



VERIFIABLE SMR COMPUTATIONAL ENGINEERING

smrforge.io · Open-core · Private beta

SMR DESIGN · ANALYSIS · OPTIMIZATION

Reactor physics your reviewers can reproduce — not just believe.

A full multi-physics toolkit for Small Modular Reactors where every result ships as a fingerprinted, independently reproducible evidence bundle — and is labeled, honestly, for how far to trust it.

```
pip install -e . --no-deps --no-build-isolation --no-compile
→ smrforge doctor → a screening k-eff in a minute
```

EVIDENCE_ENVELOPE · ADR-013 ♦ GPG-SIGNABLE

analysis	gt-mhr-350
k_eff	0.9051
fidelity	screening_estimate
sha-256	a3f1c8...9e0b
replay	bit-exact @ epoch

```
$ smrforge verify reviewer-kit
./bundle
```

✓ reproducible by a third party · exit 0

THE DIFFERENTIATOR — BUILT FOR THE PERSON WHO DOESN'T TRUST YOU

01 · FINGERPRINT

Evidence envelopes

Every output is a schema-registered bundle with a SHA-256 fingerprint of its inputs and result — optionally ed25519 GPG-signed.

02 · REPLAY

Bit-exact reproduction

A pinned SOURCE_DATE_EPOCH and the replay_bundle() contract regenerate the same bytes from the same inputs.

03 · VERIFY

exit 0 = it holds

Verification is a standalone, offline check against the open **Evidence Envelope** schema — a third party reproduces or refutes your number in seconds, without trusting or licensing SMRForge.

ONE REPRODUCIBLE MODEL, A FULL MULTI-PHYSICS STACK — ACROSS LWR-SMR, HTGR (PRISMATIC & PEBBLE), MSR & MICROREACTOR ARCHETYPES

NEUTRONICS	Full-core nodal, MC & transport · variance reduction	THERMAL-H	Coupled channel TH · conjugate heat transfer	FUEL CYCLE	Burnup & depletion · Xe/Sm poisoning
SAFETY	Kinetics transients · LOFC / ATWS / RIA / LOCA	UNCERTAINTY	MC / LHS / Sobol UQ · sensitivity	NUCLEAR DATA	In-tree ENDF/B · self-shielding, TSL
GEOMETRY	Hex / Cartesian / R-Z cores · power maps	ECONOMICS	LCOE & fuel cycle · SMR factors	INTEROP	OpenMC round-trip · Serpent, MCNP, plugins

WHAT IT GIVES YOU

- ✓ A physics_fidelity label + plausibility band on every result
- ✓ Machine-readable caveats that travel with the output
- ✓ One-command escalation to OpenMC / Serpent for a validated k-eff

WHAT IT WON'T PRETEND

- Screening k-eff is a trend, not a validated absolute
- The open tier is an evaluation surface, not licensing sign-off
- Nothing is tuned to hit a benchmark it was measured against

Community smrforge · MIT

Evaluation & integration — prove fit in the open.

- Full screening stack + evidence envelopes & reviewer-kit
- One-command archetype starter kits: `smrforge new <archetype>` → a runnable, baseline-checked project
- OpenMC export/import · Serpent round-trip
- Governance CLI, benchmarks, stable API, plugin system

Pro smrforge_pro · FSL-1.1-MIT

COMMERCIAL

Production & commercial depth for deployment.

- Production physics kernels (deep S_N/MoC, adjoint, IQS, burnup-in-loop)
- CAD / DAGMC · interactive 3D · advanced variance reduction
- Confirmatory Console — re-run a submission, diff every claim, ship a signed verification report; subsidized for regulators & national labs
- Regulatory-oriented reporting · assurance & benchmark evidence

```
# Screening k-eff, labeled honestly (private beta — from a
checkout) import smrforge as smr r =
smr.analyze_preset("gt-mhr-350") print(r["k_eff"],
r["physics_fidelity"]) # 0.9051 screening_estimate
```

Open standard & verifier →

support@smrforge.io

smrforge.io · Community MIT · Pro commercial

Export-control note: nuclear-related software may be subject to U.S. federal law (10 CFR Part 810, EAR, ITAR, OFAC) regardless of source-license terms. Community: MIT. Pro: source-available under FSL-1.1-MIT; commercial use under separate agreement. Screening results are labeled estimates, not validated or licensing-grade absolutes.