

# AMP Roadshow Tokyo 2018

Kenta Tabata

---

---

## 目次

---

---

1. はじめに .....	2
1.1. What is this? .....	2
1.2. このレポートを読む際の注意点 .....	2
2. そもそも AMP とは .....	2
2.1. AMP HTML .....	3
2.2. AMP JS .....	4
2.3. AMP Cache .....	4
3. イベント概要 .....	5
3.1. タイムテーブル .....	5
4. Introduction and Keynote .....	6
5. Rich content experiences with AMP .....	7
6. Advanced interactivity with AMP .....	8
7. Progressive Web AMPs .....	9
8. AMP in production .....	9
8.1. Generate .....	9
8.2. Optimize .....	10
8.3. Validate .....	10

8.4. Measure .....	10
8.5. Distribute .....	10
9. What's next of AMP .....	10
10. Guest Talk or Panel .....	10
10.1. CyberAgent の例 .....	10
10.2. 一休の例 .....	11
11. Q&A with AMP team .....	11
12. Monetization with AMP .....	11
13. Codelabs to learn AMPs .....	12
13.1. AMP の基本 .....	12
13.2. インタラクティブな AMP ページを作成する .....	12

---

---

## 1. はじめに

---

---

### 1.1. What is this?

AMP Roadshow Tokyo 2018 のレポート。その場で書き取ったメモ+後から追加したコメントで構成している。なお、会場のスライドはネットでは公開しないようです。

### 1.2. このレポートを読む際の注意点

青字は私のコメントです

---

---

## 2. そもそも AMP とは

---

---

<https://www.ampproject.org/learn/overview/> より , 実は日本語版があるのに訳して気づく。

AMP はオープンソースライブラリであり、魅力的で、スムーズでかつ、瞬時にロードできる web ページを簡単に作る方法を提供します。AMP ページは、あなたによってコントロールでき、あなたがリンクできる単なるウェブページでしかありません。AMP はあなたに既に

あるスキルセットや Web ページをつくるフレームワークで構築することができます。AMP は多くの異なったプラットフォームでサポートされていて、ブラウザ間で互換性を保ちます。AMP エコシステムには、出版、広告、電子商取引、地方や小規模企業など、2500 万のドメイン、100 以上のテクノロジープロバイダ、主要なプラットフォームが含まれます。

AMP ページは以下の 3 つのコアコンポーネントで構築されます。

- AMP HTML
- AMP JS
- AMP Cache

## 2.1. AMP HTML

AMP HTML は、信頼性のあるパフォーマンスのためにいくつかの制限のある HTML です。AMP HTML は基本的にカスタム AMP プロパティで HTML を拡張しています。もっともシンプルな AMP HTML は次のようになります。[style タグ、存在感があるが実際はワンライ](#)  
[ン](#)

```
<!doctype html>
<html amp>
<head>
  <meta charset="utf-8">
  <link rel="canonical" href="hello-world.html">
  <meta name="viewport" content="width=device-width,minimum-scale=1,initial-scale=1">
  <style amp-boilerplate>body{-webkit-animation:-amp-start 8s steps(1,end) 0s 1 normal both;-moz-animation:-amp-start 8s steps(1,end) 0s 1 normal both;-ms-animation:-amp-start 8s steps(1,end) 0s 1 normal both;animation:-amp-start 8s steps(1,end) 0s 1 normal both}@-webkit-keyframes -amp-start{from{visibility:hidden}to{visibility:visible}}@-moz-keyframes -amp-start{from{visibility:hidden}to{visibility:visible}}@-ms-keyframes -amp-start{from{visibility:hidden}to{visibility:visible}}@-o-keyframes -amp-start{from{visibility:hidden}to{visibility:visible}}@keyframes -amp-start{from{visibility:hidden}to{visibility:visible}}</style><noscript><style amp-boilerplate>body{-webkit-animation:none;-moz-animation:none;-ms-animation:none;animation:none}
```

```
</style></noscript>
  <script async src="https://cdn.ampproject.org/v0.js"></script>
</head>
<body>Hello World!</body>
</html>
```

AMP HTML ページのほとんどのタグは、一般的な HTML のタグですが、AMP-specific なタグに換えられているものもあります。それらのカスタム要素は AMP HTML コンポーネントと呼ばれ、一般的なパターンを効率よく簡単な方法で実装できます。例えば、amp-img タグはまだブラウザによってはサポートされていない完全な srcset を提供します。

AMP ページは、<link rel=""> HTML tag で検索エンジンや他のプラットフォームに発見されます。あなたは non-AMP と AMP で構成されたサイトか、AMP のみで構成されたサイトのどちらかを選ぶことができます。

## 2.2. AMP JS

AMP JS ライブラリは AMP HTML ページの高速なレンダリングを可能にします。AMP JS ライブラリは、AMP のパフォーマンスのための全ての最適化を実施し、リソースのローディングを管理し、上述のカスタムタグを使用することで、高速なページレンダリングを確実にします。最も効果のある最適化の一つは、外部リソースを全て非同期化することです。ページの中でレンダリングされる要素をブロックするものが無くなります。他の最適化技術は、すべての iframe をサンドボックスにすることや、リソースが読み込まれる前にページの全ての要素のレイアウトの再計算をすることや、遅い CSS セレクタを無効化することです。

## 2.3. AMP Cache

Google AMP キャッシュはキャッシュされた AMP HTML ページの配信に役に立ちます。Google AMP キャッシュは全ての正当な AMP ドキュメントを配信するプロキシベース CDN です。CDN は AMP HTML ページを取り出し、キャッシュし、ページのパフォーマンスを自動的に改善します。Google AMP Cache を使うとき、最大効率のため、ドキュメント、JS ファイル、イメージの全てが同じ場所から HTTP2 で読まれます。またキャッシュは検証システムが組み込まれており、ページが動くことや外部のリソースに依存しないことを保証します。検証システムはページのマークアップが AMP HTML 仕様に沿っていることを確認する一連のアサーションを実行します。別の検証ツールが全ての AMP ページに付属しています。これは、ページがレンダリングされる際、ブラウザのコンソールに直接バリデーションエラーをログすることができ、それをみることであなたのコードの変化がどのくらい複雑

にパフォーマンスや UX に影響を与えたかを確認することが可能です

---

## 3. イベント概要

---

- イベント名： AMP Roadshow Tokyo
- 日程： 2018 年 4 月 27 日（金） 9:30 - 17:00（開場： 9:00）
- 場所： グーグル合同会社
- 定員： 240 名

### 3.1. タイムテーブル

- 9:00am 開場
- 9:30am Introduction and Keynote
- 9:45am Rich content experiences with AMP
- 10:15am Advanced interactivity with AMP
- 10:45am 休憩
- 11:00am Progressive Web AMPs
- 11:30am AMP in production
- 12:00pm What's next in AMP
- 12:30pm ランチ
- 1:30pm Guest Talk or Panel
- 2:00pm Q&A with AMP team
- 2:30pm Monetization with AMP
- 3:00pm Amplify live!
- 5:00pm まとめ

---

## 4. Introduction and Keynote

---

ご質問は <https://www.sli.do/> に投げてください。AMP 勉強の参考サイトは以下。

- <https://www.ampproject.org>
- <https://ampstart.com>
- <https://www.ampbyexample.com>

AMP はオープンソース。Google とは独立したものとして捉えてください。

AMP には 3 つの部分がある。(AMP HTML, AMP JS, AMP Cache)

HTML と JS を拡張する(でも JS の拡張はつらいのであとで変えたい)

AMP Cache: Google のもっているやつ

ページロードを 3G 回線で 10s から 3,4s に小さくした(日本では 4G だけど、世界的には 3G がまだまだ主流)

ページロードが 1 秒遅れるとコンバージョンは 7% 低下する。3 秒以上かかるとユーザーの半分は離脱する

AMP 入れるとページロードが遅くなるから入れられない。この AMP LOADSHOW ではその誤解を解きます。

AMP サイトは 40 億を超えている。

以下 2015 までのタイムライン [図があって右矢印\(タイムライン\)](#) に以下のコンテンツが時系列順にならんでました

- Header Bidding
- AMP Stories
- AMP for Email
- AMP Ads
- WordPress

- New URL

2017 年の電子商取引収入の半分はモバイルからのものです

Ali Express の Web サイトの速度改善の例です

- 36% led to 10.5 orders
- 27% のコンバージョン増加 (そのうち、AMP で 4% のコンバージョンを増加)

PWA <-> AMP *A と P を入れ替えて、W と反転させるアニメーションを入れて、二つが密接に関連していることをアピールしていました*

AMP はライブラリであると捉えてください

---

---

## 5. Rich content experiences with AMP

---

---

*このセッションでは具体的な AMP HTML の解説をしていました*

AMP CSS 50kb JS ばつ *ここは何を書きたかったのか* ❷.

AMP は Web Components で最も成功した例

複雑を単純に

AMP ライブラリは煩雑なレスポンスをどうやるのか *発音はレスポンスインシブルでした*  
コードは以下 . layout 属性に responsive を設定すると、要素の幅はコンテナ要素 (div とか) の幅に等しくなり、高さについては、幅と高さの属性によって決まるアスペクト比になるよう自動的に調整されます。srcset 属性が画面の幅に基づいて使用される画像を指定します。AMP だと polyfill が古いやつでも利用可能 *CSS を頑張って記述しなくていいのが利点か*

```
<amp-img
  src="/img/narrow.jpg"
  srcset="/img/wide.jpg 640w,
          /img/narrow.jpg 320w"
  width="1698"
  height="2911"
  layout="responsive"
  alt="an image">
```

```
</amp-img>
```

<amp-youtube> 動画 ID からパフォーマンスを落とさず、埋め込み可能

```
<amp-youtube
  data-videoid="mGENRKrdoGY"
  layout="responsive"
  width="480" height="270"></amp-youtube>
```

<amp-twitter>

<amp-gist> gist を普通に読み込むとパフォーマンスが落ちるが

<amp-carousel> パフォーマンスを落とさないカルーセル controls をつける

他、<amp-accordion> や <amp-lightbox> など

<amp-social-hoge> ネイティブのシェアを起動できる自ドメインから AMP を配信する場合 AMP Cache ほどはやくならない

<amp-toolbox-optimizer> AMP Cache 同等の最適化を提供する node モジュール

試したいなら , AMP プレイグラウンドがあります <https://ampbyexample.com/playground/>

例 : Ali Express, 一休 .com

AMP Start にサンプルがあります <https://www.ampstart.com/>

AMP でウェブページが作れるぞ！を覚えて帰ってください

---

---

## 6. Advanced interactivity with AMP

---

---

このセッションでも引き続き具体的な AMP HTML の解説をしていましたが、インタラクティブなコンテンツを提供するための方法を説明されていました

User -> AMP Page <- AMP Cache

欲しいもののリスト コメントをする



<amp-selector> 色を、量、サイズをウェブサイト上で選択させる場合に使用する

<amp-list> & <amp-mustache>

<amp-bind> react とかのやつができる data-binding

Combining components  $X + Y = ?$

---

---

## 7. Progressive Web AMPs

---

---

メインは PWA の話ですが、後半は PWA で AMP をやるエモい話でした

PWA はブラウザ API の組み合わせ。

ネイティブアプリ。ユーザーは3つのアプリに、アプリに使う時間の8割を使っている。残りの2割を新規のアプリは奪う必要がある。また、一ヶ月につきインストールするアプリの平均は0

Service Workers: A client-side JavaScript proxy 使い方 <link rel="manifest" href="/manifest.json"> manifest.json <start\_url = "">

PWA or AMP? 別に対比するものではなく、共存可能 AMP = web page = PWA The AMP Cache Problem PWA は同一ドメインである必要がある。あれ、どうするの? <amp-install-serviceworker> URL を serviceworker 内で書き換える?

質問: 運用経験ありますか? AMP: 3割くらい. PWA: あまりいなかった.

先進的な話 PWA のページの中に AMP を入れよう! 利点: With Shadow DOM Window 一回の読み込み AMP Library document 無限大例: amp.cards \* Uses the Guardian's live AMP pages. \* ... Web Android iOS PWAShell Native Shell Native Shell AMP

---

---

## 8. AMP in production

---

---

Production-ready AMP pages. ある WebPage を本番環境に乗せるまでによく起きる問題点と対処方法

### 8.1. Generate

手ではつくらない 3p CMS (RSS Web, AppliNews) standard html -> amp html drupal am-

p /wordpress amp 環境によって、手に入る AMP ページが違ったりする。

## 8.2. Optimize

動機：機能追加とか、バグ修正とか \* いらない amp を呼ばない \* 必要な CSS のみを load する Global CSS -> Local CSS \* Image Optimization, website を読んで

## 8.3. Validate

amp page がスペック（ブラウザ）を満たしているか CI 環境に毎回チェックする取り込みましょう！ npm amphtml-validator さらによい方法はモニタリングすること amphtml-validator + cronjob でもよいが -> GoogleSearchConsole

## 8.4. Measure

lighthouse webpagetest <amp-analytics> もあるぞ！ Measure + Collect + Batch

## 8.5. Distribute

配信による問題 AMP Page - Google - Bing - Twitter AMP Page -> metadata -> cache コース  
問題：ドメインがなんちゃらで fail すること endpoint-com.ampproject.org -> endpoint.com  
<Access-Control- : htt > というタグをつける Content mismatch

---

---

## 9. What's next of AMP

---

---

\* amp-datepicker Infinite Scroll \* SSR of AMP 普通のレンダリングフロー browser loads the page AMP Boilerplate CSS hides the page AMP JS loads AMP inserts the AMP CSS AMP bootstraps web componets AMP shows the page それを SSR でやる まったくわからない ... \* CSS comverter AMP は使えない CSS がある \* hoge.. \* Allow developers to program AMP pages with JavaScript AMP library : github 11000starts Contribute: bit.ly/helpamp

---

---

## 10. Guest Talk or Panel

---

---

### 10.1. CyberAgent の例

why amp? search UX and mobile US Sessions 7x の Sessions AD CPM Change AD pos-

sions Use <amp-ad-sticky> Search Bar <amp-bird> <amp-from> <amp \* 状態管理 \* なんか Vue っぽいことしてる

## 10.2. 一休の例

デザイナーとエンジニアの役割新しい企画をする場合、とりあえずデザイナーがモックを作る。リアルタイム性の低いコンテンツはキャッシュに送る stylelint

---

---

## 11. Q&A with AMP team

---

---

もう amp を iframe を使わないでください。理由 viewport の設定が煩雑 tracking 要素に使われるのは 1 つ analyti クラックできる amp-pixel

AMP page から AMP page へ遷移するときキャッシュの URL を指定して高速化を図りたいことがあるのですが、現状スマートな実現方法はありますか？ -> 検討中

amp に適さないパターンは？ google docs とかもともと、mobile web を高速にするために生まれた。ec2 サイトでローディングを加速するために使ってください js や css が複雑かしてサイトは遅くなってるので

amp react や amp angular はでてこないよ js が amp になるかもしれけどね クッキーに personalize データを持たせて、できる .. なんの話だったか

フルアンプのサイトはある？ ない。

---

---

## 12. Monetization with AMP

---

---

Adverinig & AMP

マネタイズ

Ads on the Web Today

広告が好きな人がいますか？

なんでみんな嫌いなのか

Disrupive ads 表示が崩れる UX が悪くなる

Slow loadings ads 遅くなる

Unsafe ads マルウェアにリダイレクトする

何が広告として配信されているか publisher は よくわからない！

Ad に対する AMP の試み

set dimensions

separate the ads

prioritized content `<amp-ad> type=" data-loading-strategy="2.0"` 二つの viewport の先に ad をよみこみ `</amp-ad>`

ad impression 328increase in ad revenue paid to publishers メモ取れず

New Ad Features

\* Single Request Architecture \* Ad Fresh .. # How do AMPHTML Adds work? 1 -> Validation through crypto signing 有効な AMP ページかを検証する (上の validation フェイズ)  
Unsafe な広告をカット 2 ad を含める 3 4 Single high-rekdfkdfdf

# ads battery use AMP により、80

3x 軽く 6x 高速化 13

# celtraQ + # AMP for a landing page # Ensure that post click experience

Using AMP for your Adword landing page. 結局 4 パターンのマネタイズの方法があります。

---

---

## 13. Codelabs to learn AMPs

---

---

### 13.1. AMP の基本

### 13.2. インタラクティブな AMP ページを作成する

前書き: `<amp-bind>` を用いてインタラクティブな AMP ページを作るために data binding

の方法を学ぶ。ここでは、e- コマースの製品ページをつくる。要点：使用する技術：速く，リッチな UX を持った AMP HTML, AMP コンポーネントを使う。<amp-bind> を使用して、要素間のインタラクティブ性を追加する。<amp-state> を使用して、必要に応じて別の商品データを取得する。