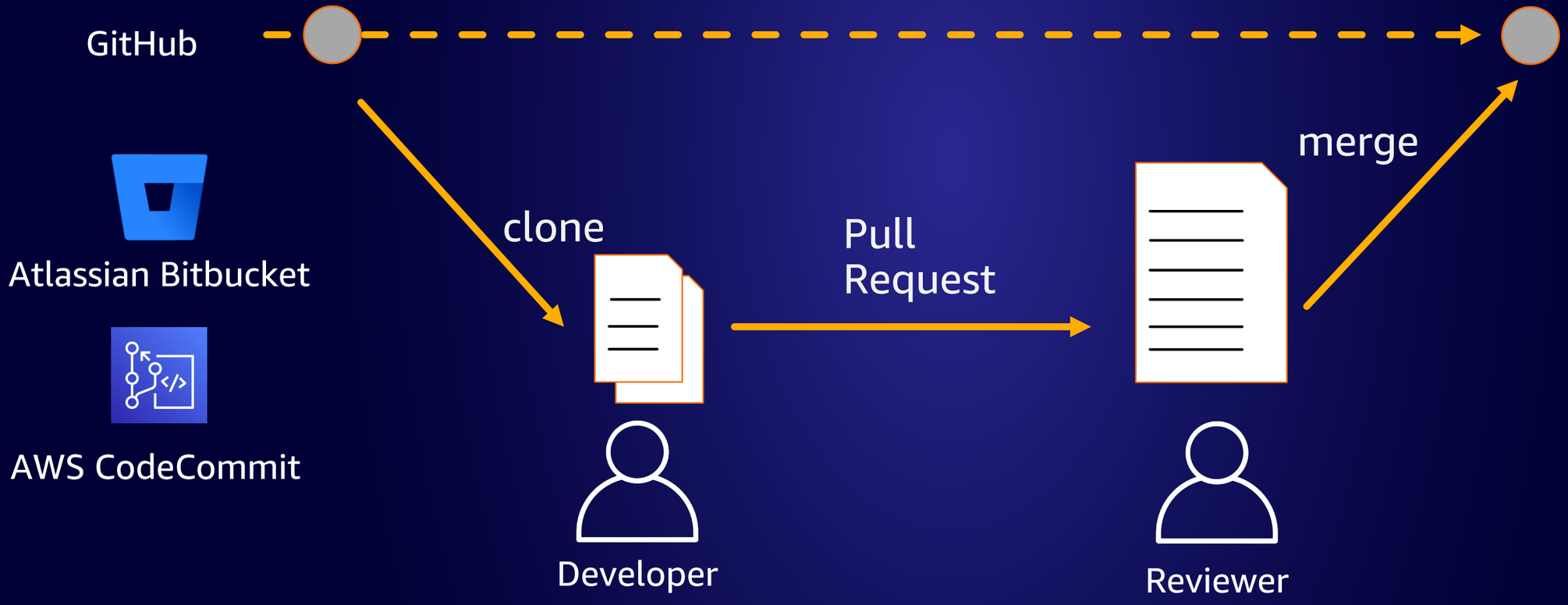


Profiler

動的解析

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

コードレビューの流れ



© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

（特に **Lambda** 用）コードレビューのよくある課題

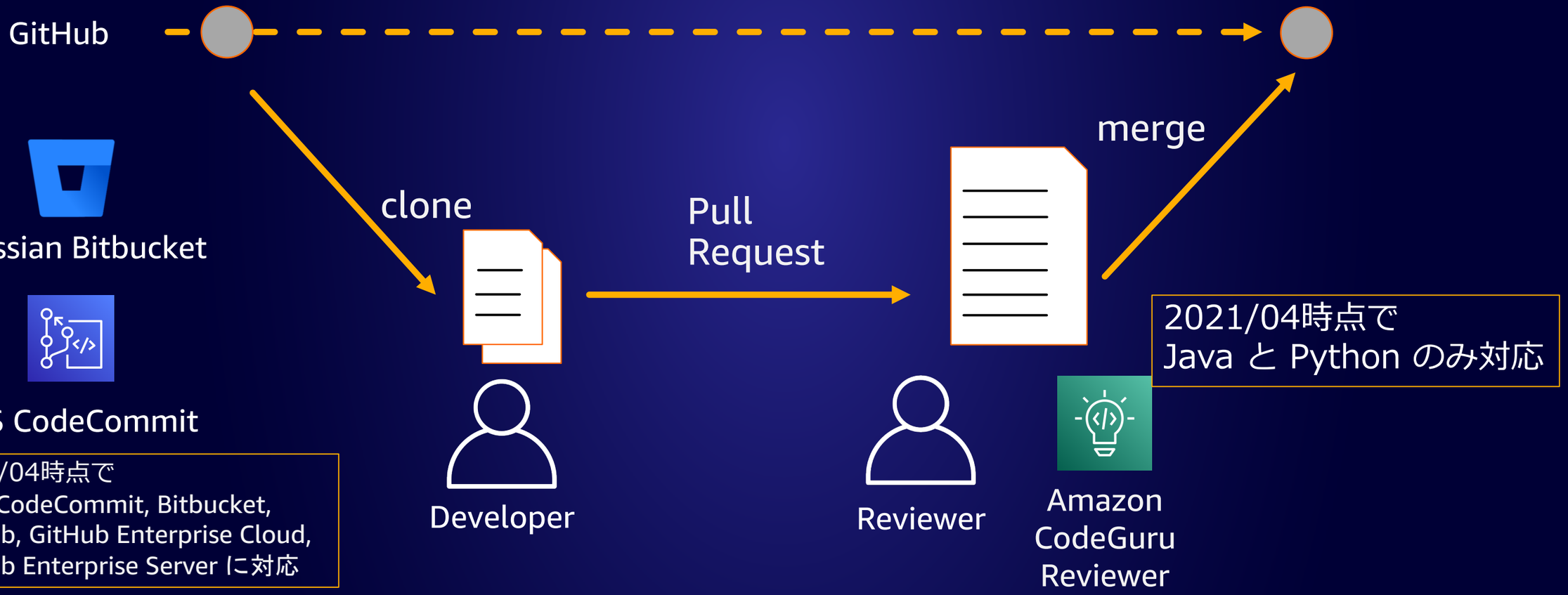
- レビュワーが足りない
- 初歩的なミスのチェックで疲弊し、本質的なチェックが行われない
- ドメインをまたがる知識が必要（e.g. アプリケーション一般 + AWS SDK）
- 規模が小さく、確認対象から抜けがち
- レビュワーが足りない

レビュワーの負荷を減らし、
継続的な利用を前提としたサービスが
CodeGuru

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

CodeGuru Reviewer の機能

リポジトリ内のコード、またはプルリクエスト中のコードを静的解析し結果を表示



© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

検出項目の例

- リソースリーク（e.g. closeし忘れ）
- データ競合
- AWS 利用のベストプラクティス
- 重複するコード
- コード品質
  - 長すぎる行数
  - サイクロマチック数（コードの複雑さを表す指標）
- など

デモで一部を見てゆきます

https://docs.aws.amazon.com/codeguru/latest/reviewer-ug/recommendations.html

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

デモ

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

利用時の **Tips**

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

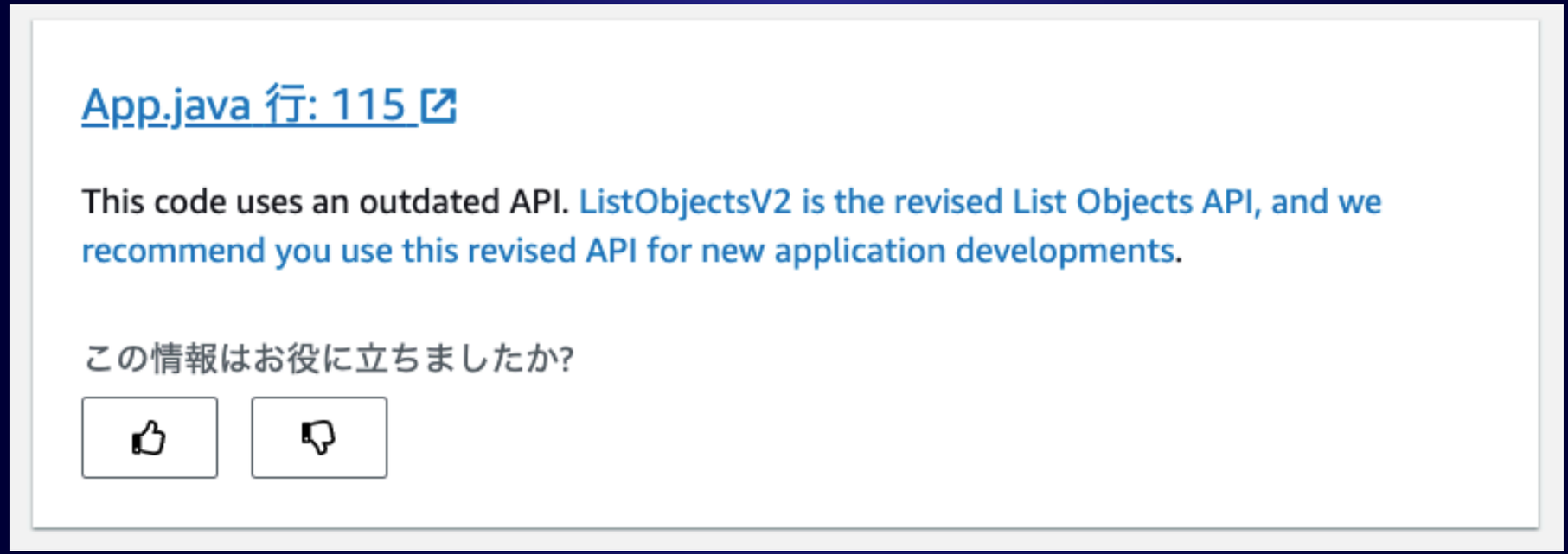
転ばぬ先の杖

- コードは劣化する
  - 今は問題なくても、将来的に問題になる可能性は十分ある
  - 定期的なブランチ全体に対する解析 + PRに対する解析
  - 価格の詳細は公式ドキュメントを参照
- 割れ窓を作らない
  - 指摘を1個見逃すと、気付いたら10個になっている
  - まずは小さなコード群に適用
  - 価値を感じたら、徐々に大きなコード群に適用、初回から0個を目指す

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

フィードバックを与える

- CodeGuru の指摘には、👍/ 👎 を付けられる
- 他の AWS サービス同様、ユーザーからのフィードバックを受けて改善されてゆく



https://docs.aws.amazon.com/codeguru/latest/reviewer-ug/give-feedback-from-code-review-details.html

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Thank you!



© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.