Where the Data Lives: Chat Interfaces, Word Add-Ins, and Harnesses

From: John A. Kelly

Date: August 5, 2026

Re: Where the data lives: chat interfaces, Word add-ins, and harnesses under the policy

There are three distinct ways a firm actually works with AI, and they have different data footprints. The policy's input tiers (Section 4) say what may go into a tool; this document says where it goes once it does, for each way of working, and which policy controls carry the load in each. Vendor terms are stated generically as of August 2026; the Tool List records the specifics per tool.

1. Four things that get conflated as "retention"

Sorting these out first makes the rest of the analysis short. When people ask "does the vendor keep my data," there are four separate questions:

1. **Training.** Does the vendor use your inputs to train its models? Consumer tiers commonly do by default; team, business, and enterprise tiers do not. This is the non-negotiable: the policy prohibits client work on any account where training is on (Section 3).
2. **Conversation storage.** Does the vendor store your chats and uploads on its servers? On web interfaces, yes by default and indefinitely until deleted; this is a feature (your history), not a leak, but it is client data sitting on a third party's servers under the vendor's terms.
3. **Memory and personalization.** Does the tool distill what it learns about you across sessions into a persistent profile it consults later? This is the cross-matter contamination vector (policy Section 4): it is a feature toggle, not a contract term, and the policy requires it off or matter-scoped.
4. **Safety retention.** Even where a vendor contractually agrees to zero data retention, at least one major vendor now requires a 30 day safety window for its newest frontier models, longer for content that trips trust-and-safety flags. The window follows the model, not the account or the modality: it applies only to traffic actually routed to a covered frontier model, and requests to non-covered models under ZDR terms are not persisted at all. Well-built platforms surface this as an explicit administrator opt-in per model; the policy treats it as a known condition to be recorded on the Tool List, managed by model choice, and considered in client communication (Section 7).

"Zero data retention" addresses items 1 and 2 by contract. It does not touch item 3, which is a setting, and as of this writing it does not defeat item 4. A firm that gets ZDR terms and leaves memory on has solved the contract and kept the contamination.

2. Way one: chat interfaces on the web

What it is. A browser or desktop app session with a general-purpose tool (a Claude or ChatGPT window) or a legal tool's chat surface. Type or paste, attach documents, read the answer.

Where the data lives. Everything you type or attach transits to the vendor's cloud and lands in three places: the inference pipeline (transient), the conversation store (persistent until deleted), and, if enabled, the memory store (persistent and cross-session). On a consumer account all of it may also feed training. Deleted conversations typically purge on a delay measured in weeks.

The exposure profile. Highest conversation-storage footprint and the only modality where the memory feature is routinely on by default. The mitigations are account posture (team tier or better, training off), memory off, and the Placeholder Card's substitution conventions for anything the tool does not need to know.

ZDR reality. Genuine ZDR on a web chat interface generally requires an enterprise agreement. The team and business tiers the policy expects most firms to hold are not ZDR: they are no-training with a stated retention period. That is Tier 2-capable under the policy only because the Tool List records the period and Section 7 makes the client-communication call.

Policy controls that carry the load. Section 3 vetting items 2 through 4 (training, memory, retention); checklist items 1, 2, 5, 6; the Placeholder Card.

3. Way two: add-ins inside Word

What it is. A vendor's add-in running inside Word (or the document management system), editing the open document, running skills against it.

Where the data lives. Exactly where way one lives; the difference is what gets sent and how it feels. The add-in is a thin client bound to a vendor account, and the open document, not just the prompt, transits to that account's endpoint on that account's terms. The document sits in the vendor's conversation or session store for the bound account's retention period. Nothing about running inside Word makes it local; the Word window is the most misleading data-residence cue in the building.

The exposure profile. The add-in inherits everything from the account it is signed into and adds two problems of its own. First, the input unit is the whole document: a client agreement opened with the add-in enabled is a Tier 2 event even if the prompt is "fix the cross-references." Anonymization is rarely practical for a live client document, so the account posture has to do all the confidentiality work. Second, the editing surface invites silent changes, which is why Section 5 requires a reviewable record (tracked changes, or before-and-after versions run through Word's Compare).

ZDR reality. The hardest of the three. Retail add-ins typically bind to team or business accounts, which are not ZDR; enterprise-bound add-ins exist but are exactly the expensive tier most small firms skip. Practically: a firm that needs ZDR-grade handling for a document should not have that document open with a team-tier add-in enabled. The alternative paths are the enterprise account, a vetted legal-specific tool whose terms cover document work (the Tool List records which), or keeping that matter's documents out of the add-in workflow entirely. One special case deserves its own Tool List row: Microsoft 365 Copilot operates inside the firm's own M365 tenant (the firm's licensed partition of Microsoft's cloud) under its enterprise data protections, which is a different residence story from a third-party add-in in the same Word window. It is not a safe harbor: Microsoft's cloud features are not on by default, so enrolling makes Microsoft one more custodian of client data, behind credentials that live outside the firm's own systems, and any cloud can be breached. The distinction and its limits belong on the Tool List, not in staff folklore.

Policy controls that carry the load. Section 3 vetting item 5 (account binding); Section 5's "AI in client documents" paragraph; checklist item 11; the Placeholder Card's add-in bullet.

4. Way three: harnesses

What it is. An agentic working environment where the model operates over a working set the firm controls: files in a folder, tools, standing instructions, running logs. Claude's Cowork, the coding-style environments now being used for document work, and the agent modes appearing inside legal tools are all harnesses. This is the "agentic AI" category on which COPRAC's 2026 guidance imposes the heightened supervision duties.

Where the data lives. Split, and this is the structural difference. The working set (documents, instructions, logs,

the matter folder) lives where the firm keeps it: local disk or firm-controlled storage. What transits to the vendor is the context actually assembled per request, over an API or app channel whose retention terms are typically the vendor's API terms, ZDR where negotiated, subject to the safety window in Section 1 above. Critically, continuity lives in the working set, not at the vendor: the harness's "memory" is a file in the matter folder that the firm can read, edit, and scope.

The exposure profile. Lowest vendor-side persistence and the only modality where matter separation can be enforced structurally (one matter, one folder, one session; the tool cannot remember what was never in its working set). In exchange, two exposures the other modalities lack. First, context assembly is opaque: an agent may read and transmit more of the working set than the user consciously selected, so the working set itself must be matter-scoped, which is Section 4's agentic access limits doing their job. Second, autonomy: the same environment that edits a document can send, file, or execute, which is why Section 5's agentic prohibition (no filing, sending, or client communication without human review of the specific output) and audit logging exist. The session logs a harness produces are attorney work product (Code Civ. Proc. § 2018.030) and belong in the matter file, which turns a compliance burden into the Section 10 documentation record almost for free.

ZDR reality. The most achievable of the three: API and enterprise channels are where vendors actually offer ZDR terms, and identifier substitution can happen in the working set before anything transits at all. Still subject to the safety window where a covered frontier model is used; model choice manages it, and the Tool List records it.

Policy controls that carry the load. Section 4's agentic access limits and memory subsection (continuity from the matter file); Section 3 vetting items 8 and 9; Section 5's agentic prohibition; Section 10 documentation.

5. The comparison in one view

Prompt and content go to. Web chat: Vendor cloud | **Word add-in:** Vendor cloud (bound account) | **Harness:** Vendor cloud (API channel)

Unit of disclosure. Web chat: What you type or attach | **Word add-in:** The open document | **Harness:** The assembled context (can exceed what you selected)

Conversation storage. Web chat: Vendor, until deleted | **Word add-in:** Vendor, per bound account | **Harness:** Minimal vendor-side; logs live in the matter file

Memory location. Web chat: Vendor memory store (toggle off) | **Word add-in:** Per bound account (toggle off) | **Harness:** The matter folder (firm-controlled)

ZDR availability. Web chat: Enterprise agreements only | **Word add-in:** Hardest; add-ins bind to non-ZDR tiers | **Harness:** Most achievable (API terms)

Survives even ZDR. Web chat: Covered frontier models' window | **Word add-in:** Covered frontier models' window | **Harness:** Covered frontier models' window

Distinctive risk. Web chat: Memory on by default; casual pasting | **Word add-in:** Whole-document disclosure that feels local; silent edits | **Harness:** Overbroad context assembly; autonomous action

Load-bearing policy controls. Web chat: §3 vetting 2-4, checklist 1-2, 5-6, Placeholder Card | **Word add-in:** §3 vetting 5, §5 documents paragraph, checklist 11 | **Harness:** §4 agentic limits and memory, §5 agentic prohibition, §10

6. What this means when choosing a setup

1. **Match the modality to the matter, not just the tool to the firm.** The same vendor can be Tier 2-capable in one modality and not another. The Tool List should record the modality alongside the account: "Claude, team, web chat" and "Claude, team, Word add-in" are different rows with different notes.
2. **For the most sensitive matters, the order of preference inverts intuition.** The harness, which looks the

most exotic, offers the most controllable footprint (firm-side working set, API-grade retention terms, substitution before transmission); the Word add-in, which looks the most ordinary, is the hardest to square with strict confidentiality demands.

3. **The safety window follows the model, not the modality.** It attaches to the vendor's newest covered frontier models even under ZDR terms; traffic routed to non-covered models under ZDR is not persisted at all. Where a client's instructions or a protective order cannot tolerate any vendor-side window, restrict the matter to non-covered models under ZDR terms, or treat it as Tier 3 and have the Section 7 client conversation; the answer is never a workaround.
4. **Memory is only solved where you can see it.** On web chat it is a setting to audit; in a harness it is a file to read. Prefer the one you can inspect.