Engineer

Core Web Vitals

tre mätningar av hur det känns att använda en sida

Vad det är

Core Web Vitals är de tre mått som Googles dokumentation på web.dev plockar ut ur det bredare programmet Web Vitals och kallar kärnan: Largest Contentful Paint för laddning, Interaction to Next Paint för respons och Cumulative Layout Shift för visuell stabilitet. Namnet används nästan alltid på engelska, ofta förkortat CWV, och de tre måtten går under sina förkortningar LCP, INP och CLS. Vart och ett beskriver något som en besökare märker, gäller för varje sida och kan mätas i fält, alltså i riktiga besökares webbläsare.

Largest Contentful Paint (LCP) är tiden från att navigeringen börjar till att den största bilden, det största textblocket eller den största videon som syns i viewporten har ritats ut. Måttet är bra vid 2,5 sekunder eller mindre och dåligt över 4 sekunder.

Interaction to Next Paint (INP) bevakar varje klick, tryck och tangenttryckning under ett besök och räknar med hela den väntan som besökaren ser: fördröjningen innan händelsehanterarna startar, tiden de kör och tiden tills nästa bildruta har ritats. Sidans INP är den långsammaste av dem, med en avvikare bortplockad för var 50:e interaktion. Rullning, hovring och zoomning räknas inte. Måttet är bra vid 200 millisekunder eller mindre och dåligt över 500. INP ersatte First Input Delay den 12 mars 2024, efter att ha introducerats som ett experimentellt mått i maj 2022; First Input Delay mätte bara fördröjningen innan den första interaktionens hanterare började.

Cumulative Layout Shift (CLS) ger poäng åt synliga element som flyttar sig utan att någon har bett dem. Varje förskjutning får poäng efter hur stor andel av viewporten de rörliga elementen täcker, multiplicerat med hur långt de flyttar sig. Förskjutningar som ligger mindre än en sekund från varandra, upp till fem sekunder sammanlagt, bildar ett sessionsfönster, och CLS är summan för det

sämsta fönstret. Förskjutningar inom 500 millisekunder efter ett tryck, ett klick eller en tangenttryckning är ursäktade; rullning och dragning ursäktar ingenting. Måttet är bra vid 0,1 eller mindre och dåligt över 0,25.

Bedömningen görs vid 75:e percentilen av sidladdningarna, separat för mobil och dator, och en sida godkänns bara när alla tre är bra där. En gräns för ”bra” godtogs bara om minst en tiondel av alla ursprung (origins) redan klarade den.

Under detta ligger webbläsargränssnitt med egna specifikationer, API:erna Largest Contentful Paint, Layout Instability och Event Timing, och några få rader visar råmaterialet till CLS på vilken sida som helst:

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

Det loggar enskilda förskjutningar, inte måttet; för de tre så som Googles verktyg rapporterar dem hänvisar web.dev till biblioteket `web-vitals` .

Begreppet är lätt att blanda ihop med prestandapoängen i Lighthouse, som kommer från en enda simulerad laddning i ett labb, utan någon besökare i den. Bedömningen av Core Web Vitals beskriver riktiga besök vid 75:e percentilen, och en sida kan klara sig bra på det ena och dåligt på det andra.

Använd begreppet i ett beslut

Använd ”Vad är Core Web Vitals?” i ett konkret beslut. Vi kan knyta begreppet till systemet ni bygger eller köper. Gränssnitt som laddar snabbt, fungerar i mobilen och beter sig likadant i alla webbläsare dina kunder råkar använda.

Se våra tjänster Hör av er →

Granska antagandena bakom ordet

Använd förklaringen och det länkade arbetet för att formulera frågor till projektet. Vi kan undersöka antagandena tillsammans innan ett verktyg väljs. Styrde full stack-utvecklingen av GIS-kartor — kartutveckling i full stack, från databas till webbläsare

Styrde full stack-utvecklingen av GIS-kartor Hör av er →

Varför det spelar roll

Definitionerna lämnar öppet varifrån siffrorna kommer, och det är just det beslut som en webbplats måste fatta: att styra efter ersättningsmått från labbet eller att betala för fältdata.

Fältdata är det som gränsvärdena är skrivna för. Chrome User Experience Report, CrUX, samlar in anonymiserade mätningar från riktiga Chrome-användare och matar PageSpeed Insights och rapporten om Core Web Vitals i Search Console utan att webbplatsen lägger till något, men först när en sida eller ett ursprung har tillräckligt många besökare. Alternativet är ett skript i besökarens webbläsare som skickar varje mätning till en server: ett beslut om integritet och en kostnad som bärs vid varje sidvisning.

Labbdata kostar inget av det. Lighthouse laddar sidan på en simulerad enhet och ett simulerat nätverk, innan en ändring levereras och så ofta man vill. Det verktyget mäter är ett besök som ingen gjorde. Det ser LCP och CLS under laddningen, men eftersom ingen trycker på något finns det ingen INP att mäta; web.dev erbjuder Total Blocking Time som ersättningsmått i labbet och påpekar att det inte kan ersätta INP.

Den hållbara uppdelningen är att använda båda och hålla dem isär: siffror från labbet som budgetar som fångar en ändring till det sämre, och siffror från fältet, där de finns, som den enda domen över vad besökarna faktiskt får.

Så ser det ut i praktiken

Ett planeringsdokument visar att ett jobb för Core Web Vitals övervägdes och valdes bort: Chrome UX Report kräver en API-nyckel och tillräckligt med trafik för att rapportera, och sidorna hålls redan till storleksbudgetar i kvalitetsgrinden. Webbplatsen samlar inte själv in vitals i webbläsaren: kolofonsidan säger att den inte levererar körbar JavaScript eller analyskod till webbläsaren. Inget skript från webbplatsen kan fördröja ett klick eller en tangenttryckning, även om layout och rendering fortfarande kan fördröja en interaktion. Samma sida pekar på PageSpeed Insights, som visar mätningar från riktiga besök när en sida har tillräckligt många av dem.

Det webbplatsen mäter i stället är labbet. Dess konfiguration för Lighthouse laddar en fast lista med sidor tre gånger var, under simulerad mobil begränsning av nätverk och processor, och rapporterar ett fel när en av de engelska eller

danska sidorna passerar någon av dessa gränser: LCP över 4 200 millisekunder, First Contentful Paint över 1 600, CLS över 0,02 och Total Blocking Time över 100, med något vidare gränser för utritning på de ukrainska sidorna. Det här är budgetar och inte fältets gränsvärden, och de skrevs efter att spridningen mellan körningar hade mätts till 299 millisekunder som mest och 80 som minst, eftersom en gräns inom den spridningen klaras eller fälls av en slump. Projektets anteckningar tillskriver det mesta av LCP en fördröjning i renderingen bakom en enda stilmall som blockerar renderingen. Vikt som kommer före allt innehåll hade redan skurits ned: paketet med stilmallar vägde 87 KB med ett base64-kodat typsnitt inuti och 31 utan. Körningarna tar minuter, så de görs vid behov, utanför grinden.

CLS-budgeten, fem gånger strängare än gränsen 0,1, fångade ett fall som är värt att spara. En kategorisida föll på 0,107. Ägaren öppnade den, såg inget fel och sa det. Lighthouse hade pekat ut parallaxkugghjulen som boven, och den första skriftliga diagnosen trodde på det. Ett kontrollerat test med en riktig rörelse på mushjulet gav 0,2477 oavsett om kugghjulen rörde sig eller inte, medan rubriken och sista stycket flyttade sig noll pixlar hela tiden. Orsaken var sidfoten: den är fäst vid viewporten och ändrar höjd när dess kolofon fälls ut, och måttet räknar den rutan mot viewporten trots att ett fäst element inte kan skjuta undan något. Budgeten fick stå kvar där den stod; om siffran någon gång tystas är planen ett undantag för just den adressen.

Var det går fel

En labbrapport utelämnar INP, och sidan antas svara snabbt. Lighthouse visar LCP och CLS, Total Blocking Time är låg, och långsamma klick i produktion kommer som en överraskning. En simulerad laddning har ingen som interagerar, och Total Blocking Time räknar bara blockering av huvudtråden under just den laddningen. Läs det som en varningslampa, mät riktiga interaktioner för hand i webbläsarens prestandaverktyg och bedöm INP utifrån fältdata där sådana finns.

Elementet som ett verktyg pekar ut visar sig vara oskyldigt. Rapporten namnger en nod för en layoutförskjutning, en ändring av noden flyttar poängen inte alls, och texten på skärmen står still. Attribueringen namnger noder, inte orsaker; förskjutningen kan höra till ett fäst överlägg som ändrar höjd, vilket räknas som rörelse och inte ursäktas av rullning. Stäng av de misstänkta ett i taget, mät om innehållet flyttar sig och animera storlek med `transform`, som web.dev rekommenderar eftersom det inte orsakar någon layoutförskjutning.

Innehållet hoppar nedåt när bilder och sena tillägg laddas. Läsarna tappar bort sig, och CLS i fält stiger över 0,1. web.dev räknar upp de vanliga orsakerna: bilder och video utan känd storlek, webbtypsnitt som ritas större eller mindre än reservtypsnittet och widgetar som ändrar storlek på sig själva. Ge varje bild attributen `width` och `height` så att webbläsaren reserverar dess ruta, som mallarna för bilder på den här webbplatsen gör, och reservera plats för allt som infogas senare.

Ett friskt medelvärde döljer en underkänd bedömning. En instrumentpanel visar ett genomsnittligt LCP långt under 2,5 sekunder, men sidan bedöms ändå behöva förbättras. Bedömningen använder 75:e percentilen per enhetsklass, och en långsam fjärdedel av mobilbesöken försvinner i ett medelvärde som även räknar in snabba datorer. Rapportera och sikta på 75:e percentilen, uppdelat per enhet.

Läs mer

- Web Vitals — de tre måtten, deras gränsvärden och verktygen som rapporterar dem.
- Defining the Core Web Vitals metrics thresholds — varför 75:e percentilen, och hur varje gräns prövades.
- Interaction to Next Paint — vad som räknas som en interaktion och hur INP skiljer sig från First Input Delay.
- Largest Contentful Paint API — vilka element som är kandidater och när deras tid tas.
- Layout Instability API — poängen för en layoutförskjutning, nylig inmatning och källorna till varje förskjutning.
- Event Timing API — hur latensen för interaktioner görs tillgänglig för skript.

Ordlista

Välj utifrån era faktiska förutsättningar

Beskriv arbetsflödet, befintliga gränssnitt och användarna som behöver förändringen. Vi kan definiera en testbar leverans och en överlämning som gör att teamet kan underhålla lösningen.

Se vad det kostar Hör av er →

Gör begreppen praktiska

Fråga hur ”Vad är Core Web Vitals?” gäller i er organisation. Ett konkret exempel från ert nuvarande system räcker för att börja samtalet.

Se våra tjänster

welcome@engineer.company +45 50 10 61 84 Köpenhamn, Danmark

Ta med dig den här sidan

Visitkort

Omsorgsfullt skapad med kärlek av D.

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

Upphovsrätt © 2025 – nu · Engineer ApS · Alla rättigheter förbehållna.

Utskrivet från https://engineer.company/sv/glossary/core-web-vitals/