

WEBパフォーマンスの 基礎知識と困難さについて

Kousuke Miyoshi

2022.12.14

LINE

登壇者紹介



三好 康介

LINE株式会社 UIT室

国内家電メーカー勤務を経てライブドアに入社、合併によりLINE社に所属、主にLINE NEWSのフロントエンドを担当、好きなフレームワークは VueとSvelte

Svelteに関しては Svelte Japan のコアメンバーでもあり、ドキュメントの日本語化プロジェクトに参画しています

<https://svelte.jp/>

WEBパフォーマンスの重要性について

CoreWebVitals とその影響

WEBコンテンツのパフォーマンスは重要であるというのは当然のことであり、それはインターネットがはじまった当初からそうでした。

特に近年では、WEBコンテンツのリッチ化や CoreWebVitals をGoogle検索のランキング指標として組み込むと発表したことにより、より一層WEBパフォーマンスに関して注目が高まっています。

引用：ページ エクスペリエンスの更新に対応するための期間、ツール、詳細情報
<https://developers.google.com/search/blog/2021/04/more-details-page-experience>

しかしWEBパフォーマンスの改善は難しい

WEBパフォーマンス改善の困難さ

しかしながら、WEBパフォーマンスの改善というのは広範な知識と経験が必要となってきます、
なぜなら、あらゆるプロセスがボトルネックになり得るからです。

しかもそれらの情報は分野が違う上に情報量が多すぎるために、まとめて説明や解説されることは稀で
ネットワーク通信やレンダリング、ブラウザの動作などといった各分野で情報が散らばった状態になっている
のが通常です。

WEBパフォーマンスの基礎知識

Smashing Magazine の Front-End Performance Checklist

そんな中、海外の有名WEBマガジン「Smashing Magazine」では、「Front-End Performance Checklist」と題してWEBパフォーマンスに関するあらゆる情報や注意点をまとめた記事を毎年発表しています。

私はパフォーマンス改善に取り組む前に、まず最初にWEBのパフォーマンス関連の記事や事例などをある程度収集してチーム内で内容や感想を共有することからはじめましたが、その中でも各分野の網羅性と情報量という点では、最も優れた記事だと思います。

またこの記事の優れた点は引用が多いことで、「xxxxの研究によるとyyyyと言われており」という内容が多く、そこからさらに情報を深掘りしていくことができます。

引用：Front-End Performance Checklist 2021

<https://www.smashingmagazine.com/2021/01/front-end-performance-2021-free-pdf-checklist/>

Smashing Magazine の Front-End Performance Checklist

Front-End Performance Checklist の注意点

従って予備知識ゼロの状態から、WEBパフォーマンスに関するある程度の知識や情報を得るのに良い資料は何かありますか？と聞かれたら私はこの「Front-End Performance Checklist」をオススメします、WEBだけではなくPDFやWord形式などあらゆる形式で配布されているのもオススメする理由です。

しかながらこの記事には難点もあります、理解するのにパフォーマンスの専門知識を要する上にメチャクチャ長大な英文であるという点です。

具体的には「**英語がスラスラ読めてかつWEBパフォーマンスの基礎知識がある人なら3時間で読める**」
というくらいの分量があります。

引用：Front-End Performance Checklist 2021

<https://www.smashingmagazine.com/2021/01/front-end-performance-2021-free-pdf-checklist/>

遠回りこそが俺の最短の道だった

分かったけどそんなの読んでられない

というわけで、

「なるほどその記事を読んでおけばとりあえず良さそう」

ということが分かってても

「分かったけどそんなの読んでられない」

というのが普通の反応だと思います。

なので、今回はその記事のさわりだけ抜粋した要約を紹介します



WEBパフォーマンスの20%ルール

競合他社より20%早い状態を目標とする

心理学の調査によると、自分のWEBサイトが競合他社のWEBサイトよりも高速であると感じて欲しい場合には、少なくとも**20%高速**である必要があります、競合他社とのパフォーマンスを比較するには ChromeUXレポート (CrUX) や、Treo が利用できます。

また他社事例を知りたい場合は WPO Stats で世界中のサービスのパフォーマンス改善の事例を見ることができるので参考にしてください。

引用：The Need For Performance Optimization: The 20% Rule

<https://www.smashingmagazine.com/2015/09/why-performance-matters-the-perception-of-time/#the-need-for-performance-optimization-the-20-rule>

WEBパフォーマンスで利用される指標

主に lighthouse の指標について

- Time To Interactive (TTI)
- First Input Delay (FID)
- Largest Contentful Paint (LCP)
- Total Blocking Time (TBT)
- Cumulative Layout Shift (CLS)

First Meaningful Paint (FMP) は不正確な場合があるため最近非推奨になりました、
それに代わるものとしてLCPが採用され、lighthouse でもサポートされなくなりました。

引用 : Developing the Largest Contentful Paint Metric

<https://calendar.perfplanet.com/2019/developing-the-largest-contentful-paint-metric/>

CoreWebVitalsの登場

新しいパフォーマンス指標のセット

2020年5月 Google が CoreWebVitals を発表しました、それはユーザー体験に焦点を当てた新しいパフォーマンス指標のセットです。

- Largest Contentful Paint (LCP) - 2.5秒未満
- First Input Delay (FID) - 100ms未満
- Cumulative Layout Shift (CLS) - 0.1未満

前述のとおり、Google検索のランキングの指標にも使われるのでビジネス的にも重要性が高いものとなっています。

CoreWebVitalsの注意点

対応ブラウザが偏っている

しかしながら CoreWebVitals は対応ブラウザが偏っており、スマートフォンにおいては

FID 以外は iOS に対応していません

※とはいえ多くの場合においてAndroid端末よりiPhoneの方がスペックが上だと思われませんが

- LCP : Chromium
- FID : Chromium、Firefox (with polyfill: Safari, Internet Explorer)
- CLS : Chromium

測定結果を評価する際には、この点を考慮していないと

iOS特有の原因でパフォーマンスが低下していることに気がつかない可能性が高いです。

引用 : GoogleChrome web-vitals

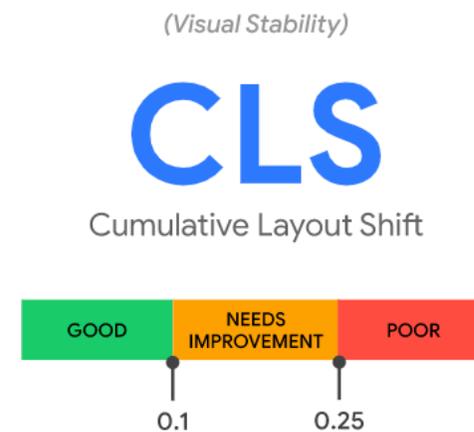
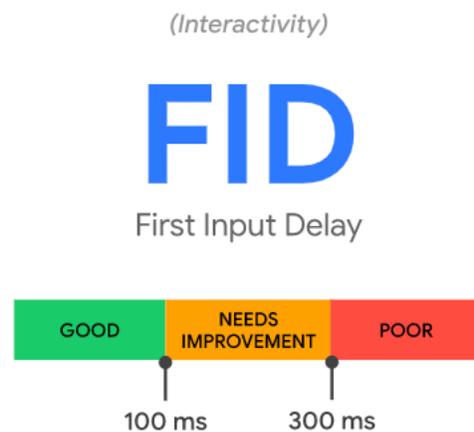
<https://github.com/GoogleChrome/web-vitals#browser-support>

CoreWebVitalsの注意点

CoreWebVitalsは定期的に更新されます

また CoreWebVitals は定期的に更新される予定であることにも注意が必要です、

具体的には指標のセットに追加・削除があったり、Good・Poorの閾値の変更の可能性があります。



引用：Chrome Developer Summit 2020 - Core Web Vitals の未来

<https://www.youtube.com/watch?v=iNfz9tg-wyg&t=89s>

ユーザーを代表するデバイス

アクセス解析だけではわからない

パフォーマンスの調査改善のために、多くの場合アクセス解析からターゲットとなるデバイスを決定していると思いますが、多くの場合アクセス解析だけでは全体像を把握することは難しいです。

なぜならターゲットユーザーのかなりの部分が、表示が遅すぎるという理由だけでサイトを放棄している（そして戻ってこない）可能性があり、そのようなデバイスがアクセス解析で最も人気のあるデバイスとして表示される可能性は低いからです。

ユーザーを代表するデバイス

代表的なユーザーのデバイス環境について

代表的なデバイスは、少なくとも 24か月経過し、200ドル以下のコストで、低速の3G、400msのRTT、400kbpsの転送で動作するAndroidデバイスですが、少し悲観的です。

もちろん、これはあなたの会社にとっては非常に異なるかもしれませんが、それはそこにいる大多数の顧客の十分に近い近似です。

※もちろん日本市場はこんな貧弱な環境ではないのですが、2020年以降にスマートフォンの価格が高騰しているの
で 2～3年前のモデルで 4G LTE が代表的なデバイスだと思います。

というところまでが第1章

正確には目次があるので第2章ですが

というところまでが「Front-End Performance Checklist」の第1章を超抜粋したものです、ちなみに全部で12章あり、下記のような内容になっています。

1. 目次
2. 準備：計画と指標（今回の説明はココ）
3. 現実的な目標の設定
4. 環境の定義
5. 資産の最適化
6. ビルドの最適化
7. 配信の最適化
8. ネットワーキングとHTTP/2
9. テストと監視
10. クイックウィン
11. 資料のダウンロード
12. 最後に

時間がない人向け

正確には目次があるので第2章ですが

「Front-End Performance Checklist」は最初の方は説明が多いので読み進めるのがキツイのですが後半になると実際の対策がメインになるので、かなり読みやすくなってきます。

また時間のない人のために10章のクイックウィンでは、とりあえずやっておくべきリストが提示されているので、まずそこから読んで試してみるのも良いアイデアです。

THANK YOU