



See the Cell before it executes.

Celluster introduces execution-native semantic computing: authored intent, execution evidence, adaptation, continuity, and operational bounds are expressed as one Cell execution contract - rather than reconstructed around execution by controllers, schedulers, and operational layers.

AUTHOR

A Cell is authored, not configured. The supplied .cell describes the execution outcomes, lifecycle, evidence, Reflex behavior, workload surfaces, resource relationships, policy, and bounded operations that execution must preserve.

COMPOSE

The Release-1 binary compiles the authored Cell into canonical projected execution truth and presents that truth through the Cell Life Surface (CLS).

INSPECT

Before running the underlying runtime, inspect how the Cell is designed to live: what must remain true, what constitutes meaningful departure, what responses are authorized, and how continuity is preserved.

THE EXECUTION STORY

AUTHORED
WILL



EVIDENCE



DEPARTURE



BOUNDED
RESPONSE



CONTINUED
LIFE

The included authored Cell is an intentionally demanding LLM inference service: latency-critical inference, economical embeddings, GPU profile changes, RoCE fabric, checkpoint continuity, evidence health, cost and utilization objectives, and bounded autonomous adaptation. The point is not the GPU itself. The example makes the execution model concrete enough to expose the breadth of what a Cell can author and preserve.

One authored Cell. Multiple ways to understand its life.

CLS is the inspection surface for projected Cell behavior. The flagship view preserves the whole execution story; perspectives let you enter that same truth through a specific concern.

Start with the flagship

cell life <contract.cell> opens the whole-Cell experience. It is the primary entrance, not a dashboard and not a deployment status screen. It shows how the authored Cell is designed to live before execution.

Then explore four primary perspectives

SEMANTIC	Authored execution meaning: outcomes, lifecycle, awareness, autonomous response, safety, workload life, realization, policy, and bounds.
RELATIONSHIPS	How execution truths bind, protect, depend, isolate, and participate in one Cell.
LIFECYCLE	How the Cell continues through creation, candidate preparation, readiness, change, acceptance, failure, recovery consequences, and decay.
ADAPTATION	Evidence → healthy expectation → meaningful departure → Reflex evaluation → safety and operational override → authorized response → continued life.

Focused perspectives

Objective · workload · named workload · evidence · drift · Reflex · continuity · recovery · resources · policy · economics · security · proof · diagnostics. These are direct entrances into the same projected execution model, not independently reconstructed views.

Presentation modes

Rich Terminal is the default interactive reading surface. **--plain** preserves deterministic accepted text. **HTML/SVG** provide portable visual presentation. **Video** is temporal traversal of the same visual truth; it does not simulate runtime occurrence.

READ CORRECTLY	Release 1 presents projected execution behavior derived from the authored Cell. Projected truth is not a claim that an event has already occurred at runtime. Runtime evidence and observed occurrence require runtime authority.
-------------------	---

Take the authored Cell into your own evaluation path.

The preview is designed to make the execution model understandable first. From there, organizations can evaluate Celluster against their own workloads and infrastructure under controlled runtime conditions.

SEE	Open the flagship Cell Life experience and understand the whole authored execution story.
EXPLORE	Enter Semantic, Relationships, Lifecycle, and Adaptation; then use focused perspectives where useful.
EXPERIENCE	Open CELL-LIFE/cell-life.html and the package-facing Cell Life film for the visual traversal.
INSPECT	Read the supplied authored .cell after seeing its projected behavior. Trace objectives, evidence, drift, Reflex behavior, continuity, resource relationships, policy, and operational bounds.
VALIDATE	Bring an authored Cell into a controlled lab workflow and evaluate the underlying runtime separately.

Runtime Lab Evaluation

Evaluate execution behavior, Reflex response, lifecycle transitions, continuity, and adaptation against real workloads, accelerators, networks, and operating conditions.

Design Partner Program

For organizations ready to validate distributed execution across real hosts, accelerators, networks, workloads, and continuity scenarios under production-scale conditions.

Start a conversation: Celluster website → Contact Celluster

Direct: nikhil@celluster.ai

INTELLECTUAL PROPERTY & EVALUATION BOUNDARY

CELLUSTER IS PROPRIETARY TECHNOLOGY. The Cell execution model, execution-native semantic computing architecture, Reflex-native execution mechanisms, and associated implementation are proprietary Celluster technology. Celluster has patent claims that have been allowed, with additional patent rights and filings at various stages.

This Release 1 package is provided to **evaluate the Celluster Execution Authoring Preview** - not to grant implementation, reproduction, redistribution, derivative-development, or other rights beyond the permitted evaluation scope.

Before using the package, review:

[RELEASE/NOTICE.pdf](#) - Celluster ownership, intellectual-property and proprietary-technology notice.

[RELEASE/EVALUATION-TERMS.pdf](#) - permitted evaluation use and the boundaries that apply to this package.