Verifying Mobile-First Indexing for Your Website

Verifying mobile-first indexing for your website is not about guessing. It is about reading signals Google leaves in its tools, logs, and HTTP exchanges. Most site owners assume it just happens. It does not. You need hard evidence that Googlebot Smartphone fetches your content, not the old desktop crawler, and that the indexed version matches what users see on a phone. One misplaced `Vary` header, one canonical pointing to an m-dot subdomain, and the entire signal collapses.

Mobile-first indexing became the default for all new websites after July 2019, and for most established ones by late 2020. Still, I routinely audit sites where half the important pages are crawled by Googlebot Desktop because of stale redirect rules or missing `last-modified` stamps. A 2023 study by Onely found that about 8% of sites in their enterprise sample still showed mixed crawling patterns, with desktop bot hits heavily outnumbering smartphone ones for key money pages. That directly hurts ranking, because now Google sees a version of your page you never intended.

You will need to look at two distinct layers: the crawl layer (which user-agent requested the page) and the index layer (which version is stored in the index). This article walks through exactly the signals you should pull, commands you should run, and mistakes that make verification useless.

What Mobile-First Indexing Actually Means at the Crawl Level

Mobile-first indexing forces Google to use the mobile version of your content for ranking. The most tangible effect for you: the primary crawler that reads your pages is no longer Googlebot/2.1 (desktop). It becomes Googlebot Smartphone, a mobile-capable bot that fetches pages with a modern smartphone viewport and respects the same `Vary: User-Agent` rules your CDN enforces. If your server responds differently to that user-agent—thinner DOM, lazy-loaded images, missing structured data—that shrunk content becomes your reality in the index.

Think of it like a camera lens. You once handed Google a high-resolution desktop frame. Now you must hand it the phone-cropped one. Verification means confirming the lens is actually in phone mode, not stuck on the old desktop setting, and that what the lens captures is what you intended.

Where to Find the Signal: Google’s Own Indicators

Start with **Search Console’s URL Inspection** tool. It reports the “Crawled as” user-agent. If you see “Googlebot smartphone” and the screenshot matches your mobile layout, the crawl layer is aligned. If you see “Googlebot desktop” for a page that should be mobile-first, something is broken—usually a redirect that treats smartphones as separate m-dot hosts. Below that, check the “Indexing” section: “Crawled – currently not indexed” or “Mobile usability” warnings are red flags that the mobile version differs too much.

The catch: URL Inspection shows only the last crawl snapshot. It does not tell you which bot consistently hits your server over weeks. For that, you need raw logs.

Also, visit the **Google Indexing API** not to submit, but to poll status programmatically. Using the `urlNotifications.get` endpoint with the URL’s canonical, you can see whether the last submission

(or crawling) was processed as `GOOGLEBOT_SMARTPHONE` in some cases. The official URL Inspection API is more reliable, though—it returns an explicit `crawledAs` field.

:::info The URL Inspection API quota is 2,000 calls per day per property and 600 queries per minute. It is not designed for bulk, but for spot-checking critical pages after a migration. :::

Automated Detection Using Logs and the URL Inspection API

A real check scans hundreds of URLs, not five. You pull server access logs, count Googlebot hits per user-agent, and compare the numbers. If 80% of crawls are still desktop Googlebot, mobile-first indexing is not truly applied across the site, regardless of what Search Console says for a single URL.

Below is a raw `grep` pipeline that separates the two bot types from yesterday's Nginx log:

```bash zcat access.log.1.gz \| grep -i "googlebot" \| awk '{print $12, $1}' \| sed 's/;//g' \| sort | uniq -c | sort -rn ```

The `$12` column usually contains the full user-agent string. A quick manual sift will show you the split. If you see `Googlebot/2.1` in the bulk, your old `robots.txt` rules or crawl budget signals may still be pointing Google toward a desktop-only path.

To confirm the indexed version, use the URL Inspection API with a short Python script. It returns a JSON blob with `mobileFriendlyResult`, `crawledAs`, and a live screenshot. The `crawledAs` value is `CRAWLED_AS_MOBILE` for smartphone crawls. This is the authoritative signal that Google itself acknowledges.

```python import requests import json API_KEY = 'YOUR_API_KEY' URL = 'https://example.com/crucial-page' endpoint = f'https://searchconsole.googleapis.com/v1/urlInspection/index:inspect?key={API_KEY}' payload = { "inspectionUrl": URL, "siteUrl": "sc-domain:example.com" } resp = requests.post(endpoint, json=payload) data = resp.json() print(data['inspectionResult']['indexStatusResult']['crawledAs']) # Expected: "CRAWLED_AS_MOBILE" if mobile-first is active ```

When the result instead shows `CRAWLED_AS_DESKTOP`, you have a configuration problem: perhaps a `Vary: User-Agent` that strips essential mobile headers, or a DNS-level redirect that funnels smartphone Googlebot to a different origin with unoptimized content. A common edge case: if the API returns `CRAWL_ERROR` or `NOT_FOUND` because the mobile version returns a soft 404 but the desktop version works fine—your server discriminates too aggressively.

:::warning Never assume a "mobile-friendly" test passes for all pages. Google's own mobile-friendly test uses a separate crawler, not Googlebot Smartphone. A page can pass that test and still be crawled by desktop bot in the wild. :::

The entire verification pipeline can be visualized as a decision loop across multiple URLs:

```mermaid flowchart LR A[Fetch URL with Googlebot Smartphone header] --> B{HTTP 200?} B -- No --> C[Debug server-side blocks] B -- Yes --> D[Check Vary: User-Agent header] D --> E[Call URL Inspection API] E --> F{crawledAs == MOBILE?} F -- Yes --> G[Record as verified] F -- No --> H[Diagnose redirect/canonical misconfiguration] H --> I[Fix and re-queue] I --> A ```

Common Misconceptions and Edge Cases

Many think that having a responsive design automatically triggers mobile-first indexing. It does not. A responsive site can still be crawled predominantly by desktop Googlebot if the server does not correctly signal cache variations or if old CDN rules serve different content to Googlebot Smartphone. I have seen a major e-commerce site where the `` tag on the desktop version pointed to itself unconditionally—so Google never realized the mobile version was even the canonical. The result: desktop-first indexing persisted for six months.

> Rule of thumb: If your site has separate URLs (m-dot or dynamic serving) and a redirect chain that treats Googlebot Smartphone as just another mobile browser, you're probably not getting mobile-first indexing for all pages.

Other gotchas: structured data. A site that lazy-loads JSON-LD only after a user gesture will appear perfectly fine on desktop but will be invisible to Googlebot Smartphone, which does not scroll. The missing structured data causes rich results to vanish overnight. Another edge case: `ServiceWorker` intercepts for mobile can rewrite responses in ways that pass the mobile-friendly test but fail the indexing criteria because the HTML delivered to Googlebot Smartphone is not the same as what the regular mobile HTML would be.

Real-World Examples and Common Failure Modes

A news publisher migrated from AMP to a fully responsive site in 2024. Six weeks passed, and traffic from Google Discover flatlined. Log analysis revealed that 92% of Googlebot hits were still desktop crawler. The culprit: a stale `sitemap.xml` cached on the CDN that contained only desktop URLs with `lastmod` dates older than the AMP versions, and the CDN's `Vary` header was set to `Accept-Encoding` only—ignoring `User-Agent`. Once they added `Vary: User-Agent` and purged the CDN, smartphone crawls jumped to 75% within four days.

A different scenario: a SaaS documentation site was built with client-side rendering and a thin mobile skeleton that loaded content via XHR. The mobile version appeared fine in the browser, but Googlebot Smartphone saw only a spinner. URL Inspection confirmed `crawledAs: MOBILE` but the HTML snapshot was empty. They had to implement dynamic rendering for that bot.

These failures are not rare. They happen when you treat mobile-first indexing as a checkbox and never actually pull the logs or script the API.

Five-Point Verification Checklist

- **Log audit**: Tally Googlebot Smartphone vs. desktop hits over the last 30 days; smartphone should be >80% for indexed URLs.
- **Canonical alignment**: Verify the canonical URL points to the mobile version, not to a separate desktop page.
- **Structured data parity**: Use the Rich Results Test with the mobile user-agent and confirm all required markup is present.
- **HTTP header validation**: Check that responses return `Vary: User-Agent` when content
```

differs, and avoid sending `X-Robots-Tag: noindex` only to Googlebot Smartphone.
- **API spot-checks**: Run the URL Inspection API on your top 50 revenue pages; all must show `crawledAs: CRAWLED_AS_MOBILE`; any desktop crawls get immediate attention.

Questions That Usually Come Up in Audits

If Search Console says “Mobile-first indexing enabled”, is the verification done? The notification covers the domain, not individual pages. I have seen cases where the notification was present but 20% of product pages still crawled by desktop bot because of an old CDN rule. The banner is a hint, not a verdict.

How quickly does Google switch to mobile-first after I fix a misconfiguration? No fixed SLA. After core issues are removed, Google re-evaluates during recrawls. Typically, for a site under 10,000 pages, a shift becomes visible in logs within 2-4 weeks. For larger sites, it can take months unless you actively use the Google Index Checker API to ping fresh indexing signals on changed URLs and accelerate discovery.

What if my site is purely desktop and we block all mobile devices? Then you are fighting Google’s fundamental system. You will probably get indexed by desktop bot, but the algorithm will increasingly treat your site as legacy because it cannot evaluate mobile page experience. Rankings will erode.

Where You Go After Confirming the Signal

Verification is not the end. Once you have proof that Googlebot Smartphone is your primary crawler across all key pages, shift focus to the actual content differences between mobile and desktop rendered versions: lazy-loaded images, the order of headings, the amount of visible text above the fold. The verification pipeline you built—logs, API, header checks—becomes your monitoring dashboard. Run it weekly, not yearly.

Further Reading

1. Google Search Central. "How Google Search Works." developers.google.com
2. Bing Webmaster. "Submit Sitemaps." bing.com/webmasters