

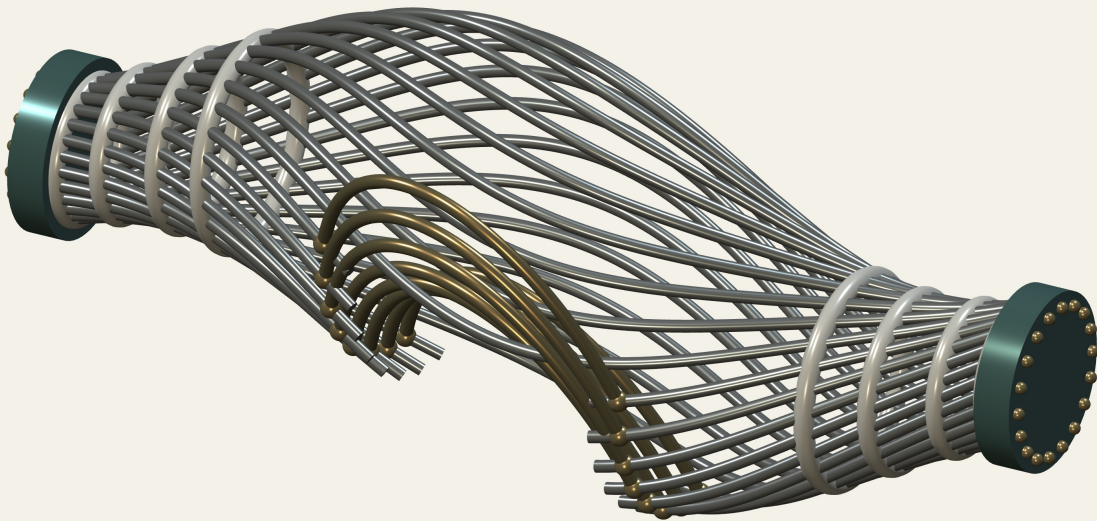
MONTREAL.AI • QUEBEC.AI

OWNER • PRACTITIONER • ENGINEER

SUCCESSOR Ω

The Field Guide

Your first mission.
Your evidence.
Your institution.



Vincent Boucher

President, MONTREAL.AI & QUEBEC.AI

6 September 2026 • Research & Use 1.1

Conceptual rendering, not a working apparatus. Research outputs confer no authority.

Find your next step

Start with chapters 1–3. Use chapters 11–13 only when you want to run code. Read chapters 14–18 before planning any real deployment.

-
- 1 Start with one useful decision
 - 2 Choose your route
 - 3 Your first 15 minutes
 - 4 Select a mission worth proving
 - 5 Write the Mission Constitution
 - 6 Prepare evidence without leaking the exam
 - 7 Give Beta a fair chance
 - 8 Calculate value once, on one horizon
 - 9 Manufacture competing candidates
 - 10 Read a world model as a hypothesis
 - 11 Run the smallest native example
 - 12 Understand your first result
 - 13 Observe revision and neural guidance
 - 14 Freeze, then commission independent proof
 - 15 Separate the decision gate from authority
 - 16 Operate with a visible evidence boundary
 - 17 Stop, preserve, recover, requalify
 - 18 Own the portable mission layer
 - 19 Install only what your route needs
 - 20 Troubleshoot without weakening controls
 - 21 A practical owner cadence
 - 22 What to delegate—and what to retain
 - 23 The mission portfolio and the long horizon
 - 24 Sources, terminology and current limits
-

This guide separates the target architecture, the current research implementation and the evidence required for real operation. Draft completeness is not proof. The illustrated workflow is not an autonomous enterprise deployment.

01 / 24

Start with one useful decision

The goal is not to own the largest model. It is to retain a mission capability that remains inspectable when people, models and suppliers change.

What SUCCESSOR Ω means

The architecture joins a mission, permitted evidence, executable hypotheses, skills, tests, comparative proof, policy and a Chronicle. A trained model is one component. An admitted institution is a stronger, separately earned state.

What this package is

A research publication, executable local examples and a practical scoping kit. The included v10.2 source snapshot is unchanged. This is not a hosted chatbot, an automatically connected enterprise system or a production deployment.

Your first deliverable

Write one sentence: “For [decision], improve [measured outcome] against [incumbent and relevant Beta], within [constraints], while [human] retains authority.” Then identify the evidence that could show you are wrong.

A useful first outcome is a well-formed mission—not a claim of superintelligence.

02 / 24

Choose your route

You can begin without installing anything. Run code only when you need to inspect or reproduce the computation.

Owner • no code

Open START_HERE.html, then First Mission. Complete the six-step local worksheet, examine the hypothetical economics and export your draft. Read the six-page AEON decision book to understand the example. No API key or payment is requested.

Practitioner • inspect the mechanism

Use AEON_Continuity_Lab.html for finite repair and dependence calculations. Use the retained AEON Mission Command for frozen model predictions and intervention enumeration. These are research calculators, not equipment controllers.

Engineer • reproduce and adapt

Run SUCCESSOR.py doctor. Start with the tiny numerical mission before the full AEON retraining. Install dependencies only inside an isolated environment. Customer data needs an explicit adapter and new evaluation; changing filenames does not implement a new domain.

Browser reading, local computation and production admission are three different activities.

03 / 24

Your first 15 minutes

This is an orientation budget, not a promise to create a qualified specialist in fifteen minutes.

Minutes 0–3: keep the package together

Extract the complete ZIP into a normal local folder. Open START_HERE.html in a desktop browser. PDFs can be read on a phone; use a desktop for code and reliable access to linked local files. On a managed device, follow its policy if local HTML is blocked.

Minutes 3–7: follow a worked example

In First Mission, load the clearly labelled fictional example. Read its decision, exclusions and best-Beta requirement. Its economics deliberately fail the stated threshold. Observe that a complete form does not become a proof receipt.

Minutes 7–15: draft your own next decision

Replace the example with your mission, owner and observable outcome. List missing evidence rather than inventing it. Export the JSON draft and the Markdown brief. The files are an engineering handoff; they are not executable runtime policy.

By the end: one decision, one accountable owner, two comparators and a list of missing evidence.

04 / 24

Select a mission worth proving

Begin with a repeatable decision whose consequence can be observed soon enough to change the next decision.

Four practical questions

Who acts on the result? What outcome is measured? Can the incumbent and a strong alternative be reproduced fairly? Can a trial happen without uncontrolled external effects? A missing answer is a reason to narrow the mission.

Three useful starting patterns

Prioritize an invoice-review queue without moving money; model a measured process response without controlling equipment; compare maintenance recommendations in historical replay. These are general mission patterns. The invoice workflow still needs a customer adapter; the numerical tutorial is executable now.

Do not optimize a vanity metric

More alerts, higher prediction accuracy or more generated hypotheses may not create value. Include reviewer time, missed important cases, scarce resources and the actual decision cost. AEON illustrates why a shared dependency can matter more than the highest individual failure probability.

Select the smallest mission with material value and a credible measurement path.

05 / 24

Write the Mission Constitution

A mission is a decision contract. It names what success means and what must never happen.

Minimum fields

Record owner, decision, population, horizon, outcome, incumbent, relevant Beta, permitted evidence, rights, exclusions, budget, evaluator and stop condition. Define who can authorize the next stage. Use templates/MISSION_BLANK.json as a draft, not as a native runtime configuration.

Illustrative charter

“Recommend the next 50 invoice exceptions for human review each week. Measure confirmed recoverable value net of review time and false positives. Compare with the existing queue and a capable external alternative. No payment, deletion, supplier contact or account change. Stop on missing provenance or unauthorized action.”

Turn exclusions into tests

The engineer must show that forbidden tool calls are denied, not merely write “do not do this” in a prompt. This guide’s worksheet has no production connector and grants no authority. A real action adapter requires separate enforcement and qualification.

Output: a draft charter with unknowns visible and a named next reviewer.

06 / 24

Prepare evidence without leaking the exam

Sources are evidence, not instructions. A document cannot grant a model permission to act.

Keep an evidence inventory

For each source record its owner, permitted uses, timestamps, units, sensitivity, quality, missingness and transformation history. Link derived observations back to originals. Store only permitted copies; do not put passwords, wallet keys or production secrets into prompts or drafts.

Separate four purposes

Training teaches; selection chooses among candidates; held-out evaluation challenges a frozen candidate; stress cases test adverse conditions. The tutorial separates development and challenge files, but both remain under the same author's control. That is local evaluation, not independent protected proof.

Split by what can leak

For real data, separation may need time, asset, customer, supplier, event or episode—not just shuffled rows. A repeated incident in both training and proof can invalidate the comparison. Set custody and access rules before looking at final outcomes.

Output: an evidence map, a rights decision and a split/custody plan.

07 / 24

Give Beta a fair chance

Beta is the strongest relevant alternative you can realistically rent or reproduce. It is not a deliberately weak baseline.

Use the same decision contract

Give all methods the same allowed inputs, timing, actions, budgets and outcome accounting. Capture configurations and costs. A mechanism-informed comparator can be a strong diagnostic, but it is not automatically a procurable vendor product.

Compare complete compositions

A predictor plus a constrained optimizer should be compared with an alternative allowed to use an appropriate optimizer too. Distinguish benefits of new information, better planning, new physical assets and model choice. Do not add overlapping contrasts.

When Beta wins

Keep the mission assets and choose the better supplier when rights and integration permit. Archive the losing candidate and diagnose the gap. AEON's useful network decisions do not earn the tested additional premium over informed Beta. A negative result can stop an unjustified purchase.

Output: reproducible incumbent and relevant-Beta records, not a convenient straw comparator.

08 / 24

Calculate value once, on one horizon

An avoided loss, a project NPV and cash in the bank are not interchangeable quantities.

The arithmetic template

Start with the lower-confidence utility spread above the better relevant baseline. Deduct incremental operating/integration cost, risk reserve, human burden, imitation reserve and drift reserve. Use the same currency, population and period. Do not deduct a cost again if it is already in utility.

A deliberately unsuccessful example

Illustrative CAD per quarter: point spread 20,000; lower spread 12,000. Deductions are 3,000 + 4,000 + 2,000 + 1,000 + 1,000. Net = 1,000. Against a threshold of 5,000, the arithmetic does not pass. No real evidence or independent proof supports these example inputs.

What the worksheet does not do

It checks supplied arithmetic; it does not estimate confidence bounds, verify invoices, certify reserves or authorize investment. A positive number remains hypothetical until its complete evidence and hard gates support a scoped claim.

Output: a traceable economic ledger, including negative and insufficient-evidence outcomes.

09 / 24

Manufacture competing candidates

Use the simplest composition capable of improving the decision. Complexity must earn its cost.

What the engine can execute here

The included scientific branch fits sparse numerical expression programs to typed observations. It can revise a parent after new contradictory observations and can use trained neural surrogates for experimental proposals. It has finite operators, variables and search budgets.

What your team adds

A customer adapter supplies domain variables, permitted actions, outcome accounting, evidence contracts and tests. AEON supplies a dependency topology; the learner does not discover that topology. A general enterprise mission is not implemented by renaming the sample files.

Preserve alternatives and failures

Retain a conventional method, at least one appropriate strong Beta, and candidates with distinct cost or robustness trade-offs. Keep failed hypotheses and stress outcomes. The Foundry and Gym produce formation evidence; they cannot certify their own final claim.

Output: candidates, executable tests and honest limitations—not a single unchallenged champion.

10 / 24

Read a world model as a hypothesis

An executable expression tells you what the system computes. It does not prove that the world follows the expression.

Inspect five things

Read the input names and units, validity bounds, explicit expression, residual/error threshold and observations that would falsify it. Record the parent identity and development evidence. A projected probability and an unconstrained raw prediction are different outputs.

Check consequences, not only fit

A precise predictor can still support the wrong decision if actions are infeasible, costs are counted incorrectly or the objective is inappropriate. Check delivered output, conservation, dependencies, constraints and full cost through separate code where practical.

Interpret status correctly

CANDIDATE means a numerical hypothesis was produced. LOCAL_PASS means it passed the declared local test. FALSIFIED_WITHIN_PROTOCOL means that test found a contradiction. INCONCLUSIVE means the bounded search did not find an adequate candidate. None grants production authority.

Programs can be inspected precisely—and can be precisely wrong.

11 / 24

Run the smallest native example

This optional route requires Python and NumPy. It actually calls the unchanged v10.2 scientific engine.

Prepare and diagnose

From the extracted package root, run the diagnostic. It never installs packages and never contacts a service. Use the installation page later in this guide if NumPy is absent.

```
python3 SUCCESSOR.py doctor
```

Discover from development observations only

The two development splits contain 35 training and 15 selection rows. They are synthetic observations of a dimensionless quadratic response. Use a new output path.

```
python3 SUCCESSOR.py discover --problem templates/numerical/PROBLEM.json --data  
↪ templates/numerical/DEVELOPMENT.json --config templates/numerical/SEARCH_CONFIG.json --out  
↪ var/first/DISCOVERY.json
```

Evaluate after freezing

The separate challenge file contains 20 held-out and 20 stress rows. Its labels are not passed to the discovery call. Both files remain public tutorial data, not protected independent proof.

```
python3 SUCCESSOR.py evaluate --theory var/first/DISCOVERY.json --data  
↪ templates/numerical/CHALLENGE.json --out var/first/EVALUATION.json
```

Expected: CANDIDATE, then LOCAL_PASS, with production_authorized set to false.

Native science CLI; templates/numerical; guide/results. Windows: use the prepared python executable.

12 / 24

Understand your first result

The tiny example teaches the file contract, not a proprietary industrial breakthrough.

What was learned

The supplied response is $y = 1.8x + 0.25x^2$. The search fits coefficients and selects an executable structure from its allowed grammar. Inspect terms, evidence, execution_contract, parent_id and id in DISCOVERY.json. A different algebraic ordering can express the same function.

What you should retain

Keep the problem, development data, untouched challenge data, search configuration, discovered theory and evaluation together. The theory identity binds the native execution contract, including NumPy version. A mismatched environment can correctly reject an old theory.

What would make this a real mission

Replace the toy response with permitted measured variables, instrument provenance and an appropriate noise model. Establish selection and protected-evaluation custody before training. Add a domain decision adapter and relevant comparators. This guide has not implemented those customer-specific steps.

A successful local test verifies the declared tutorial—not production usefulness or causal truth.

Native science/search.py validate_theory; new numerical tutorial.

13 / 24

Observe revision and neural guidance

Use the full native science demonstration when you want to see a parent model fail and a descendant change structure.

Run the science demonstration

This route trains a small neural surrogate, proposes a synthetic experiment, records a contradiction, expands the symbolic search and evaluates the descendant. It also keeps an out-of-family negative example.

```
python3 SUCCESSOR.py science-demo --out var/science-01
```

Inspect the active revision record

Open ACTIVE_REVISION.json and SUMMARY.json in the new output folder. Compare initial_formula and revised_formula; inspect the neural-ablation check. Review the failure receipt as carefully as the passing result. The research Chronicle is a local hash chain, not an authenticated multi-party institutional service.

Alternative: the standard-library world loop

The bounded storage and inventory demonstration can run without NumPy. It uses small predefined hypothesis families and explicit planning. It does not perform unrestricted scientific discovery.

```
python3 SUCCESSOR.py world-demo --out var/world-01
```

New evidence may justify a new candidate. It never silently renews old authority.

14 / 24

Freeze, then commission independent proof

An internal critic helps development. It is not the final independent verifier.

Before the trial

Commit the mission, comparator configurations, metrics, hard gates, cost accounting, sampling, stopping rule and analysis. Freeze code, models, prompts, tools, dependencies and permitted writable state. Protect final cases from the claimant.

During and after the trial

Run under the declared resources. Record failures, exclusions and human burden. The verifier signs a scoped proof or failure receipt with expiry and disclosed conflicts. A rights statement, a hash and a local test pass cannot substitute for that evaluation.

A valid no-go

A candidate may be useful but not worth its premium. It may win on average but fail a hard constraint. It may be technically good while evidence custody is invalid. Record the specific reason, preserve the candidate and choose repair, further measurement, Beta or retirement.

Output: an accountable scoped result. No proof receipt is generated by the draft worksheet.

15 / 24

Separate the decision gate from authority

D0–D4 describe institutional decisions. A0–A4 describe what an admitted release may do. They are not the same ladder.

The five decisions

D0 underwrite a mission. D1 approve bounded construction and evidence work. D2 submit an exact candidate to fresh proof. D3 admit only after proof and operational readiness. D4 renew, contract, substitute or retire. Each requires its own evidence and accountable decision.

The five authority levels

A0 observe. A1 recommend to a human. A2 act in a sealed sandbox. A3 reversible bounded action. A4 explicitly authorized consequential action. Authority is scoped by tool, target, budget and time, not a general intelligence score.

What is granted here

Nothing in this publication grants deployment authority. The guide tools produce drafts and research outputs. They do not invoke production admission, connect an actuator or prove that a real organization has authorized even A1 recommendations.

Never infer permission from capability, a version number or a completed checklist.

16 / 24

Operate with a visible evidence boundary

These are operating requirements for a separately admitted deployment—not controls already connected by this publication.

Before a consequential action

Check mission and release identity, valid proof, current policy, named approval, resource limits, monitoring and recovery. A lower-level tool must not allow the model to bypass the policy gateway. Use short-lived, scoped credentials; keep production secrets out of learning.

Daily view

Show the current release, permitted actions, proof expiry, incidents, measured outcomes, full costs and next decision. Distinguish a recommendation, attempted action, completed action and accepted outcome. An expected saving is not a receipt.

Weekly review

Ask what changed in evidence, instruments, costs, Beta, dependencies and the environment. Check false negatives, tail events, reviewer workload and recovery readiness. Assign a person and deadline to each unresolved finding. Keep a record even when the decision is “no change.”

Output: decisions and accepted outcomes traceable to their source, release and approval.

17 / 24

Stop, preserve, recover, requalify

A successor may inherit useful knowledge. It does not inherit the predecessor's proof or permissions.

When to stop or contract

Triggers include invalid identity, expired proof, compromised evidence, unauthorized actions, important drift, a stronger baseline, lost monitoring or unavailable rollback. The appropriate response depends on the mission; do not invent a universal automatic recovery.

Preserve what happened

Record release, input commitment, time, output, target, observed effect and responsible operator. Isolate questionable data from future training. Do not delete adverse results to restore a clean dashboard. Keep incident evidence under its rights and retention rules.

Research tools versus operational rollback

For this package, stop a running command with Ctrl+C and keep its partial output. The demos do not support arbitrary mid-run resumption: rerun into a new folder. Restoring research files is not undoing a real-world action. Real rollback requires the qualified action adapter and a tested runbook.

Output: an impairment record, a safe current state and an explicit re-proof path.

W §§6, 9.6, 10.5, 13.4–13.5; native demos refuse existing outputs.

18 / 24

Own the portable mission layer

Ownership requires rights and a successful reconstruction—not merely a ZIP file.

Keep the right objects

Retain the Constitution, evidence map, allowed source copies, model programs, trained weights where licensed, comparators, tests, proof/failure receipts, policies, Chronicle and restore instructions. Preserve supplier terms and exclusions with the artifacts.

Practice a replacement

Have a separate operator reconstruct the target environment, verify the release and substitute one declared supplier. Repeat reference decisions, policy-denial tests and recovery drills. Re-establish the appropriate proof after any material change.

What a checksum proves

The local manifest detects changes relative to its own record. It does not authenticate its publisher if an attacker replaces both file and manifest. Obtain trusted release identities through an authenticated channel. Recreate virtual environments; do not treat them as portable application data.

Output: a tested restoration and supplier-substitution record within the agreed rights.

19 / 24

Install only what your route needs

Reading PDFs and using the browser worksheets needs no Python installation. The commands below are optional desktop research setup.

macOS or Linux with Python 3.13 available

From the extracted root, create an isolated environment and install only the scientific tutorial dependency. The installation may access the package index. No provider API is called by the tutorial.

```
python3.13 -m venv .venv
.venv/bin/python -m pip install -r requirements-guide.txt
.venv/bin/python SUCCESSOR.py doctor
```

Windows with the Python launcher available

Use the environment's executable directly; no PowerShell execution-policy change is required for activation. This is a documented setup path, not a Windows qualification result.

```
py -3.13 -m venv .venv
.venv\Scripts\python.exe -m pip install -r requirements-guide.txt
.venv\Scripts\python.exe SUCCESSOR.py doctor
```

Full research replay

Install requirements-research.txt in the same environment for SciPy, scikit-learn and pytest. LaTeX compilation has separate requirements in BUILD_GUIDE.md. These pins describe the reproduction reference, not security certification or a fully hash-locked transitive environment.

Use a permitted, maintained interpreter and revalidate changes to the recorded environment.

20 / 24

Troubleshoot without weakening controls

Most first-run problems concern paths, dependencies or the meaning of a result—not a need to bypass security.

The browser page is blank or links fail

Extract the entire ZIP rather than opening an HTML preview inside it. Keep relative folders intact. Open the file in a browser, not a text editor. When local navigation is prohibited, read the PDFs and ask the administrator for an approved route; do not disable protections.

Python cannot find a module

Run the documented command from the package root using your chosen environment's Python. SUCCESSOR.py sets the vendor path for native calls. Install the appropriate requirements file in that same environment. A successful pip install elsewhere does not modify this environment.

An output already exists, or a theory is rejected

Choose a new output folder instead of overwriting history. THEORY_EXECUTION_CONTRACT_CHANGED can mean dependency or source drift; reproduce in the recorded environment or create a new candidate. Do not edit a theory's hash or authority to force acceptance.

INCONCLUSIVE and FALSIFIED_WITHIN_PROTOCOL can be correct research outcomes, not software failures.

21 / 24

A practical owner cadence

Scale the institution by decisions and evidence—not by the number of agents or generated pages.

First cycle: scope and baseline

Agree on one decision and its economics. Appoint the owner and evidence steward. Test data availability and reproduce the incumbent. Commission the relevant-Beta comparison before a large training spend. End with a build, narrow or stop decision.

Second cycle: form and challenge

Give the technical team an explicit research budget and deadline. Preserve candidate diversity, adverse outcomes and measured resource use. Freeze the promising candidate and hand it to the separate evaluation function. A calendar deadline cannot manufacture sufficient evidence.

Third cycle: qualify or stop

Where proof supports it, qualify the deployment target, recovery and scoped authority before operation. Review baseline movement and impairment regularly. If the claim fails, retain reusable assets and fund only a clearly identified next experiment—not endless retraining.

Time budgets are managerial choices. Evidence and consequence determine admission.

22 / 24

What to delegate—and what to retain

More bounded autonomy can be useful without giving the learner control over the evidence or permission system.

Delegate bounded research work

Within approved inputs and resources, automate extraction, candidate generation, replay, draft comparisons and report preparation. Require source references and failure disclosure. Restrict generated programs to the approved representation and execution environment.

Retain independent powers

Keep protected-case custody, final verification, policy administration and production credentials outside the claimant's control. Independence is organizational and operational, not a new role name in the same conversation.

Use prompts as drafting instruments

The bilingual prompt pack helps an approved assistant scope missions, identify evidence gaps, critique comparisons and draft a reviewer handoff. Prompts do not replace an adapter, sandbox, policy gateway or independent verifier. Do not upload restricted evidence merely to make a prompt more complete.

Automate useful bounded work; keep final proof and consequential authority separately accountable.

23 / 24

The mission portfolio and the long horizon

Each accepted improvement should expand what the owner can investigate—not erase the next mission’s proof burden.

Compound the durable layer

Reuse measurement contracts, validated interfaces, negative results, tests and supplier-substitution methods. Requalify models when their evidence, domain, costs or dependencies change. Shared infrastructure can reduce the cost of future proof, but each financial claim keeps its own baseline.

Keep physical and cash accounts separate

A good model does not fabricate parts or finance construction before cash arrives. AEON’s remote horizon makes repair crews, common failures, heat and signal delay explicit. Treat it as a research programme; do not extrapolate synthetic project NPV into a stellar valuation or completion date.

Choose the next bottleneck

Where the current candidate ties Beta, better evidence, physical process improvement, repair logistics or a changed objective may matter more than a larger predictor. Identify the decision that a new observation could change and its complete cost. Stop when the evidence does not justify proceeding.

The general method: constitute, compare, learn, challenge, admit only when earned, and renew.

24 / 24

Sources, terminology and current limits

The general workflow comes from the supplied architectural white paper. The runnable instructions come from the inspected source snapshot and the new guide adapter.

Reference key

W: Neural-Symbolic SUCCESSOR Ω, 25 August 2026, included as docs/Architecture_Reference_EN.pdf. Use pages 7, 12, 21 and 33 for the decision, architecture, succession and authority diagrams; use pages 53–58 for the claims-to-proof covenant and templates.

Key words

Incumbent: the current method. Beta: the relevant rentable or reproducible alternative. Candidate: an unadmitted proposed composition. Alpha: the scoped net advantage that survives the defined proof and deductions. Chronicle: institutional history, not proof by itself.

External practice and limits

Python's venv documentation supports isolated environment setup. pip's repeatable-install documentation distinguishes version pins from stronger reproducibility controls. NIST AI RMF provides voluntary risk-management context, not certification of this package. The current release establishes local research workflows; no field Alpha, independent security audit or production admission is asserted.

Use the source, the executable record and the unresolved-issues register together.