Engineer

Core Web Vitals

実際の閲覧で読み込み、応答性、表示の安定性を測るGoogleの三つの指標

どのようなものか

Googleのweb.devドキュメントは、広範なWeb Vitalsの取り組みから三つの指標を選び、中核となる指標としています。読み込みを測るLargest Contentful Paint、応答性を測るInteraction to Next Paint、表示の安定性を測るCumulative Layout Shiftです。いずれも訪問者が気付くことを表し、すべてのページに適用でき、実際の訪問者のブラウザーで、フィールド測定できます。

Largest Contentful Paint（LCP）は、画面遷移の開始から、画像、テキストブロック、動画のうち、ビューポート内に見える最大のものが描画されるまでの時間です。2.5秒以下が良好、4秒を超えると不良です。

Interaction to Next Paint（INP）は、訪問中のすべてのクリック、タップ、キー入力を観測し、訪問者が感じる待ち時間全体を測ります。ハンドラーが始まるまでの遅延、その実行時間、次のフレームが描画されるまでの時間です。ページのINPは、そのうち最も遅いものですが、操作50回ごとに外れ値を一つ除きます。スクロール、ホバー、ズームは含めません。200ミリ秒以下が良好、500ミリ秒を超えると不良です。INPは2022年5月に実験的な指標として導入され、2024年3月12日にFirst Input Delayを置き換えました。First Input Delayが測っていたのは、最初の操作のハンドラーが始まるまでの遅延だけでした。

Cumulative Layout Shift（CLS）は、指示していないのに移動する可視要素を点数化します。各移動のスコアは、動く要素がビューポートを占める割合に、移動距離を掛けたものです。間隔が一秒未満、全体で最大五秒の移動をセッションウィンドウにまとめ、CLSには、各ウィンドウの合計スコアのうち、最も大きな値を使います。タップ、クリック、キー入力から500ミリ秒以内の移動は除外しますが、スクロールやドラッグでは除外しません。0.1以下が良好、0.25を超えると不良です。

判定は、モバイルとデスクトップを分け、ページ読み込みの75パーセンタイルで行います。その位置で三つすべての指標が良好な場合にだけ、ページは合格します。「良好」の境界を採用する際は、少なくともオリジンの一割が既に満たしていることが条件でした。

基盤には、Largest Contentful Paint、Layout Instability、Event Timingという、それぞれ独自の仕様を持つブラウザーAPIがあります。次の数行で、どのページでもCLSの基になる情報を確認できます。

```
new PerformanceObserver((list) => {
    for (const shift of list.getEntries()) {
        if (!shift.hadRecentInput) console.log(shift.value, shift.sources);
    }
}).observe({ type: "layout-shift", buffered: true });
```

これは個々の移動を記録するもので、指標そのものではありません。三つの指標をGoogleのツールと同じように報告するには、web.devは `web-vitals` ライブラリーを案内しています。

この言葉は、Lighthouseのパフォーマンススコアと混同しやすいものです。Lighthouseのスコアは、訪問者のいないラボで、一回の読み込みをシミュレーションした結果です。一方、Core Web Vitalsの評価は、実際の訪問の75パーセンタイルを表します。一方で良い成績でも、もう一方では悪い場合があります。

概念を判断に生かす

「Core Web Vitalsとは？」を実際の判断に生かしましょう。その概念を、開発または購入するシステムに結び付けて考えるお手伝いをします。 読み込みが速く、スマートフォンでも使え、お客様がどのブラウザーを使っていても同じように動く画面を提供します。

サービスを見る お問い合わせ →

用語の背後にある前提を検証する

解説とリンク先の実績を参考に、プロジェクトについての質問を整理してください。ツールを選ぶ前に、前提を一緒に検討できます。 フルスタックのGIS地図開発を指揮 — DBからブラウザーまでのフルスタック地図開発

フルスタックのGIS地図開発を指揮 お問い合わせ →

なぜ重要なのか

定義は、数値をどこから得るかまでは決めていません。ラボの推定値を基準にするか、負担を引き受けてフィールドデータを得るかを、サイト側が決める必要があります。

しきい値が対象とするのは、フィールドデータです。Chrome User Experience Reportは、実際のChrome利用者から匿名化した測定値を集め、サイトが何も追加しなくても、PageSpeed InsightsとSearch ConsoleのCore Web Vitalsレポートに提供します。ただし、ページやオリジンに十分な訪問者がいる場合に限ります。別の方法は、訪問者のブラウザー内のスクリプトから、各測定値をサーバーへ送ることです。これはプライバシーに関する判断であり、ページビューごとにかかる負担でもあります。

ラボデータには、そのどちらの負担もありません。Lighthouseは、変更を公開する前に、シミュレーションした端末とネットワークでページを読み込み、必要なだけ繰り返せます。測っているのは、実際には誰も行っていない訪問です。読み込み中のLCPとCLSは分かりますが、誰も操作しないためINPは得られません。web.devは、最も近いラボ指標としてTotal Blocking Timeを挙げますが、代わりにはならないとも述べています。

実用的なのは、両方を使い、役割を分けることです。ラボ測定には悪化した変更を見つけるための許容上限を設け、フィールドの数値は、得られる場合に限り、訪問者の実際の体験を判定する唯一の根拠とします。

実際の活用例

計画記録には、Core Web Vitals用のジョブを検討したものの、採用しなかったことが残っています。Chrome UX Reportから報告を得るにはAPIキーと十分なアクセス数が必要であり、ページのサイズには既に品質ゲートで上限を設けているためです。このサイト自身は、ブラウザー側の指標を収集しません。クレジットページに、実行可能なJavaScriptもブラウザー解析コードも配信しないと記載しています。クリックやキー入力を待たせるサイト配信のスクリプトはありませんが、レイアウトや描画が操作を遅らせる可能性はあります。同じページは、十分な訪問数があれば実際の訪問の測定値を示すPageSpeed Insightsを案内しています。

代わりに、このサイトが測るのはラボの値です。Lighthouseの設定は、固定したページ一覧を、モバイルの速度制限をシミュレーションした条件で各三回読み込みます。英語またはデンマーク語のページが、LCP 4,200ミリ秒超、First Contentful Paint 1,600ミリ秒超、CLS 0.02超、Total Blocking Time 100ミリ秒超のいずれかに達すると、不合格を報告します。ウクライナ語のページでは、描画時間の上限を少し広くしています。これらはフィールドのしきい値ではなく、ラボ測定の許容上限です。実行ごとのばらつきが最大299ミリ秒、最小80ミリ秒と測定された後に定めました。そのばらつきの範囲内に境界を置くと、偶然で合否が決まるためです。プロジェクトのノートは、LCPの大部分を、描画を妨げる一

つのスタイルシートによる描画遅延としています。コンテンツより先に届く容量は、既に削減していました。スタイルシートのバンドルは、base64フォントを含むと87 KB、除くと31 KBでした。測定には数分かかるため、品質ゲートの外で、必要に応じて実行します。

0.1という境界より五倍厳しいCLSの許容上限が、記録に残す価値のある事例を見つけました。あるカテゴリーページは0.107で不合格になりました。所有者がページを開いても問題は見えず、そのように伝えました。Lighthouseはパララックスで動く歯車を原因として挙げ、最初の診断記録でも、それを原因と考えていました。実際のホイール操作を使い、条件を制御して行ったテストでは、歯車を動かしても止めても0.2477で、その間、見出しと最後の段落の移動はゼロピクセルでした。原因はフッターです。ビューポートに固定され、奥付を開くと高さが変わります。固定要素はほかの要素の位置をずらせないにもかかわらず、指標はそのボックスの変化をビューポートに対してスコア化します。許容上限はそのまま維持しました。将来その数値で不合格にしないようにするなら、その一つのアドレスだけを例外にする計画です。

問題が起きるところ

ラボのレポートにINPがなく、ページは十分に応答すると考えてしまいます。 LighthouseはLCPとCLSを示し、Total Blocking Timeも低いため、本番でクリックが遅いと驚くことになります。読み込みのシミュレーションでは誰も操作しておらず、Total Blocking Timeは、その読み込み中にメインスレッドをブロックした時間だけを数えます。警告灯のようなものとして受け止め、ブラウザーのパフォーマンスツールで実際の操作時間を手動で測り、フィールドデータがあれば、それでINPを判断します。

ツールが指摘した要素が、原因ではないと分かります。 レポートがレイアウトのずれに関係するノードを示しても、そのノードを変えてスコアは変わらず、画面上の文字も動きません。発生元として示されるのはノードであり、原因そのものではありません。移動は、高さの変わる固定オーバーレイに由来する場合があります。これは移動として数えられ、スクロールによっても除外されません。疑わしいものを一つずつ無効にし、コンテンツが動くかを測ります。大きさのアニメーションには `transform` を使います。web.devは、レイアウトのずれを起こさないため、この方法を勧めています。

画像や後から届く要素の読み込みで、コンテンツが下へ跳びます。 読者は読んでいた場所を見失い、フィールドのCLSは0.1を超えます。web.devが挙げる典型的な原因は、大きさが不明な画像や動画、代替フォントより大きくまたは小さく表示されるウェブフォント、自分で大きさを変えるウィジェットです。このサイ

トの画像パーシャルと同様に、すべての画像に `width` と `height` 属性を付け、ブラウザーが領域を確保できるようにします。後から挿入するものの領域も、あらかじめ確保します。

良好な平均値が、不合格の評価を隠します。 ダッシュボードの平均LCPは2.5秒を十分に下回っていても、ページは改善が必要と評価されます。評価は端末の種類ごとの75パーセンタイルを使います。速いデスクトップを含む平均値では、モバイル訪問の遅い四分の一が見えなくなります。端末別に分けた75パーセンタイルを報告し、それを目標にします。

参考資料

- Web Vitals — 三つの指標、そのしきい値、報告するツールを説明します。
- Core Web Vitalsのしきい値の定義 — 75パーセンタイルを使う理由と、各境界の検証方法を説明します。
- Interaction to Next Paint — 操作に数えるものと、First Input Delayとの違いを説明します。
- Largest Contentful Paint API — 候補になる要素と、時間を取得する時点を説明します。
- Layout Instability API — レイアウトのずれのスコア、直前の入力、各移動の発生元を説明します。
- Event Timing API — 操作の遅延をスクリプトに公開する仕組みを説明します。

用語集

実際の制約に合わせて選ぶ

業務の流れ、既存のインターフェース、変更を必要とする利用者をお知らせください。検証可能な納品内容と、お客様のチームが保守を続けられる引き継ぎ内容を定められます。

料金を見る お問い合わせ →

用語を実務に結び付ける

「Core Web Vitalsとは？」がお客様の組織にどう当てはまるかをご相談ください。現在のシステムの具体例が一つあれば、話を始められます。

サービスを見る

welcome@engineer.company +45 50 10 61 84 デンマーク、コペンハーゲン

このページを持ち歩く

連絡先カード

心を込めて丁寧に制作： D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

著作権 © 2025 – 現在 · Engineer ApS · すべての権利を留保します。

印刷元： https://engineer.company/ja/glossary/core-web-vitals/