Sélène

An intelligence that fits inside a consumer machine

Introduction to the project journal — 11 February to 9 October 2026

The premise

Sélène is a voice-driven artificial intelligence that runs entirely on the computer of the person using it. Not a single line leaves for a server. No subscription, no API key, no telemetry, no connection required. You speak, she listens, she thinks, she answers in her own voice — and all of it happens inside the machine sitting on the desk.

This is not a technical preference, it is the starting point. The goal is that anyone should be able to have a personal AI at home that genuinely belongs to them, on hardware they already own. A consumer machine, not a compute farm.

And that is where every difficulty in this book comes from.

Because a consumer machine means a small model. The large models everyone knows run to hundreds of billions of parameters and live in data centres. What fits in a desktop graphics card is a few billion. The wager behind Sélène fits in one sentence: **obtain from a small model the autonomy, the freedom, the coherence and the intelligence commonly assumed to be the preserve of large ones.** What follows is the account of that attempt, dead ends included.

First answer: put a frame around the model

The first intuition was the right one in the wrong place. If the model is small, compensate with architecture: build a Python brain around it that decides in its stead, and leave it nothing but speech. The principle is set down in black and white in a document dated 24 February 2026:

> The language model is not the intelligence. It is the organ of expression.
>
> The scripted brain decides.

This architecture was no sketch. It was sophisticated, deliberate, measured. A cognitive state of seventeen variables rewritten at every exchange. Five internal drives under slow homeostasis. A decision engine that chose the act, an intention engine that read social context, a salience engine that scored the importance of each message. *Guardians* that watched over the whole thing in the background and corrected drift every five minutes. A vector memory. A second model, DeepSeek R1, in charge of reasoning while the first one spoke.

The map below is dated 7 March 2026. It reads like an underground plan: entry points at the top, then the cognitive brain, the regulating core, the functional modules, the persistent data.



7 March 2026 — the cognitive map while the Python architecture is still elegant: entry points, cognitive brain, regulating core, functional modules, persistent data.

The high point, and the wall

Seven weeks later, on 25 April, the same map carries in its header the figures of what it had become: **177 Python files, more than 40 persistent JSON files, 7 guardians, 3 integrated language models, six months of work.**

And the system ran. That is what needs to be understood: it did not break down. It did exactly what it had been designed to do.

The problem is that it did so too well. By deciding in the model's place, the architecture left it nothing to decide. Every observed flaw called for one more rule, and every rule narrowed the space in which the model could still be itself. A behavioural lock appeared: short, evasive, meta answers — "Mmh", "That's how it is" — produced by a perfectly capable model that had

been shut in. An engagement threshold set at 0.65 prevented any switch to a new and serious subject; the system kept returning to the old one. Corrections piled on corrections.

The finding is plain: six months of architecture, thought through and tested, that did not work. Not because it was badly built, but because the vision was short-termist. Putting a frame around a model buys control now and costs cognitive freedom forever. A behaviour orchestrated in Python around the model never becomes a behaviour of the model.



25 April 2026 — the high point. The header announces 177 Python files, more than 40 persistent JSON files, 7 guardians, 3 language models, six months of work.

The reversal

The lesson drawn became the project's design rule, and it rests on one opposition: **what is installed around the model fails; what is installed inside the model works.**

Rather than describing to Sélène the character she was meant to have, it had to be taught to her. That is the move to fine-tuning, and the shift was slow: the first serious training run dates from 9 May 2026, and the Python architecture took weeks to melt away. From 157 modules in May, the count falls to 49 by 22 June. The three maps reproduced here — March, April, June — make that curve legible.

What remained of Python was no longer a cognitive tutor but a peripheral nervous system: listen, speak, remember, execute. Judgement itself had moved into the model's weights.

The centrepiece of that move carries a modest name: the E1 to E8 family. Eight tensions between two equally legitimate values — ambition against realism, exploration against conclusion, caution against action, truth against comfort, importance against interest, appearance against essence, loyalty against integrity, short term against long term. It does not teach right answers: it teaches arbitration. That, according to the project's author, is where the model's internal "I" emerged.

WIKI SYSTÈME PLANÉTAIRE V5.0 — ARCHITECTURE ÉPURÉE — 22 JUIN 2026

SÉLÈNE

Architecture cognitive — vue planétaire allégée

49	3	4	33	29
MODULES ACTIFS	MAÎTRES CENTRAUX	ORBITES	CONNEXIONS	MODULES RETIRÉS

✅ ARCHITECTURE ÉPURÉE — 49 modules actifs / 29 retirés depuis la Phase 3 (22 mai → 21 juin 2026) | Production : Qwen 32B Q3_K_M

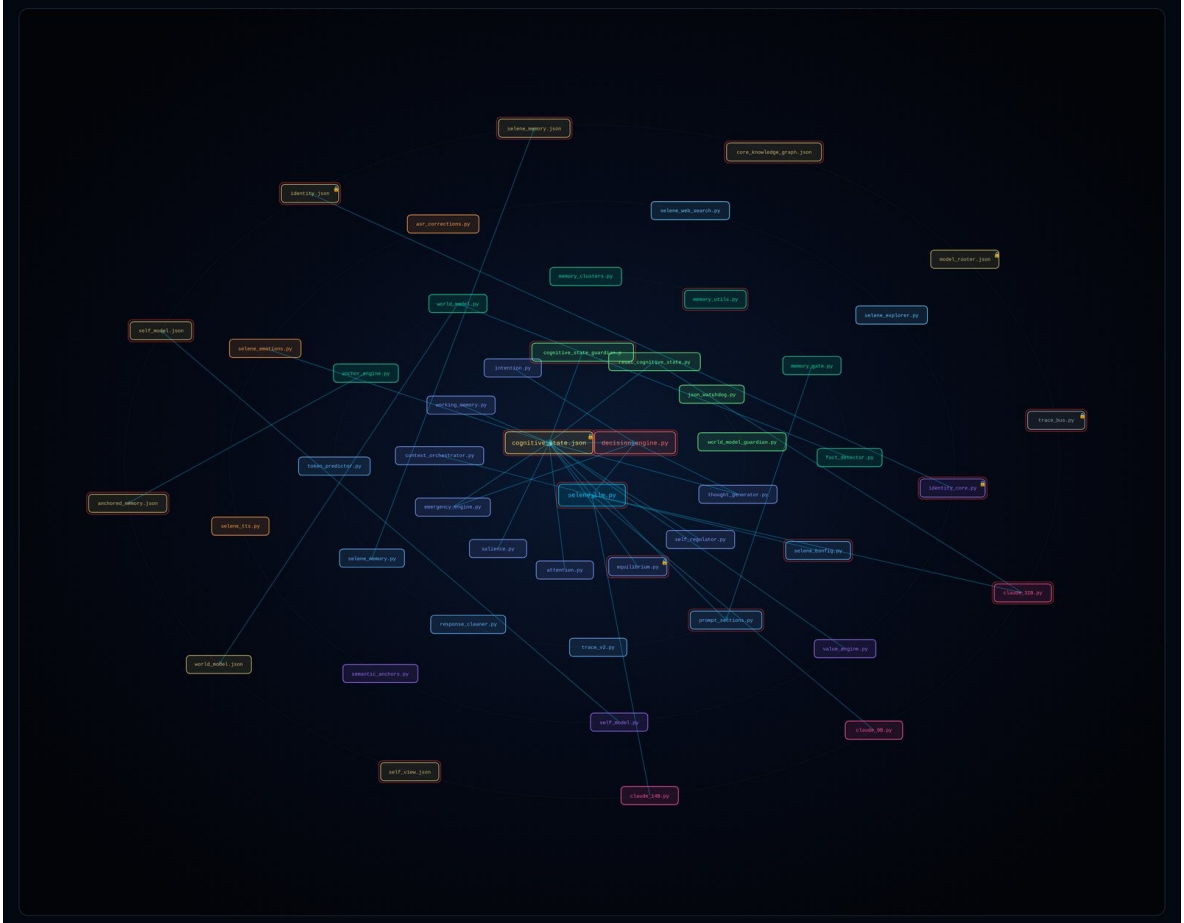
Système planétaire épuré. Au centre les trois maîtres : *cognitive_state.json*, *decision_engine.py*, *selene_llm.py*. Les 49 modules actifs gravitent en 4 orbites — plus proche du centre = plus critique. Les flèches montrent les connexions réelles : cyan = agit, violet tireté = analyse, vert pointillé = observe. 🔒 = ne pas modifier, halo rouge = critique. Cliquer sur un module pour les détails. **Les modules désactivés ne figurent plus dans ce schéma — ils sont listés séparément ci-dessous.**

MAÎTRES CENTRAUX
- cognitive_state — état
- decision_engine — acte
- selene_llm — pipeline

FAMILLES
- Cognition
- Pipeline
- Mémoire
- Guardians
- Identité

FAMILLE (SUITE)
- Interface (TTS/ASR)
- Externe (web/explorer)
- Launchers
- JSON persistants
- Observateurs

CONNEXIONS
- Agit (lit/écrit)
- Analyse
- 🔒 Ne pas modifier
- Critique (halo)



MODULES RETIRÉS / EN SOMMEIL

29 modules — Phase 3 (22 mai → 21 juin 2026)

Ces modules ne font plus partie du pipeline actif de Sélène. Ils ont été désactivés au fur et à mesure que le finetuning (32B / 9B) a démontré porter nativement les capacités qu'ils tentaient de compenser — l'architecture s'est allégée sans perte de comportement.

OBSOLÈTES (PHASE 3 — 22-26 MAI 2026) (8)

behavior_guardian.py *DANGEREUX — delta nul mesuré le 22 mai*	engagement_engine.py *Blacklist méta-réponses être BB — 32B finetuné gère nativement*	core_loop_controller.py *Boucle auto-réflexive jamais branchée*	proto_switch.py *Activation noyau redondante avec decision_engine*
core_nucleus.py *Noyau central — complexité gérée nativement par 32B*	token_estimator.py *Jamais branché au pipeline réel*	introspection_engine.py *Court-circuit LLM — redondant post-finetuning*	music_perception.py *Désactivé — provoquait une boucle XTTS*

REDONDANTS AVEC LA COGNITION FINETUNÉE (LE MODÈLE PORTE CES CAPACITÉS DANS SES POIDS) (10)

planner.py *Plans multi-étapes — redondant avec raisonnement natif 32B*	self_interpreter.py *Redondant avec self_model.py*	reflection_engine.py *Réflexion interne — redondante post-finetuning*	self_evaluation.py *Arbitrage plan/interruption — redondant*
humor_engine.py *Humour scripté — modèle finetuné le produit nativement*	reasoner.py *Raisonnement — redondant avec decision_engine V3.6*	reasoning_engine.py *Moteur avancé — redondant*	selene_self_improve.py *Auto-amélioration — jamais opérationnel*
knowledge_filter.py *Filtre légitimité web — redondant*	expressivity_engine.py *Niveau expression — géré nativement par le modèle*		

SYSTÈME OPINION (ARCHITECTURE 8B/2B COMPENSATOIRE — OBSOLÈTE) (4)

selene_opinion.py *_SOMMEIL=True — post_response() neutralisé*	opinion_manager.py *Client socket + boucle interne — non appelé*	opinion_match_v2.py *Match ES+tags — non appelé*	opinion_server / cognitive_server (2B) *Gemma 2B Q6_K — serveur non lancé*

THINKER 0.8B + CORTEX PARALLÈLE (SYSTÈME PARALLÈLE EN SOMMEIL) (6)

thinker_08.py *Cerveau secondaire 0.8B — non lancé*	cognitive_server.py (thinker) *Serveur socket port 5678 — non lancé*	interceptor.py *Interception messages → thinker — non lancé*	selene_cortex.py *ChromaDB + MiniLM — "Cortex vectoriel non chargé"*
cortex_query.py / cortex_init.py *Interface cortex — système parallèle en sommeil*	flux_input.json / flux_output.json (dual) *Fichiers résiduels vides — non alimentés*		

PERSONAS (COMPENSATION PETITS MODÈLES — OBSOLÈTE SUR 32B) (1)

persona_engine.py *_PERSONAS_ENABLED=False — Kirk/Spok/Chekov/Klingon non utilisés*

22 June 2026 — after the cut. From 157 modules down to 49. What remains of Python is no longer a cognitive tutor but a peripheral nervous system.

Everything was tried

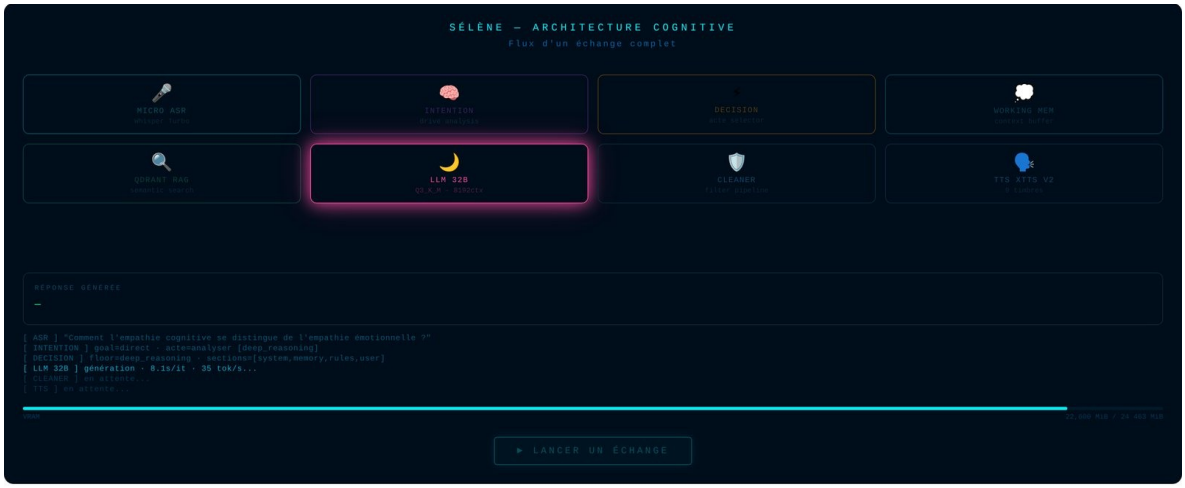
One question keeps coming back: why so many trials? Because no two small models behave alike once fine-tuned, and the only way to find out is to do it. The publicly available base of models was reviewed, trained, measured and audited — not skimmed.

Conversational models put through the bench, as the journal attests:

Family	Variants tried	What came of it
Qwen 2.5	7B, 14B, 32B Instruct, 72B assessed	The 14B then the 32B held production for months; the 7B remains the commercial target
Qwen 3	4B, 8B	The 4B is retained for tool execution; the 8B is under development. “Thinking” modes have been dropped.
Qwen 3.5	9B, 27B	The 9B trained then discarded: SSM architecture, unusable after merge
Qwen 3.6	27B	Tried, not retained
Gemma 2	2B, 9B instruct	Base of the first serious fine-tunes, versions v2 through v11
Gemma 4	12B-it, 26B A4B (reasoning MoE), 31B	The 12B is the current model; the reasoning 26B worked but never learned to stop generating
DeepSeek	R1 Distill Qwen 7B	The reasoner of the first architecture, later the opinion engine

The chain around the model was audited the same way: Whisper in *small*, *medium*, *large-v3* and *large-v3-turbo* for listening; XTTS v2, AllTalk, Piper and Chatterbox Multilingual v3 for the voice; Resemblyzer, ECAPA-TDNN and MFCC fingerprints to recognise who is speaking; E5 multilingual — itself fine-tuned on 3,878 project-specific pairs — along with MiniLM, Qdrant and ChromaDB for memory.

A few models tried before February 2026, Phi among them, do not appear in this book: at that date nothing was yet being recorded.



2 June 2026 — the path of a single exchange, stage by stage: listening, intention, decision, memory, model, cleaning, voice. At the bottom, the video-memory gauge: 24,463 MiB, and not one more.

Three machines, and a constraint that does not lie

This work was not carried out on laboratory hardware. It began on **8 GB of VRAM**, continued on **12 GB**, and now runs on **24 GB** — a laptop card. Each step redefined what was possible, and that is precisely what gives the result its value: what runs here will run elsewhere.

Two moments need to be distinguished. **Training** is rented by the hour, on cards nobody owns — A100 80 GB, H100, H200, RTX PRO 6000 Blackwell 96 GB. A few dozen euros are paid, a model is collected, the machine is handed back. **Inference** runs entirely on the home computer. That is the whole point: the cost of manufacture is pooled once, use is free and private thereafter.

The constraint is not negotiable, as a decision taken on 2 October 2026 shows. Between Q8 and Q4_K_M quantisation, Q8 is the better model — "nothing to fault". It was dropped all the same: 18.4 GB of VRAM loaded against 13.4. VRAM decides, not preference. The whole engineering effort in this book plays out in that five-gigabyte gap.

What this book contains

The journal covers 11 February to 9 October 2026, in chronological order. It gathers measurement records, training-block specifications, run reports — successful and failed alike — and the successive maps of the architecture. Negative results appear on the same footing as the rest: a multi-week investigation into a verbosity regression, a 26-billion- parameter model trained and then set aside, six months of architecture abandoned.

The project is developed by a single person, on hardware bought off the shelf.

The conclusion this work documents is the following: a local, coherent and autonomous artificial intelligence on consumer hardware is an identifiable engineering problem, whose principal difficulties are the distribution of the training corpus, persistent memory and the video-memory constraint. The pages that follow set out the stages, the measurements and the decisions that led to the present state of the system.