Google’s Open Knowledge Format Could Work For Websites, Too via @sejournal, @slobodanmanic

Google published a format for turning a body of knowledge into a folder of linked markdown files. It was built for internal company data, and by accident, it solves a problem public websites have too. Right now, the most an AI agent gets from your website is a flat read of your pages, one at a time. This format builds a graph of how your ideas connect instead, so I tried it on my own website.

Google’s Open Knowledge Format Is A Directory Of Linked Markdown Files

On June 13, 2026, Google’s data team published the Open Knowledge Format, or OKF, a way to represent a body of knowledge as a directory of markdown files with a thin layer of YAML frontmatter. Each concept, a table, a metric, a runbook, an API, gets its own markdown document. A short block of YAML carries the queryable fields, type, title, description, resource, tags, and timestamp; the markdown body carries the explanation, and concepts link to each other with ordinary markdown links, which Google says turns the directory into “a graph of relationships.” There is no runtime, no SDK, no build step. Google describes a bundle in three phrases: “just markdown,” “just files,” “just YAML frontmatter.”

The target is internal company knowledge, the context Google says is “locked behind whichever surface created it,” and it

is early, v0.1, which Google calls “a starting point, not a finished standard.” Nothing in the announcement mentions public websites. That gap is what this piece is about.

On A Website, A Knowledge Graph Beats A Flat Page-Copy

The agent-readable version of your website, the one a model or a browser actually consumes, is flat. Serving each page as markdown, the way Cloudflare does at the network edge, is close to AMP for LLMs: a second, stripped copy of every page for a machine to read. It mirrors what you already have, page for page, and it drops the same thing every page-by-page copy drops, which is how the pages relate to each other.

A knowledge graph keeps that relationship layer. When your concepts link to each other, an agent not only learns what each one is, but it also learns how they sit relative to each other, which is most of what understanding a website actually means. Two pages can both mention a concept and never tell a machine that one is the framework underneath it and the other is the narrower goal beside it. A graph says it outright, in links the machine follows. OKF is an off-the-shelf way to build that graph: markdown, so it is cheap, and structured, so it carries the relations.

I Tried OKF On The No Hacks Website

I wrote an OKF bundle for the No Hacks website, one markdown file each for the brand, the host, Machine-First Architecture, the agentic web, Agent Experience Optimization, Answer Engine Optimization, llms.txt, and WebMCP. Each follows Google’s conventions, the YAML fields on top and a plain markdown body underneath. The work was mostly deciding which concepts mattered and how they connect, not writing the files.

One file, the concept for Machine-First Architecture, looks

like this:

```
---
type: framework
title: Machine-First Architecture
description: A framework for building websites whose full meaning is available to a machine reading them, with the human experience layered on top rather than the other way around.
resource: https://machinefirstarchitecture.com
tags: [Framework, Machine-First Architecture, Agentic Web]
timestamp: 2026-06-13
--- Machine-First Architecture is [Sani](./sani.md)'s framework for the [agentic web](./agentic-web.md). The core idea: build the content so a machine reading it gets the complete meaning, the facts, the structure, the relationships, and the human reading gets that same meaning with the design on top. This is why formats that strip a website to plain text, like markdown for agents and [llms.txt](./llms-txt.md), matter. Its capability side is [WebMCP](./webmcp.md), and its measurement side is [Agent Experience Optimization](./agent-experience-optimization.md).
```

Those bracketed links at the bottom are the graph. An agent following them learns that WebMCP sits under Machine-First Architecture, and llms.txt is the same kind of bet, which a flat copy of my pages never says out loud. Across the eight files, that is the whole structure: concepts, and the relationships between them.

A bundle like this is a second copy of what the website already says, and a second copy is a second thing to keep in sync. The moment the website changes, the bundle is wrong until you update it too. That tax is not unique to OKF: it is what every parallel machine-readable layer costs, an llms.txt file, a markdown mirror of your pages, a bundle like this one. The version an agent reads is only as accurate as your discipline in keeping it current.

Google did not build OKF for this. Its target is internal company knowledge, and nothing in its plan points at public

websites, so hosting a bundle for a visiting agent is off-label, and it may stay that way. The reader I made it for, an agent that fetches the bundle and follows the graph, might never show up. The reason to do it has to stand without that payoff, and it does: writing the bundle forced me to state plainly what No Hacks knows and how its ideas connect, and that surfaced gaps I would not have found writing another page. It is the same discipline as Machine-First Architecture, put your meaning in a form a machine can read, and you find where you were vague.

Where A Website Knowledge Graph Could Lead

None of what follows is a prediction. It is a direction, and it depends on agents actually reading website knowledge graphs, which today, none do. The shape is still worth seeing.

The identity file could grow into a knowledge graph. Today, llms.txt is a single line announcing who you are. A published bundle is the full version of that idea, a map of everything your website knows and how the parts connect, so the thin identity layer and the structured knowledge layer become one thing.

Agents could query that map instead of scraping your pages. An agent that pulls your bundle and follows its links gets a cleaner, relationship-aware read than one parsing your HTML one page at a time, and you get more say in how your own concepts are represented when an AI describes you.

The map could even become the canonical layer. The version a machine reads stops being a copy of your website and becomes the source, with the human pages as one rendering of it. That is the fully machine-first website the agentic web has been pointing at, reached through a side door Google opened for internal data.

Markdown Is Not New

John Gruber created Markdown in 2004, with Aaron Swartz as his beta-tester, and the whole design goal was readability: text you can read as-is, without rendering, that still converts cleanly to HTML. Two decades later, it runs GitHub, Reddit, much of the documentation you read, and the chat boxes of the AI tools themselves. It won by being legible without being rendered, which is the exact property that makes it easy for a machine to read.

I have written most of what I write in it for 15 years, since iA Writer became my main writing app in September 2011, so a week when the agent-readable web converges on markdown is familiar ground to me, not a new trick. The knowledge behind No Hacks (No Hacks OS project) has run the same way for months: markdown files with structured frontmatter, linked to each other, the shape a machine can read and traverse.

Machine-facing formats keep landing on that same ground, llms.txt, Cloudflare's markdown, and now OKF. Google itself is not of one mind about it. Its Search side called llms.txt "purely speculative" for ranking, its Chrome side added an llms.txt check to Lighthouse's agent-readiness audit, and its data team has now published OKF.

If you want to see where your website stands, it takes thirty seconds. Open your most important page and paste it into a plain-text editor, where the links collapse into plain words. Look at what is left and find anything that states how its ideas relate to the rest of your website, not that one page links to another, but the relationship itself. There is usually nothing, and that absence is what a knowledge graph fills, whether or not you ever touch OKF.

OKF is this week's news, and the substrate under it, plain text a machine can read, has been here since 2004. What Google added was a standard and a name.

More Resources:

This post was originally published on No Hacks.

Featured Image: Roman Samborskyi/Shutterstock

https://www.searchenginejournal.com/googles-open-knowledge-format-could-work-for-websites-too/579200/