AWS麻雀・ドンジャラ CDP役一覧 Ver1.0

JAWS-UG

牌の種類

1 ECU
2 ECU
C3
C3
C3
3 ECU
4 ECU
5 ECU
6 ECU
7 ECU
8 ECU
9 ECU
1 TB
2 TB
3 TB
4 TB
R
5 TB
MS SQL
6 TB
Oracle
7 TB
MySQL
8 TB
9 TB
1 Gbps
2 Gbps
3 Gbps
4 Gbps
5 Gbps
6 Gbps
7 Gbps
8 Gbps
9 Gbps
US-
EAST
東
ap-
SOUTH
east
南
US-
WEST
西
ap-
NORTH
east
北

牌の説明（サービスレイヤー：コンピュート）

Amazon Ec2

Instances with CloudWatch

Auto Scaling

Instances

cluster

Amazon Lamda

optiized Instamnces

AMI

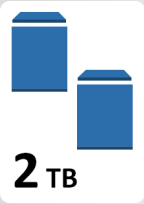
Amazon Kinesis

牌の説明（サービスレイヤー：データストア）

Dynamo DB

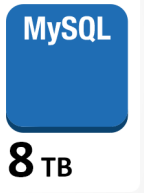
bucket

Oracle DB instance



Amazon EBS

Redis



MySQL DB instance

snapshot

MS SQL instance

Amazon Redshift

牌の説明（サービスレイヤー：ネットワーク）

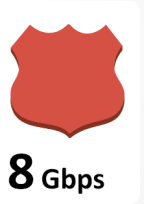
AWS Direct Connect

Internet gateway

VPN connection

Elastic Load Balancing

virtual private gateway



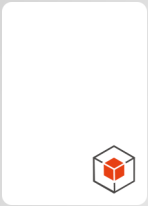
hosted zone

elastic network instance

VPC peering

Cloud Front

牌の説明



SDKs Java

AWS OpsWorks

Amazon CloudSearch

SDKs Node.js

Elastic Beanstalk

Amazon SES

SDKs PHP

AWS CloudFormation

Amazon SQS

SDKs Pyton

※白に相当

AWS CodeDeploy

※撥に相当

Amazon SNS

※中に相当

牌の説明

US-EAST 東

リージョン
バージニア

JAWS-UG

ap-SOUTH east 南

リージョン
シンガポール

JAWS-UG
エンタープライズ

US-WEST 西

リージョン
カリフォルニア

JAWS-UG
中央線

ap-NORTH east 北

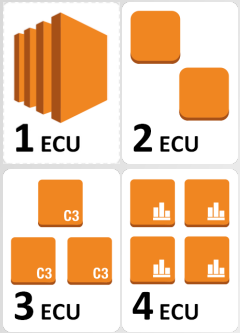
リージョン
東京

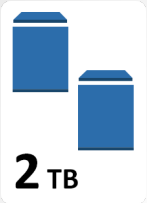
JAWS-UG
女子会

JAWS-UG牌について

麻雀の場合はドラ
ドンジャラの場合はオールマイティ
として扱う

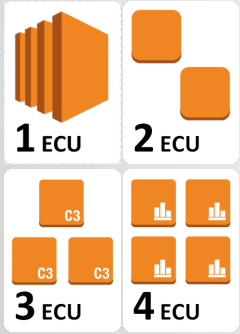
CDP

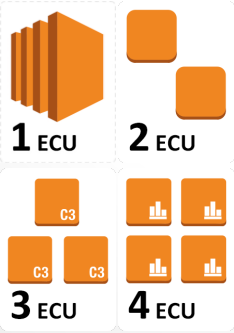




Snapshot

ある瞬間のデータをスナップショット(バックアップ)として作成しS3に保存する事でいつでも復元できるようにする。APIを利用して自動バックアップ作成がよくある使い方。

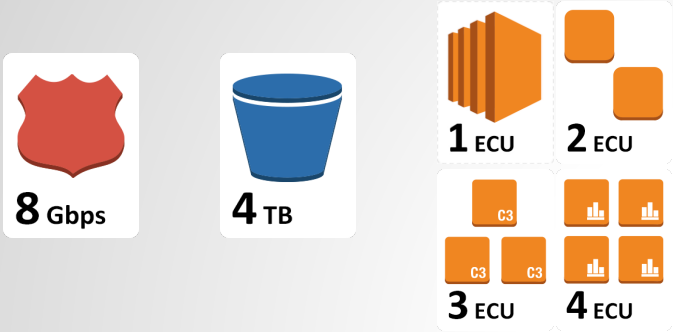




Stampパターン
(EC2は同じ牌を2枚そろえる)

AMIを利用する事で、簡単に同じ環境を用意する事が可能。同じ環境を複数台構築する場合にとても便利になる。

CDP



Web Storage

大容量のファイルや静的コンテンツなどを S3から配信する事でEC2への負荷を減らす。動的コンテンツは EC2より配信する。

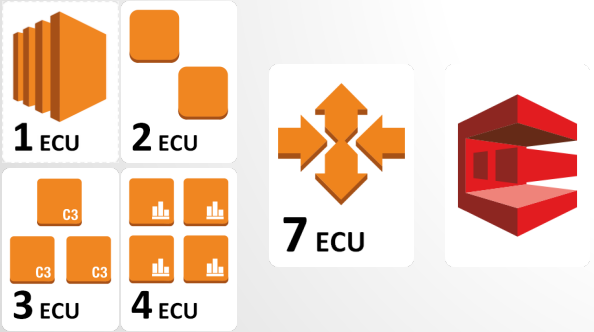
Cache Distribution

Cloudfrontを利用する事で、世界中にあるオリジンサーバーから遅延なくコンテンツを配信できる。
まとめるとサイトが早くなり、ユーザーへのレスポンスが良くなり、EC2へのアクセス負荷も減ります。

CDP

Direct Hosting

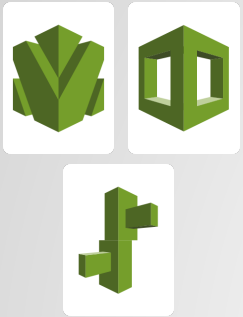
R53、Cloudfront、S3を利用する事で、絶対に落ちない静的サイトを構築する事が可能となる。



Job Observier

SQSを利用して、CloudWatcheで指定した閾値を超えた場合、自動でAutoscalingを行う。
負荷に応じて、EC2の台数を増減（スケールアウト・スケールイン）する。

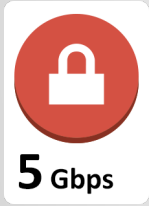
CDP

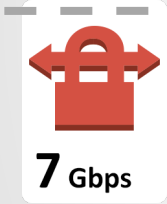


＋コンピュート、データベース、
ネットワークのいずれか
2牌

Stack Deployment

構築したい環境をテンプレートで用意する事で、複雑な環境構築を自動で起動する事ができる。





＋VPN or DX2枚
（バラバラでもOK）

＋VPN or DX5枚
（1枚ずつもOK）

CloudHub

CloudHub(ダブル)

VPC内にVPNハブを構築することで、複数拠点から接続可能なVPN接続を実現する事が可能となる。

CDP

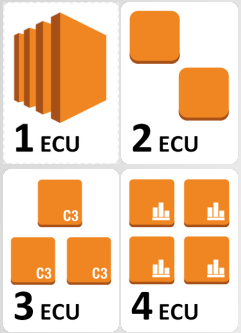
Back Net

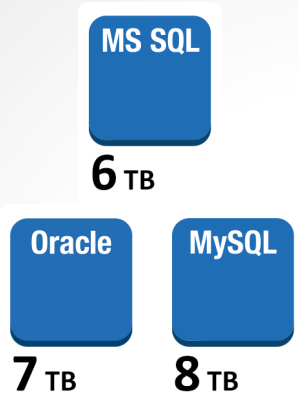
EC2に対して、2つのENI(仮想ネットワークインタフェース)を用意する事で、公開用ネットワークインタフェースと管理用ネットワークインタフェースを利用する。

State Sharing

ステート情報(セッション情報、ユーザー情報)などをDyanamoDB、Redisに保持することで、サーバー増減時にステート情報の喪失を防ぐ。

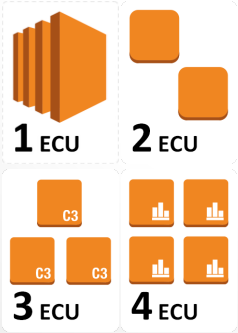
CDP





Inmemory DB Cache

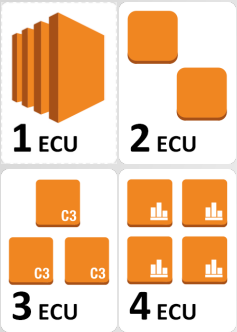
頻繁に読み込まれるデータを Redisにキャッシュする事で、DBから呼び出すことなくRedisからキャッシュデータを取り出す。



Scheduled Autoscaling

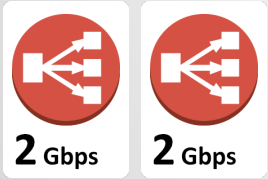
アクセスが急増するタイミングがわかってる場合、スケジューリングからスケールアウトする事で、サービスを止めずに運用が可能となる。

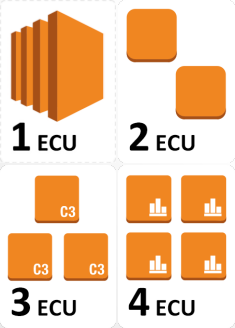
CDP



Storage Index

インターネットストレージにデータを格納する際、同時に検索性能の高いKVS(Dynamo DB)へメタ情報を格納し、その情報をインデックスとして利用する。検索時は KVS (Dynamo DB)を用い、得られた結果を基にインターネットストレージへアクセスする。

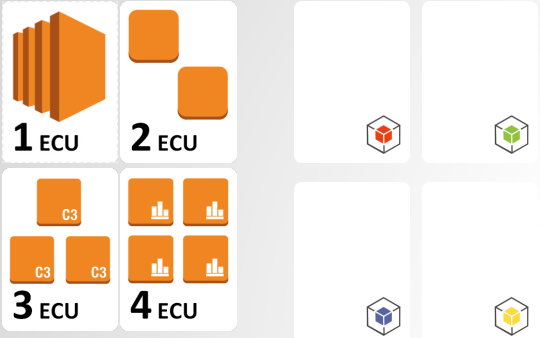




Multi Load Balancer

ELBを複数台用意する事で、同一サイトで ELB毎に挙動を変える事ができる。
PCサイト、スマホサイトを ELBを利用してアクセス先を変更できる。

CDP



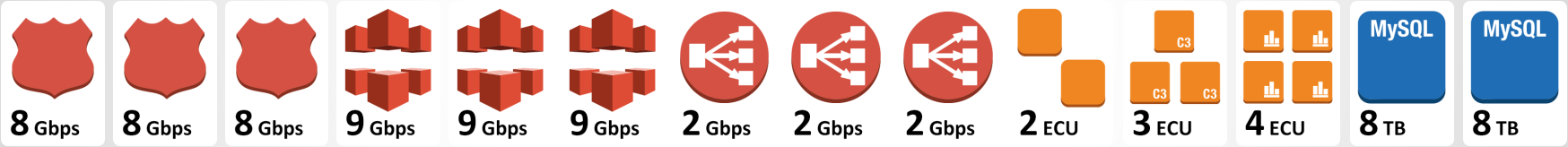
SDKを利用したS3書込

SDKを利用し、サーバー上にアップされたファイルを S3へ保存する。

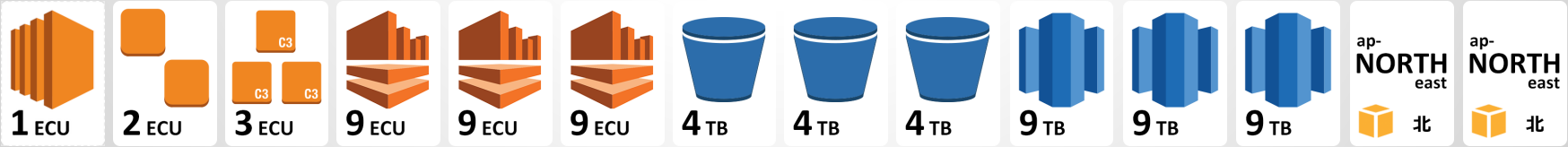
特殊あがり(麻雀限定)

- リファレンスWebアーキテクチャ(役満)

Route53　CloudFront　ELB　EC2　RDS (Muliti-AZ)

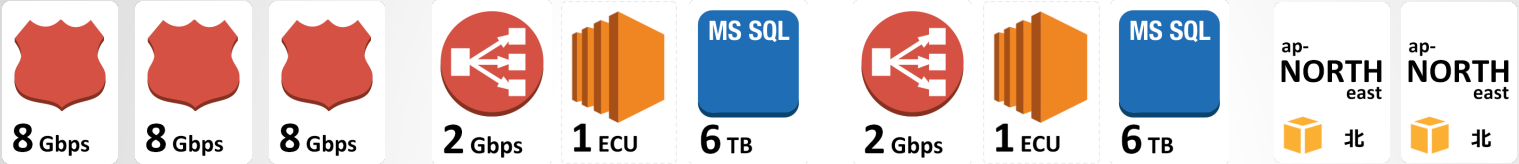


- スシロー(役満)

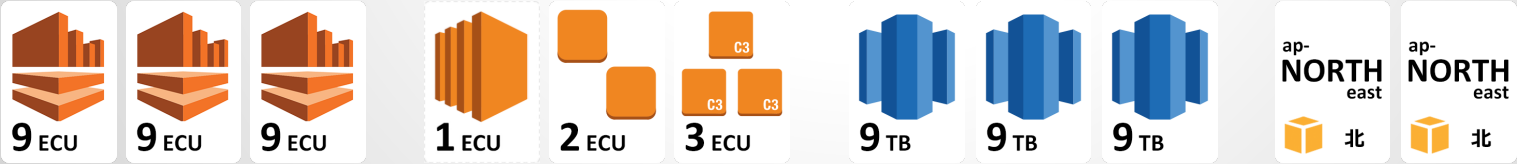


特殊あがり(ドンジャラ限定)10000点

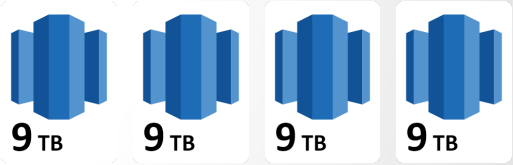
- リファレンスWebアーキテクチャ



- スシロー



特殊役

- Kinesushi
  Kinesis暗カンで+3000、明カンで+2000

- Docomo
  Redshift暗カンで+3000、明カンで+2000