

Where Sentinel is, and where it goes.

NOW — SHIPPED v5.25

NEXT — FIRST SITE & PROOF

LATER — WRITE SIDE & STANDARDS

NOW — RUNNING TODAY

A proven engine, a team server, and the record behind every number

every figure below is measured on the current build, and re-measured on every release

100% P6 date agreement — 706/706 finishes across the library; 37/37 on a genuine site export

Full analysis suite — DCMA 14-point, EVMS, Monte Carlo risk, the end milestone and its driver, the DOE report in one action

The monthly update, in Sentinel — statused at the gate as one update; the report says what moved and why

Estate scale, and the program on one page — 1,000 schedules rolled up in 25–55 ms; review, Gantt, resources, budget

One server for the team — nobody locked out; tested at 50 at once

The approval gate — nothing executes until another person approves it; the assistant drafts, a person confirms

The record a surveillance reads — every version kept; sealed evidence; a VAR lifecycle; a reconciled budget log

Zero-install, zero-egress — its own Python, every package pinned in a bill of materials

A profile for each kind of project — 413.3C Critical Decisions, an EIA-748-E evidence map, operations

NEXT

The first site installation, and the proofs that come with it

the boundary rule holds at every step: client schedules are validated on the client's machine — only aggregate metrics travel

First shared-server site — installed on the customer's own hardware behind their sign-in, measured there with real people

Pilot on real data — site-side validation against the customer's own P6 exports

The P6 round trip, proven — the scoped import run through a real P6 Import Wizard on the customer's shape, before it is relied on

Independent review, signed installer — an outside review of the threat model and security plan, and a signed build

LATER

The write side, the standards, and the flywheel

each stage gates on proof, not on a calendar

The write side at program scale — combined schedules back into P6; proposed logic between projects

Validated federal datasets — IPMDAR and PARS extracts mapped and validated against the CADE and PARS schemas

Your precedent as templates — straw-man schedules drawn from your own completed work

Verification a QA program can audit — the V&V record each build ships, checked independently

The reference-class dataset — as-built history that never leaves the boundary, compounding into estimates nobody else can make

THE GOVERNING PRINCIPLE

Features move on this roadmap; the two rules never do. **No outside AI touches your data, and nothing that changes a schedule executes.** Every stage ships only when its safety case is proven — which is why the write side comes last, and why you can put Sentinel on a controlled machine today.