

The Filing Is Free: Access, Cost and Provenance in Financial Research with AI Agents

Mosaic Finance
mosaic-finance.com

October 8, 2026

Abstract

Public companies file their accounts with regulators who publish them free of charge, yet the research built on those filings is sold back to readers at prices that run from free tiers to more than thirty thousand dollars a seat each year. We argue that the binding constraint on financial research is no longer access to the primary record but the cost of reading it: filings are long, their most informative detail sits in notes and sections that summaries leave out, and most readers meet them only through intermediaries. We describe how filing information travels from regulator to reader, what becomes hard to reach along the way, and what readers are expected to pay to recover it. We then report a pre registered benchmark in which two AI models, Sol 6.1 and Opus 5.5 at medium effort, answered questions about company filings with their native web tools, with Archivist, a retrieval service over the filings themselves, or with both. Pooled over the scored questions, Sol 6.1 used 1.93x fewer tokens with Archivist alone than with its native web tools (48% reduction; 95% CI 1.60 to 2.32; 90 questions, 3 runs each) and Opus 5.5 used 3.26x fewer (69% reduction; 95% CI 2.13 to 4.80; 30 questions, 1 run each). With both tools together, the configuration we deliver, the reduction ranged from 1.62x for Sol 6.1 to 3.57x for Opus 5.5. Accuracy stayed near the ceiling in every arm. With Archivist alone, every source the model read was a filing and no answer cited anything else. Of 400 forum and aggregator claims the agents were shown, 68.5% were confirmed by a filing and 3.5% contradicted; an audit of every contradicted row left 2.5% standing. The case for filings is not that the web is wrong. It is that a filing can be checked.

Keywords: financial disclosure, information processing costs, data access, retrieval, AI agents, provenance, benchmark

1 Introduction

Listed companies file their annual and interim reports with a regulator or an exchange, and those filings are published to anyone with a browser, free of charge [1, 2]. In principle the playing field is level. In practice it is not, because a filing is long, written to discharge a legal duty rather than to be read, and its most telling detail sits in the notes, the risk factors and the changes from one year to the next. Reading the record well takes time and expertise, and time and expertise are what the research industry sells.

The market has organized itself around that cost. Scrapers collect each filing within minutes, data vendors map its figures into fields, and terminals, screeners, news feeds and now AI tools

serve those fields to readers who pay for the convenience. Each hand off is reasonable on its own. Together they leave a reader holding a number without its caveats, and they leave whatever did not fit a field hard to reach. Throughout this paper, *buried* means exactly that: hard to reach, not concealed. Nothing we describe is hidden by the companies that file. It is buried under the cost of finding it again [3, 4].

Language models change the arithmetic of reading. An agent can open a filing, search it and quote it for a fraction of the time a person needs. But an agent left with a general web search reads what search ranks for it: the issuer’s own site, aggregators, news and forum threads, and only sometimes the filing. Its usage is metered in tokens, and every page it reads to find one number spends them. Whether the agent goes to the record or to the commentary about the record is therefore a question of cost as well as of provenance.

This paper makes three contributions. First, we describe the path from a filing to a reader and where detail becomes hard to reach along it (Sections 2 and 3). Second, we set out what readers are expected to pay for each tier of access, from free disclosure systems to institutional terminals, with dated public sources (Section 4). Third, after a short account of how Archivist, the retrieval service we build, reads the record (Section 5), we report a pre registered benchmark of two AI agents answering filings questions with their own web tools, with Archivist, or with both, measuring tokens, accuracy and what the agents actually read (Sections 6 and 7). We discuss what the results do and do not show in Sections 8 and 9.

We write as the builders of the system under test, which is a conflict of interest we state up front and again in the disclosures. Our framing is provenance: where a claim comes from, not whether its source is free of bias. A filing is an accountable source: the company filed it with a regulator and answers for it. It is also management’s own account, told in management’s own framing, and nothing in this paper says otherwise.

2 How filing information travels

2.1 The primary record is free

Securities rules require a listed company to file its annual and interim reports, and its material news, with a regulator or an exchange. The regulator or exchange publishes each filing on a public disclosure system, where anyone can search it and read it without paying [1, 2]. The details differ from one market to the next, but the essential point holds everywhere we looked: the reader pays nothing for the filing itself. The primary record is a public good. What is scarce is the attention needed to read it.

2.2 From filing to reader

Few readers meet a filing directly. Loughran and McDonald find that in 2003 to 2012 the average US firm’s 10-K was requested by investors only 28.4 times in the days right after it was filed [5]. Drake, Roulstone and Thornock show that demand for filings clusters around earnings announcements, restatements and mergers, and that the investors who do read them make prices more efficient with respect to earnings news [6]. Most readers instead receive the filing through a chain of intermediaries (Figure 1).

A company files with its regulator or exchange and the document is published. From that moment scrapers and data contractors race to collect it and extract its figures into their own

fields, each trying to be first. Terminals, screeners, news feeds and AI tools then serve those fields, or summaries of them. By the time a number reaches a reader it has usually passed through several hands, and the caveats that came with it rarely make the trip. None of the intermediaries is careless; each optimizes for speed and coverage, and a field can only hold what fits a field.

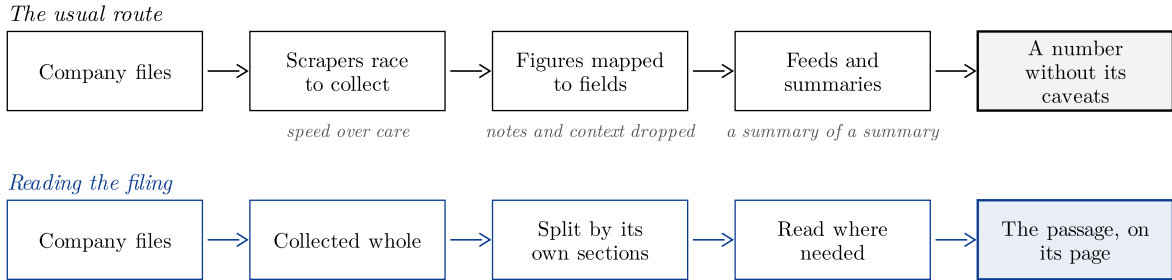


Figure 1: Two routes from a filing to a reader. The usual route passes through collection, extraction into fields and summaries, and detail falls off at each hand off; reading the filing itself keeps the passage on its page. A schematic, after Mosaic technical note MF-TN-003 [7].

3 What gets buried

Detail becomes hard to reach for three reasons, each documented in the accounting literature: filings have grown longer and harder to read, extracting information from them is costly, and the market prices costly information incompletely.

Length and boilerplate. Dyer, Lang and Stice-Lawrence trace 10-K text from 1996 to 2013 and find filings that became longer, more boilerplate and redundant, and less specific and readable, with fair value, internal control and risk factor disclosures driving much of the growth [8]. Loughran and McDonald show that the usual readability formulas misdescribe financial text and propose the size of the filing itself as a better proxy for how hard it is to read [9].

Processing costs. Blankespoor, deHaan and Marinovic review the evidence that monitoring, acquiring and analyzing disclosures is costly enough to turn public information into, in effect, costly private information, so that prices cannot be fully efficient [4]. Bloomfield’s incomplete revelation hypothesis puts the same point sharply: statistics that are more costly to extract from public data are less completely revealed in market prices [3].

Where it sits. Presentation matters even when the facts are the same. Hirshleifer and Teoh model how investors with limited attention respond differently to identical information depending on whether it is recognized or only disclosed, and how it is aggregated [10]. Michels finds a stronger price response to recognized than to footnote disclosed comparable events, with underreaction to disclosed events linked to processing costs [11]. And Cohen, Malloy and Nguyen show that changes in the language of 10-K and 10-Q filings from one year to the next predict returns, earnings and bankruptcies, without any announcement effect: investors simply miss them [12].

Table 1 lists places in a filing where this kind of detail sits. It is the part an extractor tends to leave behind, because it does not fit a field, and the part a summary tends to drop, because it is long. None of it is secret. All of it is in the filing, on a page anyone can open.

Table 1: Where detail sits in a filing, and what it can tell a reader. After MF-TN-003 [7].

Where it sits	What it can tell you
Hedging and derivatives notes	The currency, rate and commodity exposures the company pays to protect against
Commitments and contingencies	Obligations and claims that are not yet on the balance sheet
Legal proceedings	Disputes and regulatory actions, and how the company frames them
Risk factors, year over year	What was added, removed or quietly reworded since the last report
Debt notes and covenants	The terms that bind management, and how close it runs to them
Related party transactions	Money moving between the company and its insiders
Going concern and subsequent events	Doubts the auditors raised, and what changed after the balance sheet date

Reading the filing does not remove management’s framing. The notes, the risk factors and the discussion are written by management and reviewed by counsel; they carry management’s framing. What the filing offers is accountability: a statement the company made, under liability, on a date, that a reader can quote and check. That is the sense of provenance we use for the rest of this paper.

4 What access costs

If the filing is free, what do readers pay for? They pay for reading done on their behalf: collection, extraction, normalization, screening, news and, increasingly, synthesis by a language model. Burton-Taylor estimates that global spending on financial market data and analysis rose 6.5% to \$49.2 billion in 2025, as reported by FOW [13]. That figure covers far more than filings (prices, indices, reference data and news among them), but it shows the scale of what is sold around the primary record.

Table 2 sets out what a reader is expected to pay at each tier, with the date of each source. Institutional platforms do not publish prices. For them we cite third party reports and label each figure as *reported* (a figure from a client notice or procurement data) or *estimated* (a range from an analyst or procurement benchmark). These figures are indicative; contracts vary with seats, data entitlements and negotiation. We leave Mosaic’s own prices out of this paper.

Table 2: What access costs, by tier, in US dollars. Published prices are list prices on the vendor’s own page; reported and estimated figures come from third parties because the vendor publishes none. “As of” is the source’s date, or the date we read it where the page carries none.

Tier and product	Price	Basis	As of	Source
<i>The primary record</i>				
Regulator and exchange disclosure systems	Free	Published	read 2026-10-08	[1, 2]
<i>Retail research subscriptions</i>				
Stock Analysis Pro	\$79 a year, or \$9.99 a month	Published	read 2026-10-08	[14]
Seeking Alpha Premium	\$299 a year (list)	Published	2026	[15]
Koyfin Plus, Premium	\$39 and \$79 a month	Published	read 2026-10-08	[16]
<i>Data and filings APIs</i>				
Financial Modeling Prep, Starter to Ultimate	\$29 to \$139 a month	Published	read 2026-10-08	[17]
Massive (formerly Polygon.io), Stocks	\$29 to \$199 a month	Published	read 2026-10-08	[18]
sec-api.io, Personal and Business	\$49 and \$199 a month	Published	read 2026-10-08	[19]
EODHD, Fundamentals Data Feed	\$59.99 a month	Published	read 2026-10-08	[20]
Intrinio, Individual	\$150 a month	Published	read 2026-10-08	[21]
<i>General AI assistants</i>				
Claude Pro, Max	\$20 a month; Max from \$100 a month	Published	read 2026-10-08	[22, 23]
ChatGPT Plus	\$20 a month	Published	read 2026-10-08	[24]
<i>Institutional platforms (no published price)</i>				
Morningstar Direct	\$18,000 a year, first license	Reported	read 2026-10-08	[25]
AlphaSense	\$17,500 a year, median contract	Reported	2026-08-24	[26]
LSEG Workspace	\$10,000 to \$22,000 or more a seat a year	Estimated	2026-08-20	[27]
S&P Capital IQ Pro	\$12,000 to \$30,000 a user a year	Estimated	2026-03	[28]
FactSet	\$12,000 to \$40,000 a user a year	Estimated	2026	[29]
PitchBook	\$30,000 a year, median buyer	Reported	read 2026-10-08	[30]
Bloomberg Terminal	\$31,980 a year, single terminal	Reported	2024-10-09	[31]

Three things stand out. First, the range is wide: from nothing at the regulator to a reported \$31,980 a year for a single Bloomberg terminal, for 2025 and 2026 terms, according to a client notice reported by NeuGroup [31]. Second, the price rises with the amount of reading done for the reader, not with access to the record: every tier above the first resells the same public filings, enriched. Third, general AI assistants now sit at the low end of the range, at \$20 a month for the entry paid plans of Claude and ChatGPT [22, 24], which is why the efficiency of an agent’s

reading matters: a plan with a usage limit does more research when each answer costs fewer tokens.

Price also follows reach. An analyst with a terminal can pull a figure for every company in a sector in seconds; a reader with a browser can do the same only by opening every filing. The literature’s processing cost argument (Section 3) has a direct financial counterpart: the cheaper the tier, the more of the reading falls back on the reader. Democratizing financial research, in the sense we mean, is lowering the cost of that reading for everyone who already has free access to the record.

5 How Archivist reads the record

Archivist is a filing search and reading service. We describe it at the level of Mosaic’s public technical notes [7, 32]: what each stage is for, not how it is built.

Collect and label. Automated pipelines collect new filings every day from the regulators and exchanges that publish them in the United States, Canada and Turkey. Each document is labelled by issuer, form type, filing date and exchange, so that the same company is recognized across every form and year, and a search can narrow by company, filing type and date as precisely as a question demands. As of 23 September 2026 the searchable record held 14,164 active companies and 5,771,165 filings [7].

Convert and split. Every source is converted into one text format, including scanned pages and tables, and split along the document’s own sections rather than at arbitrary lengths, so that a passage keeps its heading and its place in the filing.

Index and retrieve. Passages are indexed both by meaning and by exact terms, and retrieval balances relevance with recency and variety, so that one long filing does not crowd out the rest and a stale filing does not outrank the current one.

Answer with provenance. Every result carries a passage permalink: a Mosaic link that opens the filing at the exact passage, highlighted. An agent connects through the Model Context Protocol (MCP) or a command line tool, searches with filters, reads a passage, a section or a whole filing, and cites what it read. Archivist does not change what a filing says. It changes how cheaply an agent can find the passage that says it.

6 Benchmark design

We fixed the design in a pre registration on 6 October 2026, before any scored run [33]. Five dated amendments followed: judges on our own plans (amendment 1), the first campaign limited to Codex (2), the provenance measures and the forum cross check (3), a smaller Claude Code replication (4), and key corrections with a light audit (5). Each is recorded with its date and reason in the public repository [34]. Table 3 summarizes the design.

Table 3: Benchmark design. Both agents ran headless from the command line with no shell or file tools, signed in to their vendors’ consumer plans; every arm had the same prompt, model and effort.

Item	Sol 6.1	Opus 5.5
Agent and model	Codex CLI 0.160.0, <code>gpt-6.1-sol</code>	Claude Code 2.1.289, <code>claude-opus-5-5</code>
Effort and plan	medium; ChatGPT plan	medium; Claude subscription
Archivist	archivist-cli 0.2.33 as an MCP server (8 tools)	
Questions	90 scored, 15 per type, plus 2 controls	30 scored, 5 per type, plus 2 controls
Runs	3 per question and arm; 828 runs on 7 October 2026	1 per question and arm; 96 runs on 8 October 2026
Arms	native web tools only; Archivist only; both together	
Grading	blinded judge per key fact; light audit (amendment 5)	

Questions. The set holds 92 questions: 15 in each of six types and 2 controls that need no research and measure the fixed overhead of each arm’s tools. The types are lookup (one to three facts from one filing), multi hop within one filing (two or more passages, one depending on another), several filings of one company, one metric across 5 to 10 filings of one company, breadth (the same fact across 7 to 10 companies) and reach (one company across 20 to 26 filings, sized to strain an agent’s context and turns). 33 of the 92 questions use Canadian or Turkish filings. Every answer key was built from the filings themselves, never from an agent answer: each fact names its filing, its passage and a quote checked against the stored text.

Agents and arms. We name the models Sol 6.1 and Opus 5.5 in the text; Table 3 names the agent products and versions. Each agent answered every question in three arms: its native web tools only, Archivist only, and both together, which is how we deliver Archivist. Every run had a fresh configuration and session and the same prompt: research the question with the available tools, be concise and cite sources. Sol 6.1 answered all 90 scored questions three times in each arm. Opus 5.5 answered a fixed stratified subset of 30 once in each arm, a smaller replication whose design was fixed after the Codex results were known.

Measures. Tokens are the agent’s own count of total tokens per run, input including cached input plus output. The token ratio is the web arm’s total tokens divided by the other arm’s, over the same questions; above 1 the other arm used fewer. Accuracy is the share of key facts an answer states correctly. Intervals are 95% paired bootstrap percentile intervals over questions (10,000 resamples), except for the forum cross check, which uses Wilson intervals over sampled claims.

Grading. A Codex judge graded every scored answer fact by fact against the key, blind to the arm: links, hosts and tool names were hidden. A Claude judge also graded the 90 scored Claude Code answers. Under the light audit of amendment 5, four facts the judge marked wrong were checked by hand against the filings; all four were errors in our keys, which we corrected and rescored. A blind second pass checked the other sampled facts. The audit covers 385 of the 7,047 judged Codex fact instances; the rest is judge only.

Hypotheses. We pre registered five hypotheses about tokens, accuracy and completion (H1 to H5) and, with amendment 3, a sixth about contamination by wrong web claims (H6). Table 9 reports each verdict. The provenance measures of amendment 3 were defined after we had seen

raw counts of forum results in the Codex logs, so every provenance figure over the main question set is exploratory.

7 Results

7.1 Tokens

Table 4 reports the average tokens consumed per question in each arm, pooled over the scored questions, with the change against the model’s native web tools in brackets. Both models used fewer tokens with Archivist than with their own web tools. The 95% bootstrap intervals for the factors are, for Sol 6.1, 1.93x (1.60 to 2.32) with Archivist alone and 1.62x (1.35 to 1.93) with both, and for Opus 5.5, 3.26x (2.13 to 4.80) and 3.57x (2.44 to 5.03).

Table 4: Average tokens consumed per question, by arm. In brackets: the change against the model’s native AI web tools. Medium effort, scored questions, completed runs.

Model (agent)	Questions x runs	Tokens consumed per question		
		Native AI web tools	Archivist alone	Both (web tools and Archivist)
Sol 6.1 (Codex CLI 0.160.0)	90 x 3	302,665	156,457 (48% reduction / 1.93x less)	186,831 (38% reduction / 1.62x less)
Opus 5.5 (Claude Code 2.1.289)	30 x 1	317,363	97,404 (69% reduction / 3.26x less)	88,805 (72% reduction / 3.57x less)

Tables 5 and 6 break the result down by question type, Opus 5.5 first. For Opus 5.5 the saving held on every type, largest on multi hop questions within one filing (8.94x with Archivist alone) and on breadth (5.26x); with five questions per type and one run, a single question can move a type. For Sol 6.1 the saving was concentrated where filings research is heaviest: breadth (2.88x) and reach (2.31x). On single lookups Sol 6.1 spent more tokens with Archivist than with its native web tools, and with both tools it also spent more on questions spanning several filings of one company.

Table 5: Opus 5.5 (Claude Code 2.1.289), 5 questions per type x 1 run: average tokens consumed per question, with the change against native AI web tools in brackets.

Question type	Tokens consumed per question		
	Native AI web tools	Archivist alone	Both (web tools and Archivist)
Lookup: 1 to 3 facts from one filing	74,245	29,290 (61% reduction / 2.53x less)	24,175 (67% reduction / 3.07x less)
Multi hop within one filing	439,856	49,193 (89% reduction / 8.94x less)	61,356 (86% reduction / 7.17x less)
Several filings of one company	157,965	39,330 (75% reduction / 4.02x less)	40,609 (74% reduction / 3.89x less)
One metric across 5 to 10 filings	333,619	125,283 (62% reduction / 2.66x less)	102,617 (69% reduction / 3.25x less)
Breadth: the same fact across 7 to 10 companies	510,694	97,058 (81% reduction / 5.26x less)	103,917 (80% reduction / 4.91x less)
Reach: one company across 20 to 26 filings	387,797	244,272 (37% reduction / 1.59x less)	200,156 (48% reduction / 1.94x less)

Table 6: Sol 6.1 (Codex CLI 0.160.0), 15 questions per type x 3 runs: average tokens consumed per question, with the change against native AI web tools in brackets.

Question type	Tokens consumed per question		
	Native AI web tools	Archivist alone	Both (web tools and Archivist)
Lookup: 1 to 3 facts from one filing	63,928	82,067 (28% increase / 1.28x more)	73,822 (15% increase / 1.15x more)
Multi hop within one filing	157,191	88,387 (44% reduction / 1.78x less)	104,432 (34% reduction / 1.51x less)
Several filings of one company	109,105	88,918 (19% reduction / 1.23x less)	119,330 (9% increase / 1.09x more)
One metric across 5 to 10 filings	225,155	184,633 (18% reduction / 1.22x less)	218,214 (3% reduction / 1.03x less)
Breadth: the same fact across 7 to 10 companies	598,310	207,516 (65% reduction / 2.88x less)	264,723 (56% reduction / 2.26x less)
Reach: one company across 20 to 26 filings	662,303	287,218 (57% reduction / 2.31x less)	340,464 (49% reduction / 1.95x less)

Figure 2 plots the same factors with their intervals. For Sol 6.1 the lookup interval lies wholly below 1 (0.78, 95% CI 0.62 to 0.99, with Archivist alone): Archivist’s tool list carries a fixed overhead before any research begins. On the two control questions Sol 6.1 used 39,982 tokens per run with Archivist alone against 26,282 with its web tools (3 runs each); Opus 5.5 used 14,545 against 19,043 (1 run each).

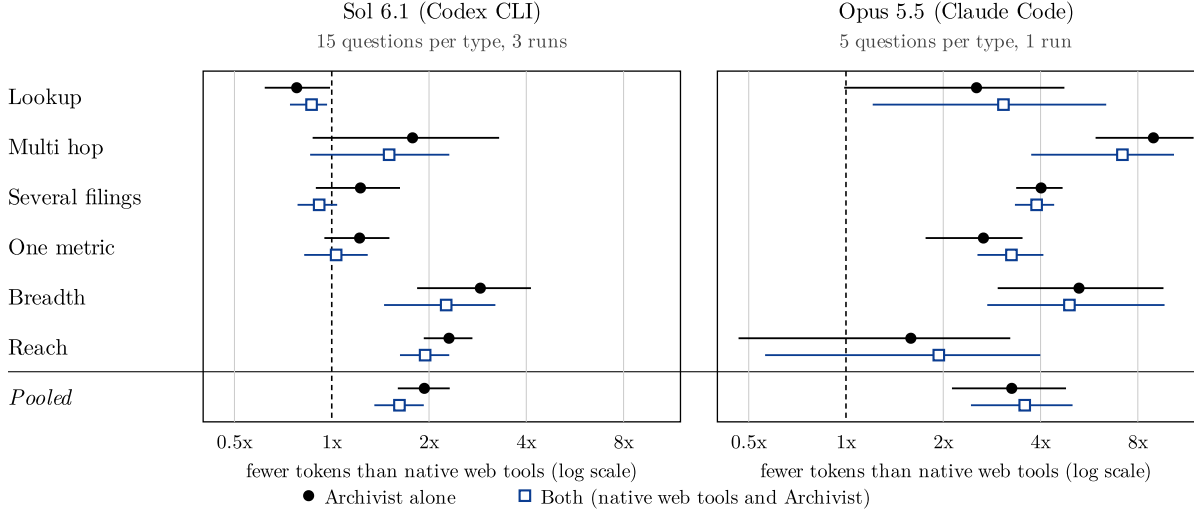


Figure 2: Native web tool tokens divided by the arm’s tokens, by question type and pooled, with 95% paired bootstrap intervals over questions (10,000 resamples). Right of the dashed line the arm used fewer tokens than the native web tools. Medium effort. Data: the 81.6 and 81.10 comparison tables in the public repository.

Why does Sol 6.1 save less with both tools than with Archivist alone? Because given both, it often did not use Archivist. Sol 6.1 called Archivist in 122 of 270 scored both arm runs (90 questions, 3 runs; an exploratory count) and answered from web search alone in the other 148. Opus 5.5 went the other way: it called Archivist in every scored both arm run, and its only web search in that arm was on a control question, so its both arm behaves like its Archivist arm. The configuration we deliver is the both arm, and its saving depends on how readily the agent reaches for the filing.

7.2 Accuracy

Accuracy sat at the ceiling in every arm. For Sol 6.1, question mean accuracy ran from 0.991 to 1.000 across arms and types (90 questions, 3 runs), and pooled over the 90 questions the difference against native web tools was 0.000 (95% CI -0.004 to $+0.004$) with Archivist alone and $+0.001$ (-0.003 to $+0.004$) with both. For Opus 5.5, pooled over 30 questions with 1 run, the differences were $+0.026$ (-0.006 to $+0.076$) with Archivist alone and $+0.016$ (-0.011 to $+0.060$) with both. Every one of these intervals includes zero. The benchmark therefore measures cost at near equal accuracy: pooled, it found no accuracy loss with Archivist for either model, within the pre registered margin of five points, and no accuracy gain either. We claim none.

Before the amendment 5 key corrections, the Opus 5.5 differences were $+0.033$ ($+0.001$ to $+0.082$) and $+0.024$ (0.000 to $+0.067$); one corrected key moved them, which shows how thin any difference at the ceiling is. The two judge families agreed on all 831 Claude Code facts on the original keys, and a check for judges favoring their own family’s answers found nothing: -0.0013 (95% CI -0.0042 to 0.0000 ; 22 questions both agents answered), though at ceiling accuracy the check had little room to.

7.3 Reach

Reach questions follow one company across 20 to 26 filings. We sized them to strain context and turn limits and pre registered that web search would fail tasks Archivist completes (H5). It did not happen: Sol 6.1 with web search alone completed all 45 reach runs, as did both Archivist arms, and no run in any arm timed out, hit a turn limit or compacted its context. What Archivist did on reach was make the same work cheaper: 2.31x fewer tokens alone (95% CI 1.93 to 2.72) and 1.95x with both (1.63 to 2.31), for Sol 6.1 over 15 questions and 3 runs. Going farther, in this benchmark, means the same work for fewer tokens, not tasks the web could not finish. Opus 5.5 had one incomplete run among its 90 scored runs: a web arm answer to a multi hop question that gave 2 of 5 facts after 1.02 million tokens.

7.4 What agents read and cite

With Archivist alone, every source the model read was a filing and no answer cited anything else. That holds for both models in the Archivist alone arm, which has no web tools: Sol 6.1 over 90 questions and 3 runs, Opus 5.5 over 30 questions and 1 run, with citations measured on the fact claims a grader extracted from each answer (for Opus 5.5 one answer yielded no claims and is excluded). The interesting comparison is between the web arm and the both arm, where the agent chooses (Figure 3 and Table 7). These measures are exploratory.

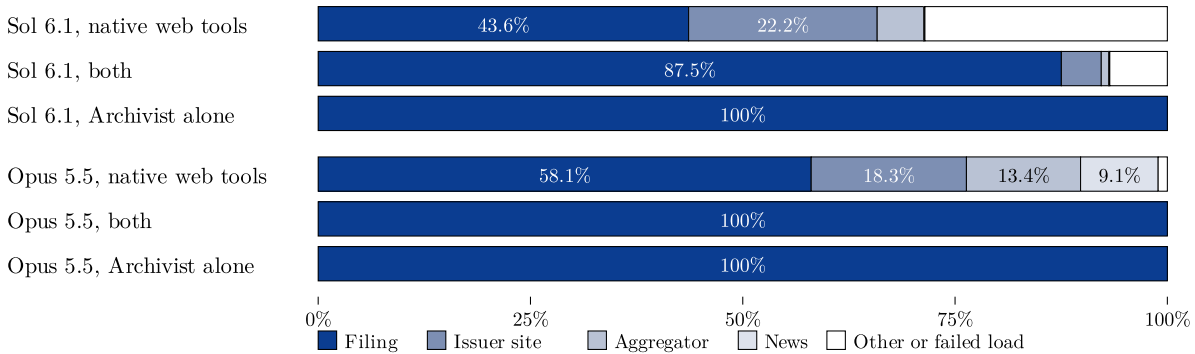


Figure 3: What the agent chose to read: pages it opened plus Archivist results, by source category, pooled over scored runs. Pages that failed to load count as other. Sol 6.1: 90 questions x 3 runs; Opus 5.5: 30 questions x 1 run. Exploratory. Data: the 81.9 and 81.10 exposure tables in the public repository.

With web search alone, filings were 43.6% of what Sol 6.1 chose to read (95% CI 36.1 to 52.5) and 58.1% for Opus 5.5 (36.9 to 77.0). With Archivist added, they were 87.5% for Sol 6.1 (80.9 to 92.2) and 100% for Opus 5.5. Search put forum content in front of the agents often: forum results appeared in 90.7% of Sol 6.1 web arm runs (95% CI 84.8 to 95.9; 270 runs), always as search results. In 828 Sol 6.1 runs no agent opened a forum page.

Table 7: Where the facts in the answers came from, by arm. A matcher labels a claim *filing* when its usable figures and names occur in a filing the same run read (strict rule) or when its figures do (figures only rule). Exploratory; 95% CIs in brackets. Sol 6.1: 90 questions x 3 runs; Opus 5.5: 30 questions x 1 run.

Model and arm	Claims	From a filing, strict	From a filing, figures only	Cites a non filing source
Sol 6.1, native web tools	2,624	46.2% (38.4 to 54.3)	68.9% (62.6 to 75.0)	28.5% (22.1 to 35.0)
Sol 6.1, both	2,577	52.3% (43.6 to 61.3)	84.5% (79.8 to 88.4)	8.6% (5.2 to 13.1)
Sol 6.1, Archivist alone	2,705	53.9% (45.5 to 62.3)	83.2% (77.6 to 87.9)	0%
Opus 5.5, native web tools	500	17.6% (10.1 to 26.8)	28.6% (17.6 to 41.0)	23.0% (12.4 to 35.1)
Opus 5.5, both	512	44.0% (31.7 to 58.5)	83.0% (78.8 to 86.6)	0%
Opus 5.5, Archivist alone	457	41.8% (30.4 to 55.6)	81.4% (75.5 to 85.8)	0%

Claims that cite a source other than a filing fell from 28.5% of Sol 6.1 web arm claims to 8.6% with both tools and 0% with Archivist alone; most of the web arm citations were issuer websites (651 of 749). The matcher errs both ways. It can miss computed or restated figures, which may explain much of what the Archivist arms leave unattributed under the strict rule; we have not audited why. It can also match by chance, because a value counts as read when any number in the read text rounds to it, and the web arm reads far more text per run.

7.5 Do the forums have it wrong?

Because search showed forum and aggregator content so often, we checked it. From the forum and aggregator texts the Sol 6.1 runs were shown, we sampled 400 claims and checked each against the filings with an Archivist only checker (Table 8). Most of it held up. Of the 400 claims, 68.5% were confirmed by a filing and 3.5% contradicted; 12.8% could not be checked against a filing (opinion, forecast, estimate or rumor) and 15.3% were off target. Our orchestrating agent then audited every contradicted row and a sample of confirmed rows against the filings, an agent audit we delegated, not a human one. All 10 sampled confirmed rows stood; of the 14 contradicted rows, 10 stood, which is 2.5% of the 400 (Wilson 95% CI 1.4 to 4.5) and 3.5% of the 286 checkable claims (1.9 to 6.3). The other 264 confirmed rows were not audited. These rates describe content that search ranked for our questions, not forums in general.

Table 8: Forum and aggregator claims shown to Sol 6.1, checked against the filings, before the audit of contradicted rows. Wilson 95% intervals in brackets. Scope: content search ranked for these questions.

Source	Claims	Confirmed	Contradicted	Not checkable	Off target
All	400	68.5% (63.8 to 72.9)	3.5% (2.1 to 5.8)	12.8% (9.8 to 16.4)	15.3% (12.1 to 19.1)
Aggregator	279	76.0% (70.6 to 80.6)	3.2% (1.7 to 6.0)	6.5% (4.1 to 10.0)	14.3% (10.7 to 18.9)
Forum	121	51.2% (42.4 to 60.0)	4.1% (1.8 to 9.3)	27.3% (20.1 to 35.8)	17.4% (11.6 to 25.1)

Forum posts were less often checkable than aggregator pages, but when they were checkable they were usually right: 62 of 67 checkable forum claims were confirmed. The case for filings is not that the web is wrong. Most of what we checked was right. The case is that a filing can be checked.

7.6 Contamination (H6)

To test whether reading filings protects an agent from wrong web claims, we built 17 questions phrased as users ask them: 12 traps where a web narrative differs from the filing, drawn from contradicted cross check rows, and 5 controls. We pre registered that Archivist would reduce how often agents repeat the wrong figure (H6). No agent in any arm adopted a trap value: Sol 6.1 in 153 runs (117 trap judgements) and Opus 5.5 on 6 trap questions with 1 run each. H6 is not supported, and the test shows nothing about protection: with no arm misled, there was nothing for Archivist to prevent. On these single figure questions Sol 6.1 used about 1.9x the web arm’s tokens with Archivist alone (web over Archivist 0.52, 95% CI 0.45 to 0.59), consistent with its lookup result, while Opus 5.5 used fewer (3.72x fewer, 95% CI 1.62 to 6.71; 6 trap and 2 control questions, 1 run each).

Table 9: Pre registered hypotheses and verdicts, applied by hand to the audited figures. Sol 6.1: 15 questions per type x 3 runs. Opus 5.5: a replication with 5 questions per type x 1 run, where a single question can move a verdict.

Hypothesis	Sol 6.1	Opus 5.5
H1: one metric across many filings, at least 2x fewer tokens	Not supported: 1.22x (0.95 to 1.51)	Partly: 2.66x (1.76 to 3.52)
H2: both tools as accurate as the best single arm, and cheaper than web	Supported pooled: 1.62x (1.35 to 1.93)	Supported pooled: 3.57x (2.44 to 5.03)
H3: lookups no worse than web	Not supported: 0.78x (0.62 to 0.99)	Supported: 2.53x (0.99 to 4.74)
H4: breadth at least equal	Supported: 2.88x (1.84 to 4.13)	Not supported: one wrong fact
H5: reach completes tasks web fails	Not supported: web completed every task	Partly, Archivist alone, on one fact
H6: fewer wrong web claims repeated	Not supported: no arm adopted a trap	Not supported: no arm adopted a trap

8 Discussion

Where Archivist did not win. We report the losses as plainly as the gains. For Sol 6.1, single lookups cost more with Archivist than with web search, because Archivist’s tool list costs tokens before any research begins. The lane we predicted, one metric across many filings, gave 1.22x against a pre registered 2x. With both tools, Sol 6.1 spent more tokens than with Archivist alone in every type but lookup, and more than web search alone on questions spanning several filings of one company. On the trap questions Sol 6.1 spent about 1.9x the web arm’s tokens with Archivist alone. For Opus 5.5, both Archivist arms reported one mine’s production instead of the company total on one breadth question, which cost H4 on that type despite 5.26x fewer tokens, and on multi hop questions the both arm got one fact wrong that the Archivist arm got right. With one run per question these may be noise; they are in the record.

What the results support. Across two models from two vendors, reading the filing through a retrieval service took fewer tokens than reading the web about the filing, pooled over the question set, at accuracy we could not tell apart. The saving is largest exactly where the processing cost argument of Section 3 predicts it: where a reader must reach across many companies or many filings, the work that terminals and data vendors charge for. When the agent used Archivist, what it read was mostly filings, and its answers cited fewer sources nobody is accountable for.

What they do not support. They do not show that filings make answers more accurate: accuracy was at the ceiling. They do not show that Archivist protects against misleading web content: no agent was misled. And they do not show that the web is unreliable: most forum and aggregator claims we checked were confirmed. The argument for reading the record is narrower and, we think, sturdier. A claim that traces to a filing can be checked against a statement the company made under liability, on a date; a claim that traces to a forum post has no one who answers for it. Provenance is not truth, and a filing is management’s own account. But it is an account someone answers for.

Democratizing the reading. The filing has been free for decades. What the research industry sells, at the prices in Table 2, is reading done on the reader’s behalf. General AI assistants put a capable reader in front of anyone with a \$20 plan, but a plan has limits, and an agent that reads the web about a filing spends those limits on pages nobody is accountable for. Pointing the agent at the record itself is a small change with a direct effect on cost: in our benchmark, the same questions for 52% of the web arm’s tokens with Sol 6.1 and 31% with Opus 5.5, pooled, with Archivist alone.

9 Limitations

Ceiling accuracy. Every arm scored near 1.0, so accuracy comparisons can show no loss within the five point margin, pooled, but cannot show a gain. Tokens are the measure that separates the arms.

Tokens are not a plan’s meter. Our totals include cached input. ChatGPT and Claude plans count usage their own way, which we cannot see, so a token ratio estimates plan headroom; it does not measure it.

One judge family for Sol 6.1. A Codex judge graded the Codex answers, so judge and agent share a model family. Arms were graded blind. For Opus 5.5 two judge families agreed on every fact, but at ceiling accuracy that check had little room to find a difference.

A light audit. The audit covers 385 of 7,047 judged Codex fact instances; the rest is judge only. The cross check audit was delegated to an agent, and only 10 of 274 confirmed rows were audited.

A smaller replication. Opus 5.5 answered 5 questions per type once, against 15 questions three times for Sol 6.1. Its design was fixed after the Codex results were known, and it ran a day later. Claude Code’s web fetch tool hands the model an extract of each page, written by a model, rather than the page itself.

Exploratory provenance. The source and citation measures were defined after we had seen raw forum counts, and they rely on matching values in text, which can miss computed or restated figures.

Our questions, our coverage, our product. We wrote the questions inside Mosaic’s coverage (filings from the United States, Canada and Turkey, and expanding); outside it Archivist has nothing to read. We build the system under test. We did not test the Claude or ChatGPT apps, other models or other effort levels, and the answers made too few evaluative statements to measure framing. The prices in Section 4 are list prices, reports and estimates as of the dates shown, not quotes.

10 Conclusion

The filing is free. Reading it is what costs, and the research market is priced on that cost, from free disclosure systems to terminals reported above thirty thousand dollars a seat. What becomes hard to reach along the way is not hidden; it sits in the notes and the changes that summaries leave out. In a pre registered benchmark, two AI models answered filings questions with fewer tokens when they could read the filings directly, pooled, at accuracy we could not tell apart, and when they did, what they read and cited was the filing. That does not free the answers of management’s framing, and it does not make the web wrong. It makes them checkable, at a lower cost, for anyone with access to an agent.

Disclosures

This paper is written by Mosaic Finance, which builds and sells Archivist, the system under test. The benchmark was designed, run and graded by Mosaic, on Mosaic’s own ChatGPT and Claude plans; no vendor funded or reviewed it. Codex and ChatGPT are products of OpenAI. Claude and Claude Code are products of Anthropic. Mosaic is not affiliated with, sponsored by or endorsed by OpenAI, Anthropic or any other company named in this paper. Product and company names are used only to identify what was tested or priced. Prices are as reported in the cited sources on the dates shown. Nothing in this paper is investment advice.

Data and code availability

The pre registration with its five amendments, the 92 questions with answer keys, the harness, every run’s tokens, tool calls and grades, the answers and the redacted run logs are public in the repository [mosaicss/archivist-benchmark](https://github.com/mosaicss/archivist-benchmark) on GitHub [34], with steps to reproduce. Code is licensed under Apache 2.0 and data and text under CC BY 4.0. Filings in the answer keys are

named by their exchange document identifiers and linked only through Mosaic passage permalinks.

References

- [1] U.S. Securities and Exchange Commission. Accessing EDGAR Data. Last reviewed 26 June 2024; read 8 October 2026.
- [2] Kamuyu Aydınlatma Platformu (KAP). General Information. Read 8 October 2026.
- [3] R. J. Bloomfield. The “Incomplete Revelation Hypothesis” and Financial Reporting. *Accounting Horizons* 16(3):233-243, 2002. doi:10.2308/acch.2002.16.3.233.
- [4] E. Blankespoor, E. deHaan and I. Marinovic. Disclosure processing costs, investors’ information choice, and equity market outcomes: A review. *Journal of Accounting and Economics* 70(2-3):101344, 2020. doi:10.1016/j.jacceco.2020.101344.
- [5] T. Loughran and B. McDonald. The Use of EDGAR Filings by Investors. *Journal of Behavioral Finance* 18(2):231-248, 2017. doi:10.1080/15427560.2017.1308945.
- [6] M. S. Drake, D. T. Roulstone and J. R. Thornock. The Determinants and Consequences of Information Acquisition via EDGAR. *Contemporary Accounting Research* 32(3):1128-1161, 2015. doi:10.1111/1911-3846.12119.
- [7] Mosaic Finance. How financial information travels, and what falls off along the way. Technical note MF-TN-003, revision 2026-09-28. <https://mosaic-finance.com/sources/how-filings-travel.pdf>.
- [8] T. Dyer, M. Lang and L. Stice-Lawrence. The evolution of 10-K textual disclosure: Evidence from Latent Dirichlet Allocation. *Journal of Accounting and Economics* 64(2-3):221-245, 2017. doi:10.1016/j.jacceco.2017.07.002.
- [9] T. Loughran and B. McDonald. Measuring Readability in Financial Disclosures. *Journal of Finance* 69(4):1643-1671, 2014. doi:10.1111/jofi.12162.
- [10] D. Hirshleifer and S. H. Teoh. Limited attention, information disclosure, and financial reporting. *Journal of Accounting and Economics* 36(1-3):337-386, 2003. doi:10.1016/j.jacceco.2003.10.002.
- [11] J. Michels. Disclosure Versus Recognition: Inferences from Subsequent Events. *Journal of Accounting Research* 55(1):3-34, 2017. doi:10.1111/1475-679X.12128.
- [12] L. Cohen, C. Malloy and Q. Nguyen. Lazy Prices. *Journal of Finance* 75(3):1371-1415, 2020. doi:10.1111/jofi.12885.
- [13] Z. Jakubowski. Market data spend hits \$49.2bn as AI, volatility and algos drive growth. FOW, 8 April 2026, reporting Burton-Taylor (TP ICAP). <https://www.fow.com/insights/market-data-spend-hits-usd49-2bn-as-ai-volatility-and-algos-drive-growth>.
- [14] Stock Analysis. Stock Analysis Pro. <https://stockanalysis.com/pro/>. Read 8 October 2026.
- [15] Seeking Alpha. Premium subscription page, 2026. https://subscriptions.seekingalpha.com/lp_premium_bundle_sitewide_sale_2026_ppc/. Read 8 October 2026.
- [16] Koyfin. Pricing. <https://www.koyfin.com/pricing/>. Read 8 October 2026.
- [17] Financial Modeling Prep. Pricing. <https://site.financialmodelingprep.com/developer/docs/pricing>. Read 8 October 2026.
- [18] Massive. Pricing. <https://massive.com/pricing>. Read 8 October 2026.
- [19] sec-api.io. Pricing. <https://sec-api.io/pricing>. Read 8 October 2026.
- [20] EODHD. Pricing. <https://eodhd.com/pricing>. Read 8 October 2026.
- [21] Intrinio. Pricing. <https://intrinio.com/pricing>. Read 8 October 2026.
- [22] Anthropic. Claude pricing. <https://claude.com/pricing>. Read 8 October 2026.

- [23] Anthropic. What is the Max plan? Claude Help Center. <https://support.claude.com/en/articles/11049741-what-is-the-max-plan>. Read 8 October 2026.
- [24] OpenAI. What is ChatGPT Plus? OpenAI Help Center. <https://help.openai.com/en/articles/6950777-what-is-chatgpt-plus>. Read 8 October 2026.
- [25] FitGap. Morningstar Direct. <https://us.fitgap.com/products/morningstar-direct>. Read 8 October 2026.
- [26] AllMind. AlphaSense pricing 2026, citing Vendr contract data. 24 August 2026. <https://allmind.ai/research/alphasense-pricing-2026>.
- [27] FlashAlpha. FlashAlpha vs LSEG Workspace (Refinitiv Eikon) 2026. 20 August 2026; a competitor’s estimate. <https://flashalpha.com/articles/flashalpha-vs-lseg-refinitiv-workspace-options-data>.
- [28] CostBench. S&P Capital IQ pricing, as of March 2026. <https://costbench.com/software/financial-data-terminals/sp-capital-iq/>.
- [29] Multiples. FactSet alternatives, 2026. <https://multiples.vc/compare/factset-alternatives>. Procurement benchmarks elsewhere give lower ranges.
- [30] Vendr. PitchBook, marketplace pricing data. <https://www.vendr.com/marketplace/pitchbook>. Read 8 October 2026.
- [31] NeuGroup. Bloomberg Terminals: How Much More You’ll Pay Next Year. 9 October 2024, reporting a Bloomberg client notice. <https://connect.neugroup.com/public/blogs/bloomberg-terminals-how-much-more-youll-pay-next-year>.
- [32] Mosaic Finance. The groundwork: from exchange feeds to an answer you can check. Technical note MF-TN-001, revision 2026-09-28. <https://mosaic-finance.com/sources/the-groundwork.pdf>.
- [33] Mosaic Finance. Pre registration: Archivist versus native agent web tools. 6 October 2026, with amendments 1 to 5 (6 to 8 October 2026). Published in the [mosaicss/archivist-benchmark](#) repository, [preregistration/preregistration.md](#).
- [34] Mosaic Finance. Archivist benchmark: Codex and Claude Code, own web tools versus Archivist. <https://github.com/mosaicss/archivist-benchmark>.