What Is Actually In Your Fine-Tuned Model?

A machine-readable bill of materials for a model, and what it does not establish

By Joshua Bolick · Awareness Software Group _Published: 2026-09-03_

Ask a commercial AI vendor what is inside the model you are paying for. In most cases there is no document to give you. Not a redacted one, not one under NDA: the artifact does not exist.

That is a strange gap, because for ordinary software the question was settled years ago. A software bill of materials lists the components in a build, where each came from, and what licence each carries. For anything sold into government it is routine. The model, which is the component actually making the decisions, has been the exception.

This paper is about closing that gap: what a bill of materials for a model should contain, what a real one from a real training run looks like, and, at least as important, what it does not establish. The last part is where most vendor documentation goes wrong.

Why this became answerable in 2026

Three things converged, and only one of them is about us.

The regulation moved. OMB M-26-05, on 23 January 2026, rescinded M-22-18 and M-23-16, ending the government-wide CISA "Common Form" secure software attestation. Requirements became agency-specific and risk-based, and agencies may request bill-of-materials data on demand. The attestation form stopped mattering and the underlying document started mattering more.

The standards caught up with AI. CISA published *2026 Minimum Elements for an SBOM* on 29 July 2026, with NSA, FBI and fifteen international partners, replacing the 2021 NTIA elements: ten new fields, four revised, and scope explicitly extended to AI software. Both dominant formats already had the vocabulary. CycloneDX 1.5 added `machine-learning-model` and `data` component types with an embedded model card. SPDX 3.0, which is ISO-standardised, added an AI Profile and a Dataset Profile.

A pipeline already knows the answers. This is the part that decides whether the document is cheap or expensive to produce. If models are trained in notebooks, assembling provenance after the fact is archaeology. If they are trained by a pipeline, the base-model catalogue already carries provenance, the licence gate already recorded its decision, the dataset builder already recorded the splits, and the evaluator already recorded the harness. Nothing new is collected, and no training run is required to produce the document: it describes a model that already exists.

A caution before the argument

It is tempting to present the regulatory movement above as a requirement. It is not one. Agencies **may** request this data as part of risk-based review. There is no general mandate that a model ship a bill of materials.

We state that plainly because the entire value of a document like this rests on the buyer being able to check it. A supplier who overstates the requirement has told you something about how they will describe everything else.

What the document contains

The example throughout is generated from a real run: a contract-extraction fine-tune of a 4-billion-parameter open-weight model, 60 examples on a 48/12 split, QLoRA, 60 iterations. It is a smoke-scale run rather than a claims-grade result, and its own limitations block says so. It is here to demonstrate the document, not the model.

Where the base model came from

The question a procurement reviewer asks first:

"publisher": "Alibaba Cloud (Qwen team)", "licenses": [{"license": {"id": "Apache-2.0"}}], "properties": [{"name": "camp:model:originCountry", "value": "China"}, {"name": "camp:model:released", "value": "2025-04"}]

Publisher, country of origin, licence, release date. Whether a Chinese-origin base model is acceptable is the buyer's decision and it varies enormously by sector. The point is that it becomes a decision made on a recorded fact rather than an assumption nobody wrote down.

The licence chain, and the decision that was made about it

Our pipeline gates on the base model's licence before training begins. That decision used to live in a log nobody would read again. It now travels with the model:

{"name": "camp:governance:licenseDecision", "value": "allow"}, {"name": "camp:governance:licenseClass", "value": "permissive"}, {"name": "camp:governance:licenseReason", "value": "permissive license, cleared for commercial delivery"}

The derived model inherits the base licence, and the document records that inheritance rather than leaving counsel to reconstruct it from a model card and a download page.

What trained it

Dataset name and version, example counts, and the exact train and evaluation split. Also the source shapes, the connectors data arrived through, and how many records were ingested. Split sizes are the ones that matter for reading the metrics: an evaluation on twelve examples cannot support the same claims as one on twelve hundred, and the document should make that visible without the reader having to

ask.

Every metric with the baseline it was measured against

A quoted accuracy figure with no baseline is the most common way an AI result misleads without anyone lying. So each metric carries the base model's score beside the tuned model's.

This is the same discipline as 0004 and 0007, moved from the evaluation report into the supply-chain document, where a reader who never sees the evaluation report will still meet it.

The limitations, computed rather than composed

This is the part that decides whether the document is worth anything.

Evaluation split contains 12 example(s), too few to detect small differences reliably.

field accuracy: base 68.2% -> tuned 72.7% (+4.5% absolute) field 'agreement date': 87.5% (base 62.5%) field 'document name': 71.4% (base 85.7%) <- below the base model field 'effective date': 57.1% (base 57.1%)

Read the third line. On one of three fields the fine-tune left the model **worse than the one it started from**, 85.7% down to 71.4%. The headline gain is real and so is that regression, and both ship in the same document, generated from the run, without anyone having to remember to disclose it.

That matters for a reason beyond honesty. It is operationally useful: on that field, use the base model. No headline accuracy figure would have told you.

A bill of materials that reported only the improvement would be a brochure with a schema. The disclosure has to be produced by the same process that produces the claim. Otherwise it depends on somebody choosing to be forthcoming on a day when the number is disappointing, which is exactly the day it will not happen.

The rights boundary, stated rather than implied

The document says this, in both formats, in these words:

> Base-model licensing was evaluated by the CAMP license gate and its decision is recorded below. Rights in the customer-supplied training data are declared by the data owner and were not independently verified by the platform.

We document what we received. We do not audit whether the client held rights in the data they supplied, and we are not positioned to. A document that quietly implied otherwise would be worse than no document at all, because it would move a risk from one party to another without either noticing.

Both formats, always

Every run emits CycloneDX and SPDX. Buyers ask for one or the other and rarely say which up front, generating both costs nothing, and it removes a round trip from a procurement conversation. CycloneDX

leans toward security and inventory; SPDX leans toward licence governance. The underlying facts are identical.

What this does not establish

Four limits, stated here rather than left for a reader to discover.

It is not certified conformance. The documents target CycloneDX 1.6 and the SPDX 3.0 AI Profile and are self-consistent, but they have not been run through the official validators. "Targets" is the honest verb and we use it deliberately.

It is not complete against the 2026 CISA minimum elements. Weight-file hashes and an author signature are not implemented. Both are known gaps rather than oversights, and both are on the roadmap.

It does not verify the client's data rights, as above.

It does not make the model good. A bill of materials for a weak model documents a weak model accurately. The example in this paper is a 60-example smoke run and its own limitations block says as much. Provenance and quality are different questions and a document that blurred them would be doing the reader a disservice.

Why this follows from owning the model

The case for a private model you own, made in 0002, usually stops at the weights: the file is yours, it runs on your infrastructure, nobody can withdraw it or reprice it. True, and incomplete. Owning a file whose contents you cannot describe is a strange kind of ownership, and it is the version an auditor is least impressed by.

"You own the weights" is half a sentence. *"...and here is exactly what is in them, with the licence chain, the evaluation, and the places it underperforms"* is the whole one.

The question to put to any supplier

Ask for the bill of materials for the model you are being sold.

The answer is informative either way. A supplier who has one hands it over. A supplier who does not have one but has tracked the underlying facts can assemble it. A supplier who has to reconstruct provenance from memory will take weeks, and what you receive will be a description of what somebody believes rather than a record of what happened.

That difference is not visible in a demo. It is visible in this document, which is why it is worth asking for before rather than after you sign.