CDN in Mercari

Fastly Yamagoya Meetup 2017

Masahiro Nagano @kazeburo

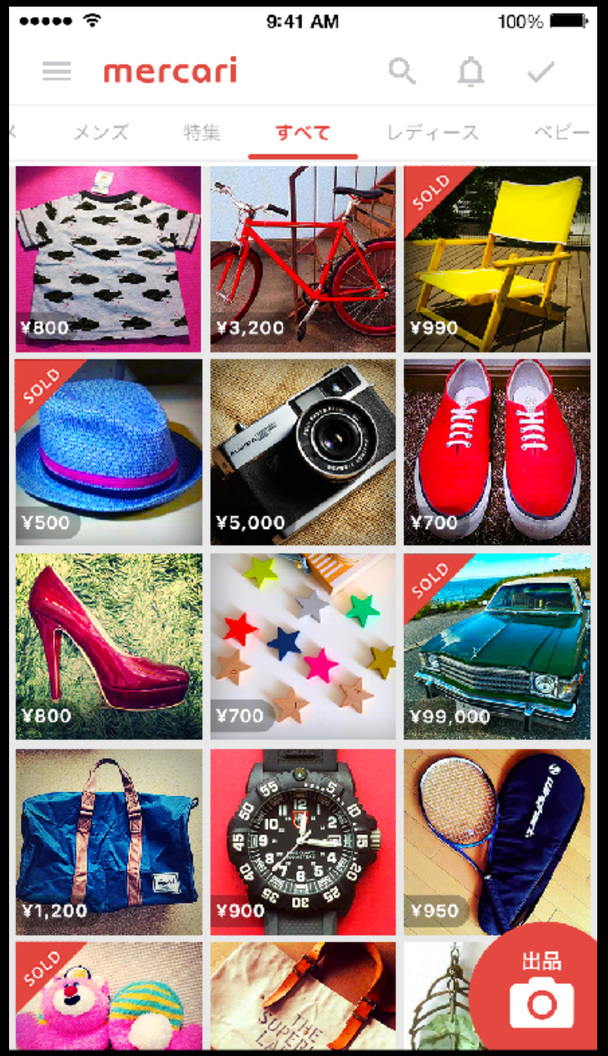
Me

- Masahiro Nagano / 長野雅広
- @kazeburo
- Mercari, Inc
  Principal Engineer
  Site Reliability Engineering (SRE) Team

Agenda

- メルカリの紹介
- 先日の個人情報流出インシデントについて
- メルカリのインフラストラクチャとCDN
- 今後の課題・取り組み

Mercari

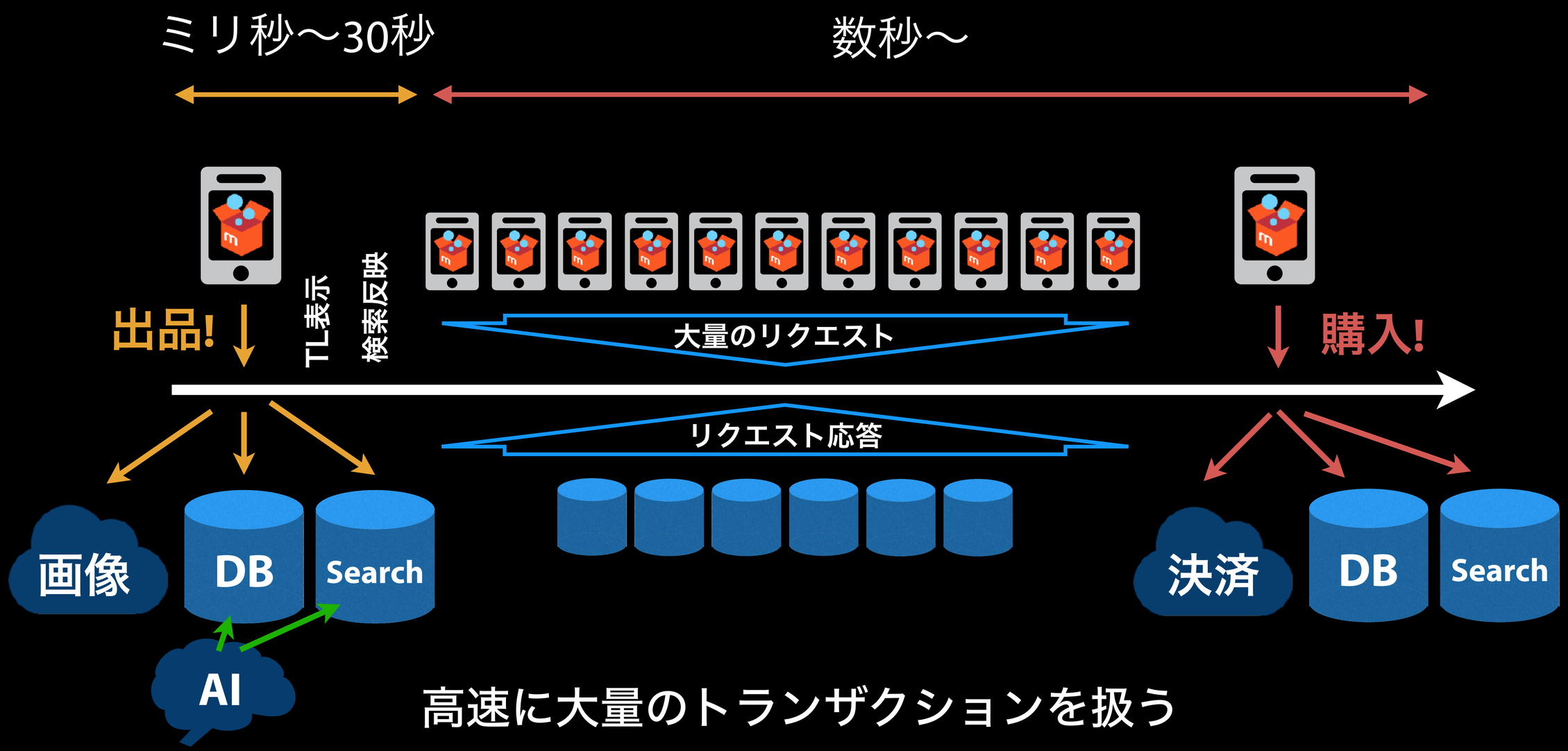


- フリマアプリ
- スマホで写真をとって簡単に出品
- 安心・安全な決済
- 便利な配送

Mercari

ダウンロード数	7500万DL(JP+US)
出品数	1日100万品以上
流通額	月間100億円以上

システムからみたメルカリ

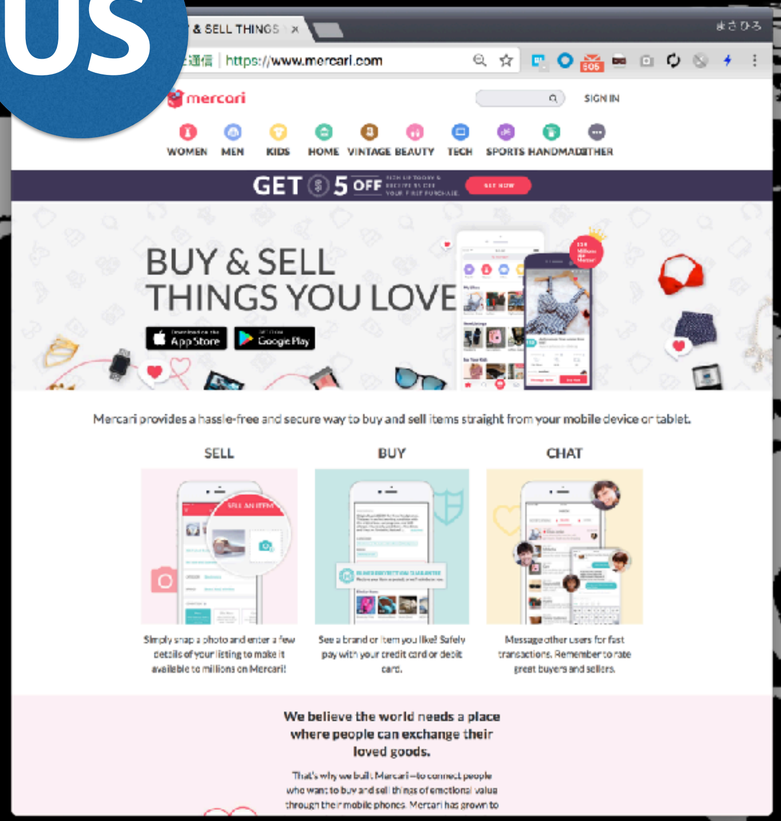


高速に大量のトランザクションを扱う

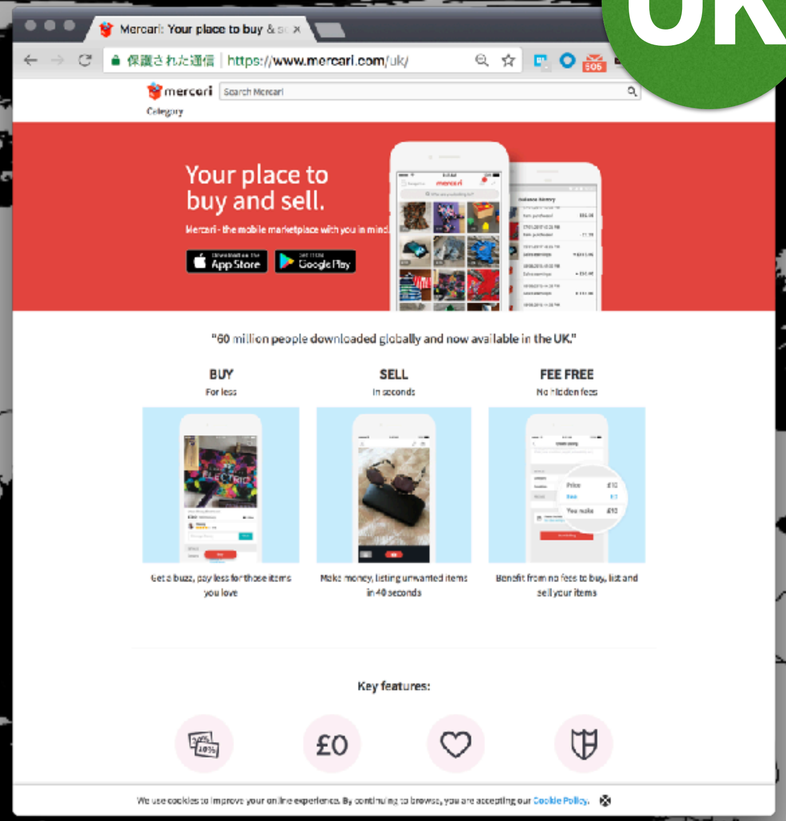
Mercari

JP

US



UK



JP/US/UKで展開中

CDN切り替えに伴う
Web版メルカリにおける個人情報流出
インシデントについて

多くの皆様にご心配、ご迷惑をお掛けいたしました
深くお詫び申しあげます

多大なる協力をFastly様にいただきました。改めてお礼申し上げます

発生した事象

- メルカリWeb版へのアクセス速度向上と、セキュリティ向上のため、CDNの切り替えを行いました。その際に切り替え先のCDN(Fastly)におけるキャッシュの動作についての把握が不足しており、お客さまへのレスポンスが別のお客さまに意図せず表示され、結果として個人を特定できる情報を含む内容が本人以外に閲覧される状態となりました

Timeline

- 6/22
  - 9:41　CDNの切り替えを実施（問題発生）
  - 14:41　カスタマーサポートにてお客さまからの問い合わせを確認し、社内へ報告
  - 15:05　CDNの切り替えを中止し、従来のCDNへ戻す
  - 15:16　Web版のメルカリをメンテナンスモードへ切り替え
  - 15:38　CDNの配信設定をdeactivateし、アクセスを遮断
  - 15:47　Web版のメルカリメンテナンスモードを終了
  - 17:55　コーポレートサイトにお知らせを掲載
  - 20:45　Tech blogにて詳細公開

Timeline

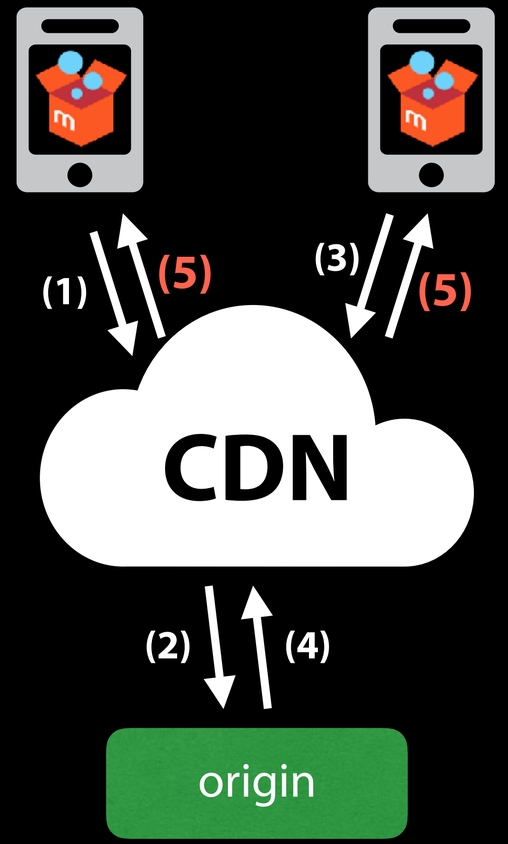
- 6/28
  - 配信設定を再Activate
  - Cacheの完全無効化設定を行い、Fastly社のエンジニアにレビューを依頼
  - nginxの設定を更新し、cacheに関するヘッダを変更
  - 社内で検証
- 6/29
  - Fastly社エンジニア待機のもと、切り替えを実施

Cache無効化

- Cacheを無効化するためには“Cache-Control: private”もしくは "Set-Cookie" が必要
  - ”Cache-Control: no-cache”や“no-store”は無視される
- Expiresヘッダも利用されるが、日付の解釈に失敗あるいは過去日付の場合は“0秒”として扱われる
  - 0秒はcache無効ではなく「0秒のcacheが存在する」

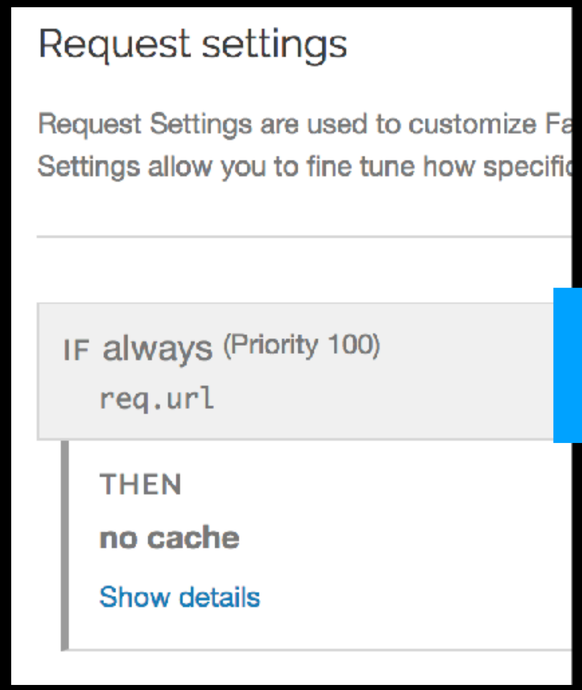
0秒のcache

- CDNからオリジンへのリクエストの処理中に、同じURLに対してリクエストが発生すると、最初のレスポンスを待って、2つ目以降のリクエストにも同じレスポンスが返される
  - `Request collapsing`



対策

- VCLの設定変更
  - 複数のbackendに対応するため、Priorityを低く設定




```
# ここにbackend切り替えのvclが描いてある

# Request Condition: always Prio: 100
  if( req.url ) {  ## 必ずtrueになる
        if (!req.http.Fastly-FF) {
          if (req.http.X-Forwarded-For) {
            set req.http.Fastly-Temp-XFF = req.http.X-Forwarded-For ", " client.ip;
          } else {
            set req.http.Fastly-Temp-XFF = client.ip
          }
        } else {
          set req.http.Fastly-Temp-XFF = req.http.X-Forwarded-For;
        }
        return(pass);
}
```

対策

- ログをS3に送信し、lambdaによりリアルタイムに解析、mackerelで可視化。監視

- nginx/Apacheにてheaderを追加

```
more_clear_headers 'Expires';
more_set_headers "Cache-Control: private, no-cache, no-store, must-revalidate" "Pragma: no-cache";
add_header Set-Cookie  "merCtx=\"\"; HttpOnly" always;
```

- headerやVCLの変更の監視

キャッシュしない

特定のリソースをキャッシュしない場合、ヘッダを次のように設定します。

```
Cache-Control: private
```

事前に `max-age=0` または `Expires` を設定しただけでは、あるタイミングにおける複数の未処理リクエストを処理するために、そのコンテンツが使用されないことは保証されません。また、エラーの場合や、バックエンドの同じオブジェクトへのリクエストをすでに処理中の場合に、期限切れの古いオブジェクトが使用される可能性があります。

https://docs.fastly.com/ja/guides/tutorials/cache-control-tutorial

メルカリのインフラストラクチャとCDN

CDN in Mercari

- より多くのトラフィックを処理するため
  - インフラ運用コストの削減としてCDNを利用
  - 画像配信/動画配信
- UX/セキュリティの改善のため
  - Clientとの通信環境最適化
  - サイト/APIを丸ごとCDN化

Infrastructure

JP

SAKURA internet

S3

商品画像

US

amazon web services™

Backend API

S3

商品画像

Google Cloud Platform

Frontend for Backend microservices

UK

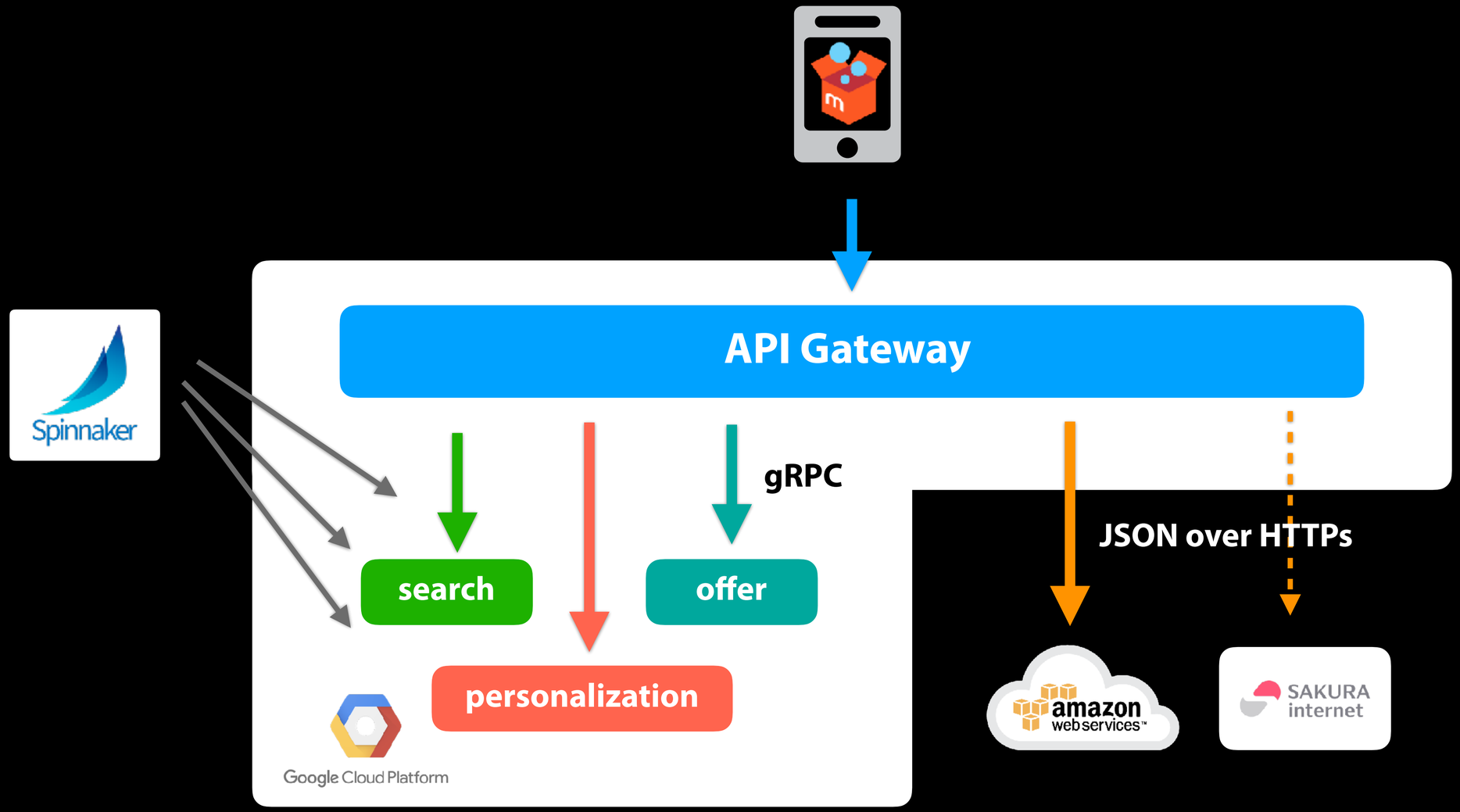
Google Cloud Platform

S3

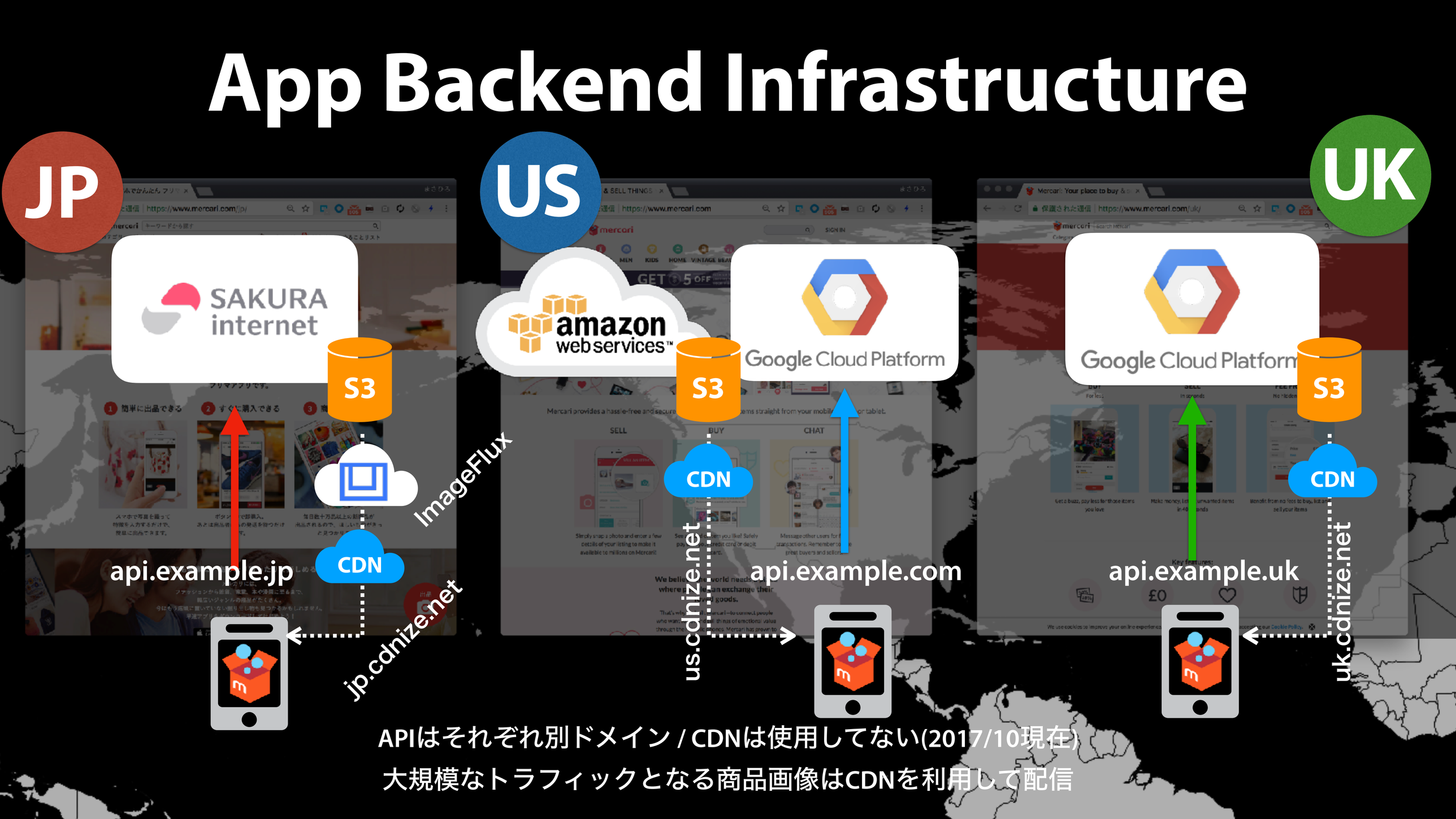
商品画像

それぞれの域内のDCを利用

Frontend for Backend/microservices



App Backend Infrastructure



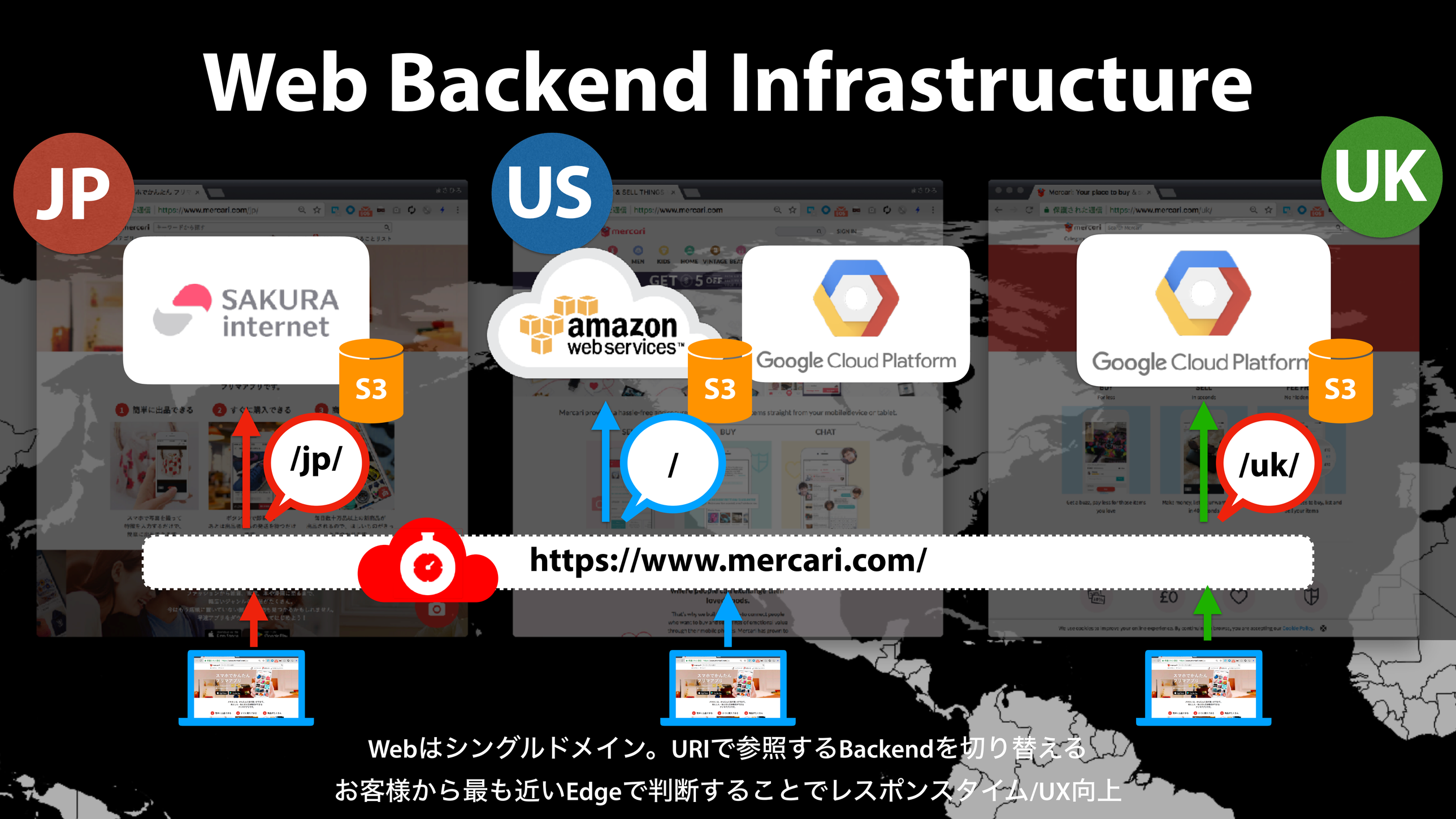
APIはそれぞれ別ドメイン / CDNは使用してない(2017/10現在)

大規模なトラフィックとなる商品画像はCDNを利用して配信

ImageFlux

- さくらインターネットとpixivの協業
- 画像変換 + Cache Storage
  - WebPへの対応
  - CDNのCache Hit Rate改善

Web Backend Infrastructure



Webはシングルドメイン。URIで参照するBackendを切り替える

お客様から最も近いEdgeで判断することでレスポンスタイム/UX向上

今後の課題・取り組み

今後の課題・取り組み

- **Securityへのフォーカス**
  - L3-L7まで大規模なDoS攻撃からの防御
  - WAF の検証
  - Bot制御
- **高度化するEdge Computing**
  - Microservices を実現する手段としてのCDN
  - CDNの設定/VCLのContinuous Integration
- **コスト最適化** 

以上

Fastlyの皆様、今後ともよろしくお願いします

=> www.mercari.com/jp/jobs/