AWS SDK

AWS Black Belt Tech Webinar 2015 (旧マイスターシリーズ)

アマゾンデータサービスジャパン株式会社
ソリューションアーキテクト　西谷圭介
2015.03.18

自己紹介

- 名前
  - 西谷圭介
  - @Keisuke69 
- ロール
  - ソリューションアーキテクト
  - WebサービスやEC、スタートアップを担当
  - モバイルなどアプリ寄りなプロダクトを担当

AWS SDKの概要

と言えば

プログラマブル

プログラマブルとは？

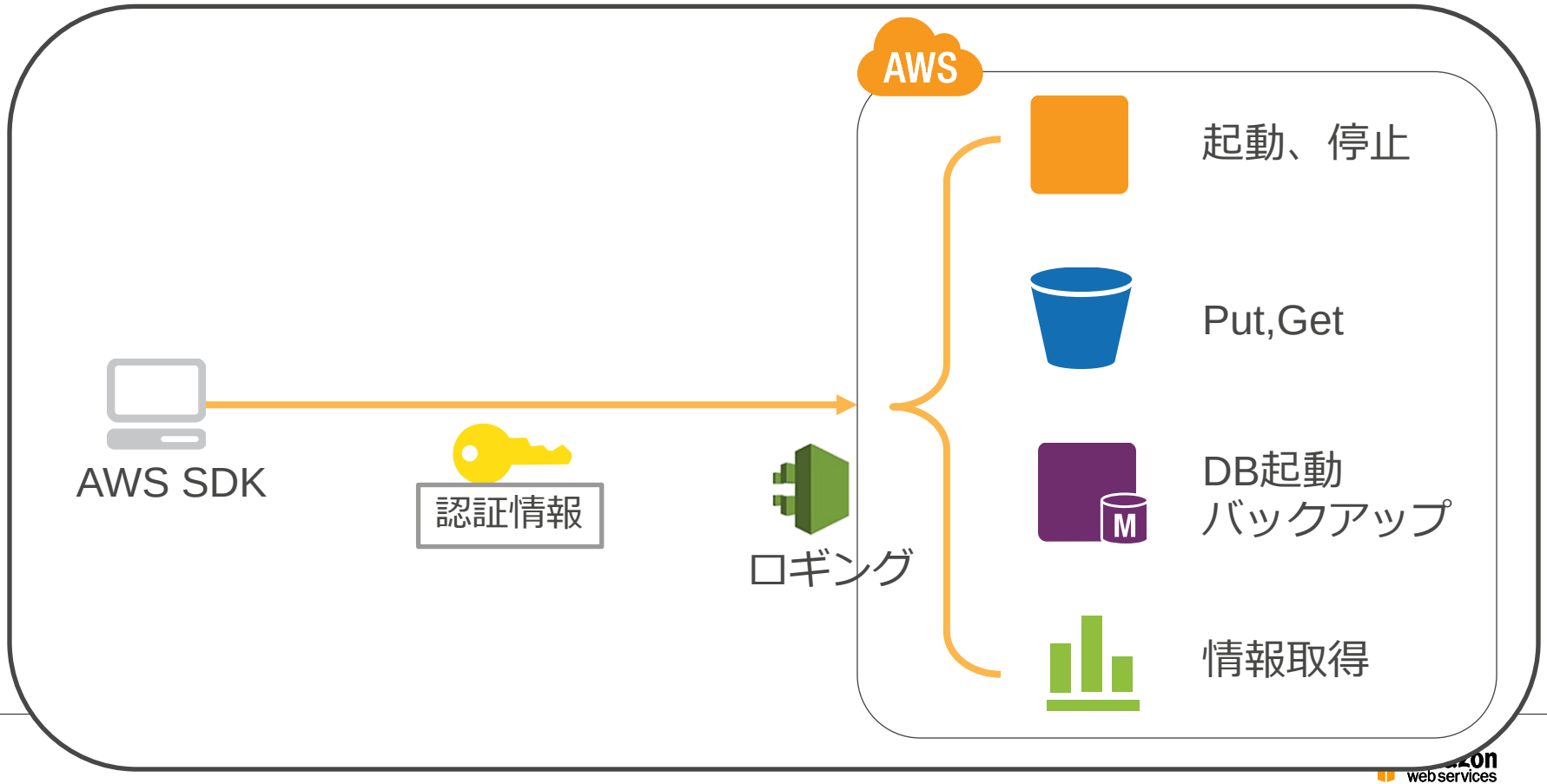
- クラウドはAPIでコントロールできるのが当たり前
- AWSの場合、ほぼ全てのサービスにAPIがある
- ネットワークからプラットフォーム自体まで、どのレイヤもプログラマのコンテキストでコントロールできる
- 各サービスの利用を抽象化したフレームワークを用意して透過的に利用するといったことも可能


つまり全てが意のままに

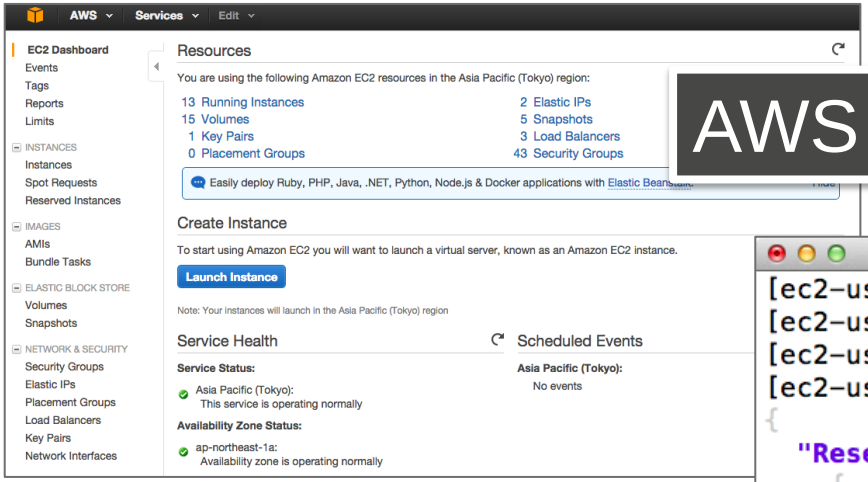
AWS SDK

- AWSのサービスをプログラムから操作できるSDK
- AWSの各サービスで提供されているAPIをwrapしたもの
- 通信はHTTP/HTTPS
  - 通信先のサービスごとのエンドポイントによる
  - http://docs.aws.amazon.com/general/latest/gr/rande.html
- 各種言語向けのSDKが用意されている

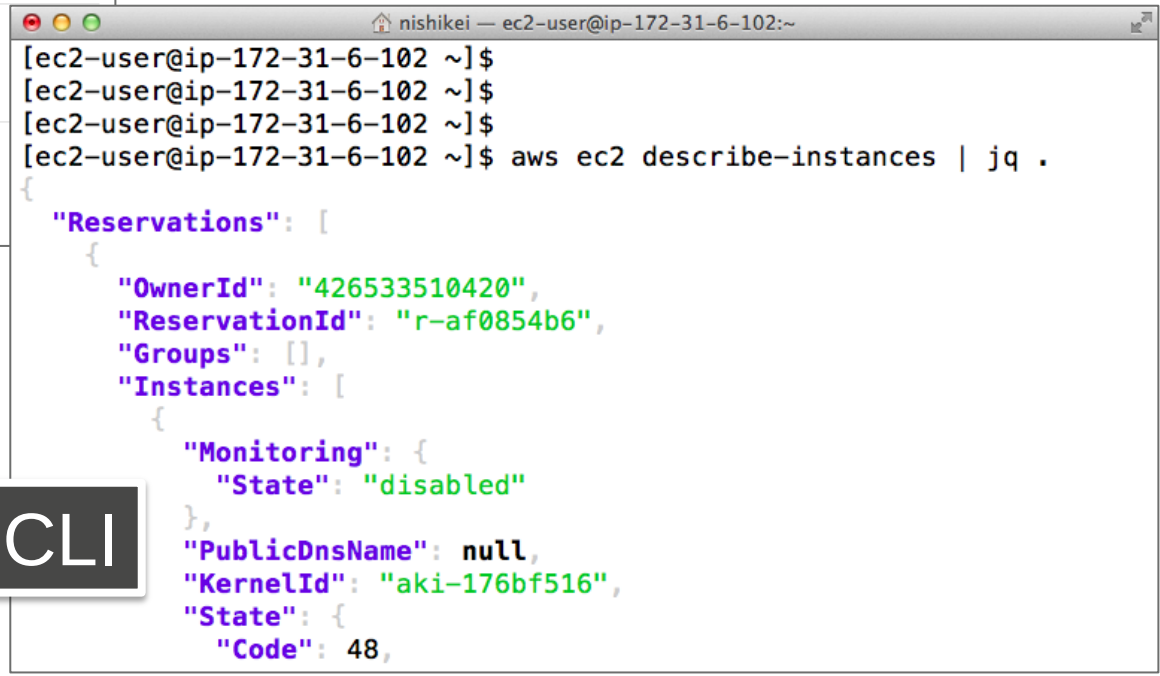
動作イメージ




ちなみに…



AWS Management Console



AWS CLI

これらの裏側では・・・

- 各サービスの各操作にAPIが定義されている
- AWS Management ConsoleやCLIもそれらを実行している
  - だから、AWS CloudTrailでManagement Console上の操作をAPIログとして記録することもできる

AWS SDK

開発者の環境（サーバやバッチ処理ワーカーなど）で動かすコードで利用


Java


NodeJS


.NET


PHP


Python


Ruby

エンドユーザの端末あるいはサービスのクライアント側で動くコードで利用


Android


iOS


Javascript
in
Browser

クライアント側SDK

AWS SDK

開発者の環境（サーバやバッチ処理ワーカーなど）で動かすコードで利用


Java


NodeJS


.NET


PHP


Python


Ruby

エンドユーザの端末あるいはサービスのクライアント側で動くコードで利用


Android


iOS

AWS Mobile SDK


Javascript in Browser

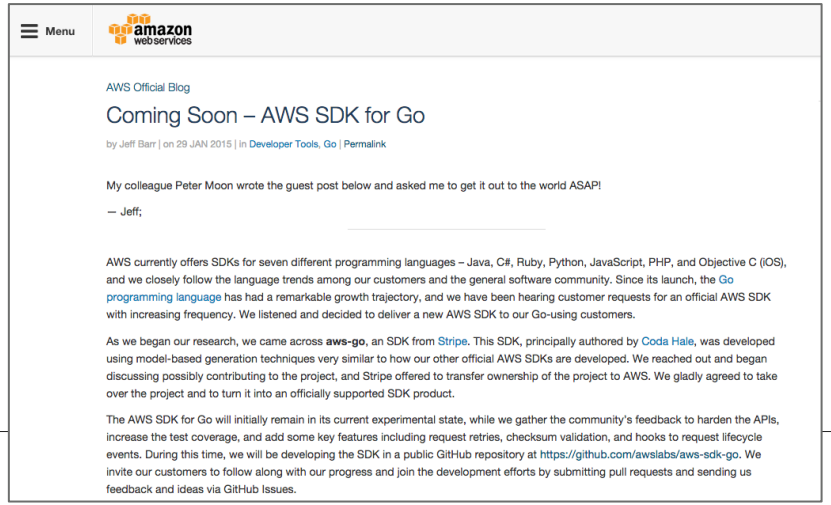
クライアント側SDK

AWS Mobile SDK

- モバイルアプリケーション用SDK
- 全てのサービスに共通の認証機構
- オンライン・オフラインを自動でハンドリング
- クロスプラットフォームのサポート
- Mobile OSへの最適化
  - 例： ローカルオフラインキャッシュを利用するアーキテクチャ
- メモリフットプリントの削減
  - 導入するパッケージをサービス単位で選択することが可能

AWS SDK for Go

- 元々、Stripe社で開発していたものをAWSが譲り受け公式SDKとして提供
  - 当面はDeveloper Previewとして提供
    https://github.com/awslabs/aws-sdk-go



Menu

amazon web services

AWS Official Blog

Coming Soon – AWS SDK for Go

by Jeff Barr | on 29 JAN 2015 | in Developer Tools, Go | Permalink

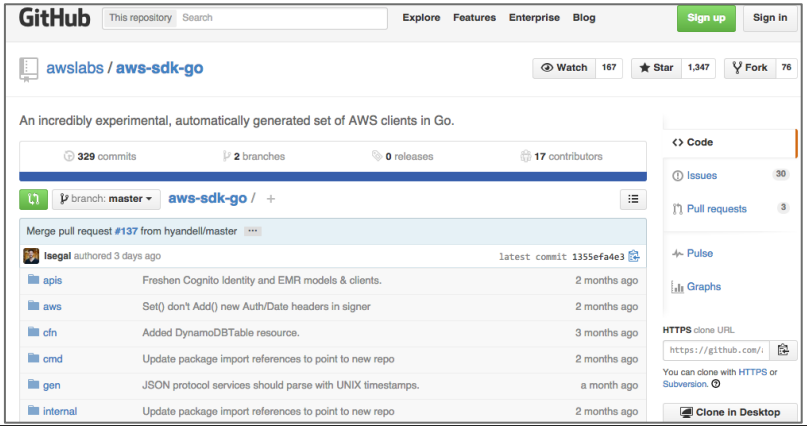
My colleague Peter Moon wrote the guest post below and asked me to get it out to the world ASAP!

— Jeff;

AWS currently offers SDKs for seven different programming languages – Java, C#, Ruby, Python, JavaScript, PHP, and Objective C (iOS), and we closely follow the language trends among our customers and the general software community. Since its launch, the Go programming language has had a remarkable growth trajectory, and we have been hearing customer requests for an official AWS SDK with increasing frequency. We listened and decided to deliver a new AWS SDK to our Go-using customers.

As we began our research, we came across **aws-go**, an SDK from Stripe. This SDK, principally authored by Coda Hale, was developed using model-based generation techniques very similar to how our other official AWS SDKs are developed. We reached out and began discussing possibly contributing to the project, and Stripe offered to transfer ownership of the project to AWS. We gladly agreed to take over the project and to turn it into an officially supported SDK product.

The AWS SDK for Go will initially remain in its current experimental state, while we gather the community's feedback to harden the APIs, increase the test coverage, and add some key features including request retries, checksum validation, and hooks to request lifecycle events. During this time, we will be developing the SDK in a public GitHub repository at https://github.com/awslabs/aws-sdk-go. We invite our customers to follow along with our progress and join the development efforts by submitting pull requests and sending us feedback and ideas via GitHub Issues.



GitHub | This repository | Search | Explore | Features | Enterprise | Blog | Sign up | Sign in

awslabs / aws-sdk-go | Watch 167 | Star 1,347 | Fork 76

An incredibly experimental, automatically generated set of AWS clients in Go.

329 commits | 2 branches | 0 releases | 17 contributors

branch: master | aws-sdk-go /

Merge pull request #137 from hyandell/master

lsegal authored 3 days ago | latest commit 1355efa4e3

apis	Freshen Cognito Identity and EMR models & clients.	2 months ago
aws	Set() don't Add() new Auth/Date headers in signer	2 months ago
cfn	Added DynamoDBTable resource.	3 months ago
cmd	Update package import references to point to new repo	2 months ago
gen	JSON protocol services should parse with UNIX timestamps.	a month ago
internal	Update package import references to point to new repo	2 months ago

Code | Issues 30 | Pull requests 3 | Pulse | Graphs

HTTPS clone URL

https://github.com/

You can clone with HTTPS or Subversion.

Clone in Desktop

amazon web services

AWS Mobile SDK for Unity

- クロスプラットフォームなゲーム開発環境として人気の高いUnityのPlugin
  - .NETベースのクラス群で構成
- 現在はDeveloper Previewでありサポートするサービスは限定的
  - Amazon Cognito
  - Amazon S3
  - Amazon DynamoDB
  - Amazon Mobile Analytics
- Unity4.0以降をサポート

AWS Mobile SDK for Xamarin

- クロスプラットフォームの開発環境であるXamarinのPlugin
  - 評価用のベータ版として提供中
  - https://github.com/awslabs/aws-sdk-xamarin
- サポートするサービス
  - Amazon Cognito
  - Amazon S3
  - Amazon DynamoDB
  - Amazon SNS Mobile Push

AWS SDKの用途

- AWSリソースのコントロール
  - インフラ構築/運用の自動化
  - EC2やRDSといったAWSリソースをプログラムから操作する
  - SDKによってサポートするサービスや操作が異なるため注意
- AWSサービスの利用
  - アプリケーション的なサービスを利用する場合に使う
  - アプリケーションの一部として組み込む
  - S3にデータを保存したり、DynamoDBやSQSへのデータ入出力など

SDKを使って利用するサービス（一例）

- Amazon Kinesis
  - 大量のストリームデータをリアルタイムに欠落なく処理
- Amazon SQS
  - フルマネージドなメッセージキューイングサービス
- AWS Lambda
  - 簡単にイベントドリブンアプリケーションの実装を実現
- Amazon SNS
  - プッシュ通知を含む各種メッセージングサービス
- Amazon DynamoDB
  - フルマネージドなKVS型NoSQLデータベース
- Amazon SES
  - 大量配信向け送信専用メール送信サービス

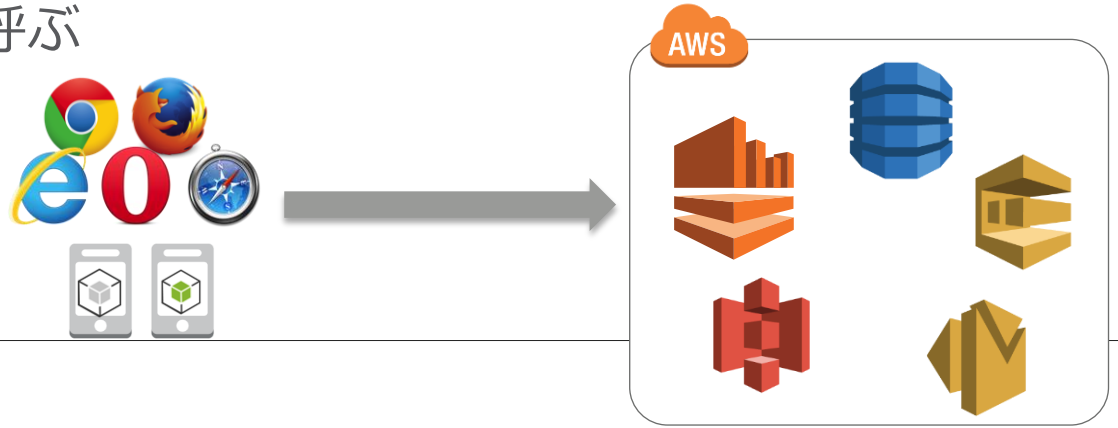
AWS SDKの基本的な使い方

- 必要な言語のSDKをインストール
  - 言語ごとにインストール方法が異なる
- Credential(AWS APIの認証情報)もしくはIAMロールを用意する
  - SDKから操作する必要最低限の権限に絞ったIAMユーザもしくはIAMロールを作成する
- サービス(例えばS3)のクライアントオブジェクトを生成
  - このときにCredentialを渡す
- クライアントオブジェクトのメソッドを使ってオペレーション(例えばPutObject)
- 言語によってはより高度に抽象化されているSDKもあります。

（参考）2-Tier Architecture

（参考）2-Tier Architecture

- Mobile SDKやAWS SDK for JavaScriptを利用することで、クライアントとバックエンドだけのアーキテクチャを実装可能
  - SDKとマネージドサービスを積極的に利用したサーバレス構成
  - モバイルアプリやブラウザ上のJavaScriptから直接AWSサービスを呼ぶ

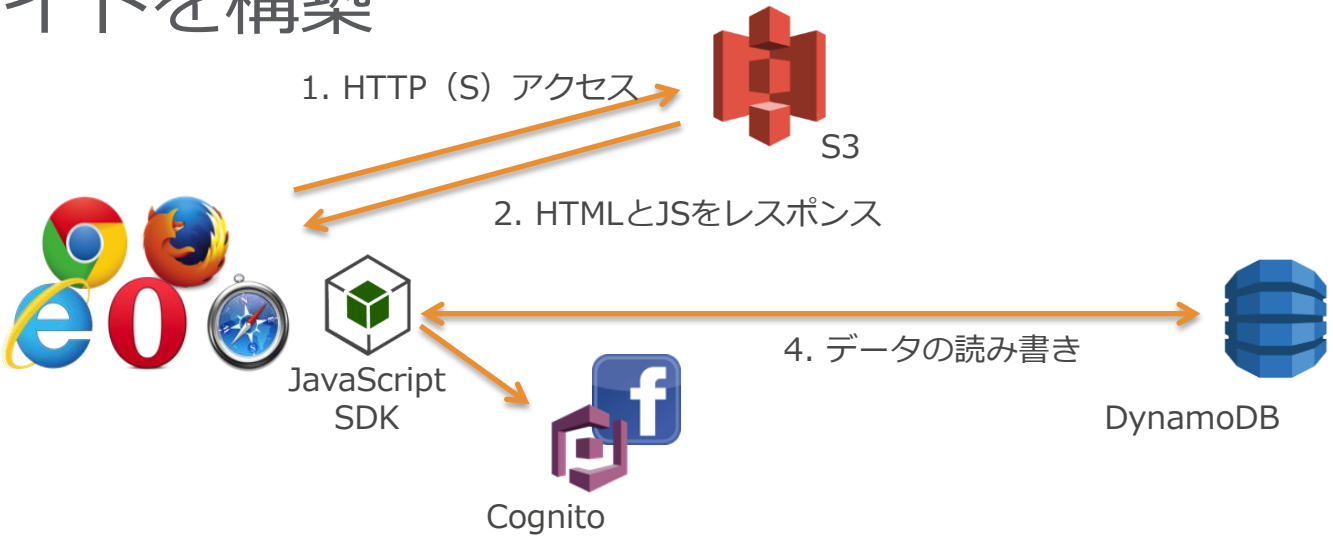


（参考）2-Tier Architectureのメリット

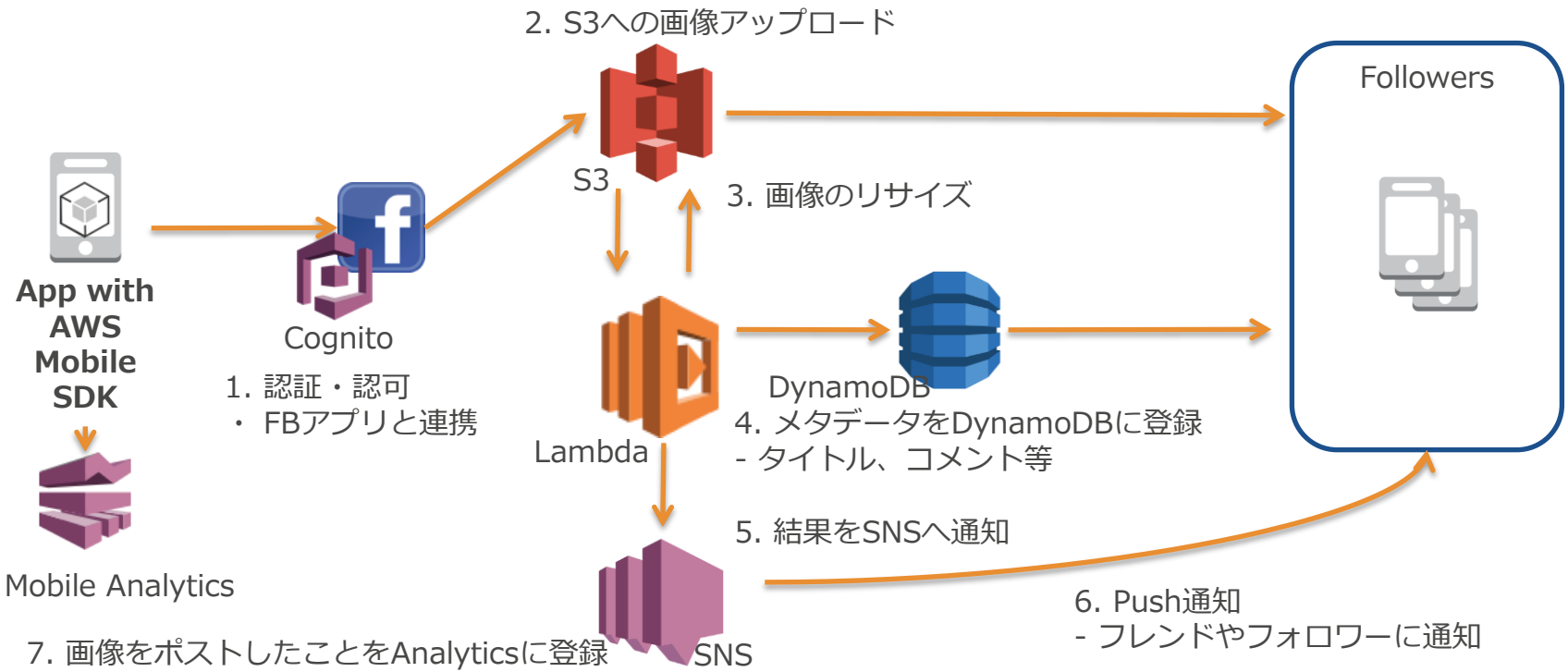
- アプリの開発に多くのメリット：
  - バックエンド側の開発コストを最小化
  - バックエンド側の運用コストを最小化
  - スケーラビリティの心配なし
  - バックエンドのEC2を減らせるため金額面でもローコスト（当社比*）
- 必要に応じてEC2も導入できる安心感
  - 後からバックエンド側にロジックを入れてシステムの最適化することも可能
- よりアプリ開発やビジネスにフォーカスできる

（参考）例1：静的ファイルだけで動的サイト

- DynamoDBやS3などをデータの保存先とするHTML+JSをS3に置いてWebサーバ無しで動的サイトを構築



（参考）例2： 写真共有モバイルアプリ



SDKを使う際の認証情報の扱い

よくある話

- 認証情報をプログラム内に埋め込んだ状態でGithubにpushして公開してしまう
  - JavaScriptで丸見えって話もたまに聞く
  - 管理者アカウントが漏れてEC2大量起動、大量請求なんていう怖い事例も
- モバイルアプリに認証情報を埋め込んでしまいにっちもさっちもいかなくなる

SDKを使う際の認証情報の扱い

- アプリに認証情報を埋め込むべきではない
  - アクセスキーが広範囲に配布されてしまう
  - アクセスキーの更新はアプリのアップデートを伴うため非現実的
- エンドユーザ/端末ごとに異なる認証情報を提供すべき
  - ユーザごとに必要最小限の権限を与えるのは重要
  - 不正利用発覚時に不正ユーザのみ権限を停止
- 認証情報は期限が来たら無効化されるべき
  - 不正ユーザの影響も期限付きに

SDKを使う際の認証情報の扱い

- プログラムを実行する場所によっていくつかのやり方がある
  - EC2上で動かす場合、IAMロールを使うのがオススメ
  - モバイルアプリの場合、Amazon Cognitoを利用するのがオススメ
- いずれにせよプログラム内に直接埋め込むパターンはダメ
  - セキュリティ上の問題
  - メンテナンス性の問題
- IAMの権限は必要最低限に絞ることを忘れない

SDKを使う際の認証情報の扱い

- 多くのSDKでは以下の場合、いずれも自動で読み込まれる
  - Shared credentialsファイルを用意
  - 環境変数としてAWS_ACCESS_KEY_IDとAWS_SECRET_ACCESS_KEYをセット
  - IAMロールを使う（AWS上で動かす場合のみ）
  - ただし、SDKによって見に行く順序等の細かい挙動の違いがあるので注意

コードサンプル（Ruby）

```
require 'aws-sdk'

s3 = Aws::S3::Client.new(region: 'ap-northeast-1')
resp = s3.list_buckets()
puts resp.buckets.map(&:name)
```

Shared credentialsファイル

- 認証情報を記述したファイルを事前に用意しておく

- 保存場所
  - ~/.aws/credentials（Linux/Macの場合）

記述例

```
[default]
aws_access_key_id = XXXXXXXXXXXXXXXXXXXX
aws_secret_access_key = XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

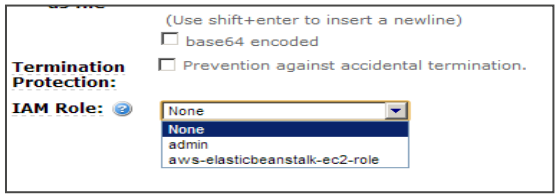
IAMロール

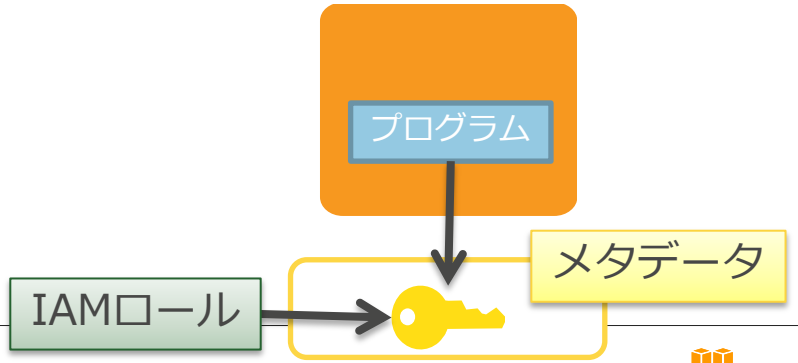
- AWSサービスやアプリケーション等、エンティティに対してAWS操作権限を付与するための仕組み
  - 例えば実行するアプリケーションにロールを付与する事で、そのアプリケーションからAWSを操作出来るようになる
- IAMユーザーやグループには紐付かない

- EC2ほか、Beanstalk,Data Pipelineなどでも利用

IAMロール

- EC2インスタンスに、指定のロールを付与する
  - EC2起動時にロールを指定すると、認証情報がメタデータに設定される
  - 認証情報はSTS(Security Token Service)で生成
    - インスタンス毎に異なるキー
    - 有効期限付きで、期限が来るとローテート
  - アプリケーションから認証情報を取得し、AWSサービスへアクセス
    - インスタンス内からメタデータにアクセス
    - アクセスキーID、シークレットアクセスキー、セッショントークンを取得
    - 3つの認証情報でAPI呼び出し





IAMユーザー利用との比較

- IAMロールを利用する事で、インスタンスと鍵管理を分離し、管理の簡素化とキーローテートによる、よりセキュアな運用が可能に

モバイルアプリの場合、Amazon Cognitoを使うことで

- 認証情報をアプリ内に埋め込む必要なし
  - IAMロールが割り当てられた一時的な認証情報をアプリで「簡単に」取得可能
- AWSの各種リソースへのアクセスをきめ細やかに設定可能
  - 細かいアクセス権の設定はIAM Policyを用いて行う
- セキュリティのベストプラクティスの実装が容易
  - 従来、STSとTVMを用いて行っていた面倒な実装が簡単にサーバーレスで行える

コードサンプル（Android）

```
CognitoCachingCredentialsProvider provider = new CognitoCachingCredentialsProvider(
    myActivity.getContext(),
    "IDENTITY_POOL_ID", // Identity pool ID
    Regions.US_EAST_1
);

s3 = new AmazonS3Client(provider);
List<Bucket> buckets = s3.listBuckets;
```

各SDKの概要

AWS SDK for Java

AWS SDK for Java

- Amazon提供のAWS開発用のJava向けSDK
  - http://aws.amazon.com/jp/sdk-for-java/
  - https://github.com/aws/aws-sdk-java
- APIリファレンス
  - http://docs.aws.amazon.com/AWSJavaSDK/latest/javadoc/index.html
- 環境：Java6以降
  - 64bitのJVMを推奨
- 特徴
  - Amazon S3のクライアント側のデータ暗号化のサポート
  - Amazon DynamoDB Object Mapper
  - Amazon S3 Transfer Manager
  - Amazon SQSのクライアント側のバッファリング

操作可能サービス

Auto Scaling	EC2	OpsWorks
CloudFormation	ECS	Redshift
CloudFront	ELB	RDS
CloudHSM	Elastic Beanstalk	Route53
CloudSearch	Elasticache	S3
CloudTrail	Elastic Transcoder	SES
CloudWatch	EMR	SimpleDB
CloudWatch Logs	Glacier	SNS
CodeDeploy	IAM	SQS
Cognito	Import/Export	Storage Gateway
Config	Kinesis	STS
Data Pipeline	KMS	Support
Direct Connect	Lambda	SWF
DynamoDB	Mobile Analytics	VPC

※薄字のサービスは未サポート

amazon web services

インストール方法

- 1.9.0以降Mavenを利用したコンポーネント単位でのインストールが可能

例：S3とDynamoDBしか利用しない場合

```
<dependencies>
    <dependency>
        <groupId>com.amazonaws</groupId>
        <artifactId>aws-java-sdk-s3</artifactId>
        <version>1.9.0</version>
    </dependency>
    <dependency>
        <groupId>com.amazonaws</groupId>
        <artifactId>aws-java-sdk-dynamodb</artifactId>
        <version>1.9.0</version>
    </dependency>
</dependencies>
```

AWS Toolkit for Eclipse

- EclipseにAWS SDK for Javaを使ったプロジェクトを追加するプラグイン
- AWSを使用したアプリの開発/テストを効率化
- EC2やS3などのサービス管理コンソールも付属
- Elastic Beanstalkへのデプロイも可能

AWS SDK for .NET 

AWS SDK for .NET

- Amazon提供のAWS開発用.NET SDK
  - http://aws.amazon.com/sdkfornet/
  - https://github.com/amazonwebservices/aws-sdk-for-net
- APIリファレンス
  - http://docs.aws.amazon.com/sdkfornet/latest/apidocs/Index.html
- 環境：
  - .NET Framework 3.5以降
  - Visual Studio 2010以降
- C#およびVisual Basicをサポート

AWS SDK for .NET

- Windowsストア および Windows Phone アプリのサポート
  - Windows サーバー、デスクトップ、タブレット、電話の環境をサポートするライブラリが含まれる
- Amazon DynamoDB オブジェクト永続フレームワーク
- Amazon DynamoDB Session State Provider
  - ASP.NET セッション状態を DynamoDB に簡単に格納
- Amazon S3 TransferUtility
- Amazon S3 クライアント側暗号化
- Amazon Glacier ArchiveTransferManager
  - 大きなファイルを自動的にパーツに分割し、チェックサムを計算

操作可能サービス

Auto Scaling	EC2	OpsWorks
CloudFormation	ECS	Redshift
CloudFront	ELB	RDS
CloudHSM	Elastic Beanstalk	Route53
CloudSearch	Elasticache	S3
CloudTrail	Elastic Transcoder	SES
CloudWatch	EMR	SimpleDB
CloudWatch Logs	Glacier	SNS
CodeDeploy	IAM	SQS
Cognito	Import/Export	Storage Gateway
Config	Kinesis	STS
Data Pipeline	KMS	Support
Direct Connect	Lambda	SWF
DynamoDB	Mobile Analytics	VPC

※薄字のサービスは未サポート

amazon web services

AWS SDK for .NETに含まれるもの

- AWS Toolkit for Microsoft Visual Studio
- Visual Studioプロジェクトテンプレート
- AWS Tools for Windows PowerShell
- AWS .NETライブラリ
- C#コードサンプル
- ドキュメント

インストール方法

- 以下のページの右上隅にある「AWS .NET for SDK」ボタンをクリック
  - http://aws.amazon.com/jp/sdkfornet/
- ファイルを保存するかどうかをたずねるメッセージがブラウザに表示されたら、ローカルのディスクに保存
- 保存したインストーラを開いてインストールプロセスを開始

AWS Toolkit for Visual Studio

- Microsoft Visual Studioを使用してのサービスの管理が可能
- AWS SDK for .NETによるアプリケーション開発に対応
- AWS Elastic Beanstalk/AWS CloudFormationによる.NETアプリケーションのデプロイに対応

AWS SDK for PHP 

AWS SDK for PHP

- Amazon提供のAWS開発用のPHP向けSDK
  - http://aws.amazon.com/jp/sdkforphp/
  - https://github.com/aws/aws-sdk-php
- APIリファレンス
  - http://docs.aws.amazon.com/aws-sdk-php/latest/
- 環境：PHP5.3.3以降
  - OpenSSLとZlibを有効にしたcURL extention
- 特徴
  - Amazon DynamoDB Session Handler
    - アプリケーションセッション状態を DynamoDB に簡単に格納
  - Amazon S3 and Glacier Multipart Uploader
  - Resource Convenience Helper
    - コードを排除し、Iterator、Waiter、および Batch ヘルパーによりロジックを簡素化

操作可能サービス

Auto Scaling	EC2	OpsWorks
CloudFormation	ECS	Redshift
CloudFront	ELB	RDS
CloudHSM	Elastic Beanstalk	Route53
CloudSearch	Elasticache	S3
CloudTrail	Elastic Transcoder	SES
CloudWatch	EMR	SimpleDB
CloudWatch Logs	Glacier	SNS
CodeDeploy	IAM	SQS
Cognito	Import/Export	Storage Gateway
Config	Kinesis	STS
Data Pipeline	KMS	Support
Direct Connect	Lambda	SWF
DynamoDB	Mobile Analytics	VPC

※薄字のサービスは未サポート

amazon web services

利用方法

- Composerによるインストール（推奨）
  - http://docs.aws.amazon.com/aws-sdk-php/guide/latest/installation.html#installing-via-composer
- Pharよるインストール
  - http://docs.aws.amazon.com/aws-sdk-php/guide/latest/installation.html#installing-via-phar
- Zipファイルからインストール
  - http://docs.aws.amazon.com/aws-sdk-php/guide/latest/installation.html#installing-via-zip

AWS SDK for Python(boto)

AWS SDK for Python(boto)

- AWS開発用のPython向けSDK
  - Botoとしても知られており、AWS CLIでもboto-coreを利用
  - http://aws.amazon.com/jp/sdk-for-python/
  - https://github.com/boto/boto
- APIリファレンス
  - http://docs.pythonboto.org/en/latest/ref/
- 環境：Python 2.6、2.7、3.3、3.4
- 現在Version3がDeveloper Preview
  - Python 2と3をネイティブサポート
  - 一貫性のあるサービスインターフェース
  - カスタマイズ可能なプラグイン

操作可能サービス

Auto Scaling	EC2	OpsWorks
CloudFormation	ECS	Redshift
CloudFront	ELB	RDS
CloudHSM	Elastic Beanstalk	Route53
CloudSearch	Elasticache	S3
CloudTrail	Elastic Transcoder	SES
CloudWatch	EMR	SimpleDB
CloudWatch Logs	Glacier	SNS
CodeDeploy	IAM	SQS
Cognito	Import/Export	Storage Gateway
Config	Kinesis	STS
Data Pipeline	KMS	Support
Direct Connect	Lambda	SWF
DynamoDB	Mobile Analytics	VPC

※薄字のサービスは未サポート

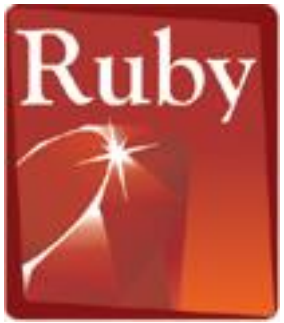
インストール

```
$ pip install boto
```

AWS SDK for Ruby

AWS SDK for Ruby

- AWS SDKのRuby版
  - V2が正式リリースされ大幅に改善
  - http://aws.amazon.com/jp/sdkforruby/
  - https://github.com/aws/aws-sdk-ruby
- APIリファレンス
  - http://docs.aws.amazon.com/sdkforruby/api/index.html
- 特徴
  - レスポンスのスタブ（テスト用）
  - レスポンスのページング
  - パラメータのバリデーション
  - 構造化されたレスポンスデータ
  - プラグインによる拡張性



操作可能サービス

Auto Scaling	EC2	OpsWorks
CloudFormation	ECS	Redshift
CloudFront	ELB	RDS
CloudHSM	Elastic Beanstalk	Route53
CloudSearch	Elasticache	S3
CloudTrail	Elastic Transcoder	SES
CloudWatch	EMR	SimpleDB
CloudWatch Logs	Glacier	SNS
CodeDeploy	IAM	SQS
Cognito	Import/Export	Storage Gateway
Config	Kinesis	STS
Data Pipeline	KMS	Support
Direct Connect	Lambda	SWF
DynamoDB	Mobile Analytics	VPC

※薄字のサービスは未サポート

amazon web services

インストール

```
$ gem install aws-sdk
```

Gemfileで指定する場合

```
gem 'aws-sdk', '~> 2'
```

AWS SDK for JavaScript in Node.js

AWS SDK for JavaScript in Node.js

- Amazon提供のAWS開発用のnode.js向けSDK
  - http://aws.amazon.com/sdkfornodejs/
  - https://github.com/aws/aws-sdk-js
- APIリファレンス
  - http://docs.aws.amazon.com/AWSJavaScriptSDK/latest/frames.html

操作可能サービス

Auto Scaling	EC2	OpsWorks
CloudFormation	ECS	Redshift
CloudFront	ELB	RDS
CloudHSM	Elastic Beanstalk	Route53
CloudSearch	Elasticache	S3
CloudTrail	Elastic Transcoder	SES
CloudWatch	EMR	SimpleDB
CloudWatch Logs	Glacier	SNS
CodeDeploy	IAM	SQS
Cognito	Import/Export	Storage Gateway
Config	Kinesis	STS
Data Pipeline	KMS	Support
Direct Connect	Lambda	SWF
DynamoDB	Mobile Analytics	VPC

※薄字のサービスは未サポート

amazon web services

インストール

```
$ npm install aws-sdk
```

AWS Mobile SDK for iOS

AWS Mobile SDK for iOS

- Amazon提供のモバイルアプリ開発用のiOS向けSDK
  - http://aws.amazon.com/mobile/sdk/
  - https://github.com/aws/aws-sdk-ios-v2
- APIリファレンス
  - http://docs.aws.amazon.com/AWSiOSSDK/latest/
- モバイルに最適化された高レベルインターフェースを持つクライアントライブラリを同梱
  - Amazon DynamoDB
  - Amazon S3
  - Amazon Kinesis
- Amazon CognitoとAmazon Mobile Analyticsのクライアントも同梱
- Bolts frameworkを利用した非同期処理

操作可能サービス

Auto Scaling	EC2	OpsWorks
CloudFormation	ECS	Redshift
CloudFront	ELB	RDS
CloudHSM	Elastic Beanstalk	Route53
CloudSearch	Elasticache	S3
CloudTrail	Elastic Transcoder	SES
CloudWatch	EMR	SimpleDB
CloudWatch Logs	Glacier	SNS
CodeDeploy	IAM	SQS
Cognito	Import/Export	Storage Gateway
Config	Kinesis	STS
Data Pipeline	KMS	Support
Direct Connect	Lambda	SWF
DynamoDB	Mobile Analytics	VPC

※薄字のサービスは未サポート

インストール

- CocoaPodsを利用してインストールする
  - http://cocoapods.org/
- Xcode上のプロジェクトフォルダにてpodfileを作成

```
source 'https://github.com/CocoaPods/Specs.git'
pod 'AWSiOSSDKv2'
```

- ターミナル上でプロジェクトフォルダに移動した上で以下を実行

```
$ pod install
```

AWS Mobile SDK for Android 

AWS Mobile SDK for Android

- Amazon提供のモバイルアプリ開発用のAndroid向けSDK
  - http://aws.amazon.com/mobile/sdk/
  - https://github.com/aws/aws-sdk-android/
- APIリファレンス
  - http://docs.aws.amazon.com/AWSAndroidSDK/latest/javadoc/
- iOS版同様、モバイルに最適化されたクライアントライブラリを同梱
  - Amazon DynamoDB
  - Amazon S3
  - Amazon Kinesis
- Amazon CognitoとAmazon Mobile Analyticsのクライアントも同梱
- 手動でダウンロードして配置するだけでなく、Mavenを利用したインストールが可能

操作可能サービス

Auto Scaling	EC2	OpsWorks
CloudFormation	ECS	Redshift
CloudFront	ELB	RDS
CloudHSM	Elastic Beanstalk	Route53
CloudSearch	Elasticache	S3
CloudTrail	Elastic Transcoder	SES
CloudWatch	EMR	SimpleDB
CloudWatch Logs	Glacier	SNS
CodeDeploy	IAM	SQS
Cognito	Import/Export	Storage Gateway
Config	Kinesis	STS
Data Pipeline	KMS	Support
Direct Connect	Lambda	SWF
DynamoDB	Mobile Analytics	VPC

※薄字のサービスは未サポート

インストール

- Mavenを利用したインストールが可能
  - コンポーネントごとにインストール可能

```
<dependencies>
    <dependency>
        <groupid>com.amazonaws</groupid>
        <artifactid>aws-android-sdk-core</artifactid>
        <version>2.1.3</version>
    </dependency>
    <dependency>
        <groupid>com.amazonaws</groupid>
        <artifactid>aws-android-sdk-s3</artifactid>
        <version>2.1.3</version>
    </dependency>
</dependencies>
```

AWS SDK for JavaScript in the Browser

AWS SDK for JavaScript in the Browser

- Amazon提供のブラウザ上で実行するJavaScript向けSDK
  - http://aws.amazon.com/sdk-for-browser/
  - https://github.com/aws/aws-sdk-js
- APIリファレンス
  - http://docs.aws.amazon.com/AWSJavaScriptSDK/latest/frames.html
- モダンブラウザを全てサポート
- ブラウザ上のJavaScriptからAWSサービスに直接アクセスが可能
  - サーバ不要
  - CORSに対応した全てのAWSサービスをサポート

サポートするブラウザ

Google Chrome	28.0+	Microsoft Internet Explorer	10.0+
Mozilla Firefox	23.0+	Apple Safari	5.1+
Opera	17.0+	Android Browser	4.3+

操作可能サービス

Auto Scaling	EC2	OpsWorks
CloudFormation	ECS	Redshift
CloudFront	ELB	RDS
CloudHSM	Elastic Beanstalk	Route53
CloudSearch	Elasticache	S3
CloudTrail	Elastic Transcoder	SES
CloudWatch	EMR	SimpleDB
CloudWatch Logs	Glacier	SNS
CodeDeploy	IAM	SQS
Cognito	Import/Export	Storage Gateway
Config	Kinesis	STS
Data Pipeline	KMS	Support
Direct Connect	Lambda	SWF
DynamoDB	Mobile Analytics	VPC

※薄字のサービスは未サポート

インストール

- HTML内にscriptタグを記載

```
<script src="https://sdk.amazonaws.com/js/aws-sdk-2.1.17.min.js"></script>
```

- Bowerを使ったインストールも可能

```
bower install aws-sdk-js
```


AWS SDK
を使えばプログラマも
スケーラブルなプラットフォームを
今すぐ簡単に使えます
amazon
web services

参考資料

- ブログ
  - AWS PHP Development
    http://blogs.aws.amazon.com/php/
  - AWS Ruby Development
    http://ruby.awsblog.com/
  - AWS Java Development
    http://java.awsblog.com/
  - AWS Mobile Development
    http://mobile.awsblog.com/
- SDK全般
  - http://aws.amazon.com/jp/tools/

Webinar資料の配置場所

- AWS クラウドサービス活用資料集
  - http://aws.amazon.com/jp/aws-jp-introduction/

プロダクト別：				
Amazon S3		AWSマイスターシリーズ Re:Generate Amazon Simple Storage Service (S3)	Slideshare	PDF
Amazon Glacier		AWSマイスターシリーズ Reloaded Amazon Glacier Amazon Glacierのご紹介 機能編	Slideshare（Reloaded） Slideshare（機能編）	PDF（Reloaded） PDF（機能編）
Amazon Route 53		AWSマイスターシリーズ Re:Generate	Slideshare	PDF

公式Twitter/Facebook
AWSの最新情報をお届けします

@awscloud_jp

もしくは
http://on.fb.me/1vR8yWm

最新技術情報、イベント情報、お役立ち情報、お得なキャンペーン情報などを
日々更新しています！