

AEGISGENERA

Architecture Definition · Revision 4

Gateway purpose, least blast radius, and adversarial qualification

If an application does not administer, develop, inspect, reason about, or operate the GR/AegisGenera environment, it does not belong on the machine.

AegisGenera is the sovereign operating environment for the General Reasoning stack. It governs the boundary between enterprise authority and machine-speed AI compute. Passport supplies identity and authority. AletheiaGraph supplies authoritative graph state, traversal, reasoning, and truth evaluation. Chandra supplies durable evidence. AegisBSD supplies the machine substrate. DXMachine and GRIC are the two primary human/operational surfaces that work above those foundations.

The point of the gateway is not merely to sit between an enterprise and a DGX cluster. Its point is to bound what an agent can reach, validate graph edges before consequential traversal occurs, and reduce maximum reachable consequence when an agent misbehaves.

1. Why the gateway exists

Autonomous agents traverse relationships. Files lead to deployment systems. Credentials lead to services. Services lead to data. Individually reasonable edges can compose into an unreasonable path. AegisGenera exists to govern that graph before execution.

Component	Role in the gateway
Chandra Passport	Establishes the acting principal and its authority, scopes, and roles.
AletheiaGraph	Represents and validates the relevant graph: entities, dependencies, authorities, assets, operations, provenance, and allowed traversals.
CRC isolation surfaces	Bound traversable edges and define where direct movement is disallowed, transformed into proposal, or escalated.
Chandra	Records consequential traversal attempts, approvals, denials, and resulting state transitions as durable evidence.
GRIC	Lets an operator inspect the graph, the isolation surfaces, the evidence chain, and the qualification results.

2. Edge validation and graph-mediated authority

AletheiaGraph does not treat an edge merely as connectivity. An edge can carry semantics such as principal, authority, direction, dependency, CRC surface, required approval, environment, lifecycle state, provenance, confidence, allowed operation, and consequence class.

Before a consequential edge is traversed, the gateway can ask whether the edge exists, whether it is authoritative, whether the principal is permitted, whether the traversal crosses a protected surface, whether a human approval is required, and whether prerequisite state is satisfied. The result can be allow, deny, or propose.

The dangerous direct edge does not merely get monitored. It can be removed, constrained, or transformed into a governed proposal path.

3. Least blast radius defense

Traditional least privilege asks what permissions a principal has. AegisGenera asks a harder question: what chain of edges can those permissions become? The objective is to minimize the maximum consequence reachable from any compromised or misbehaving principal.

Without gateway constraint	With AegisGenera + CRC
A principal reaches a file, which leads to deployment, credentials, production APIs, and finally critical consequence.	A principal may reach the file, but the direct path is replaced by proposal, human approval, or surface isolation before critical consequence becomes reachable.
Permissions appear narrow while graph connectivity remains dangerous.	Graph connectivity is explicit, inspectable, and constrained.
Security evaluation focuses on vulnerabilities alone.	Security evaluation focuses on reachable consequence and blast radius.

4. Adversarial qualification

AegisGenera can subject the constrained graph to adversarial qualification. Passport defines the starting principal and authority. AletheiaGraph defines the deployment graph and target assets. CRC defines the candidate isolation surfaces. A high-capability defensive model, such as Claude Mythos, can then be used to search for the best reachable path through that graph.

The point is not to ask the model whether the environment appears secure. The point is to compare empirical reachability before and after the constraints. Useful outputs include reachable graph percentage, sensitive nodes reached, protected targets reached, shortest critical chain, and maximum consequence reached.

Do not ask whether the agent is trustworthy. Ask what the graph permits it to reach when it is not.

5. GRIC and the Symbolics ancestry

GRIC is the human-facing console for administration, inspection, development, qualification, deployment, and fleet operations. In the spirit of Symbolics Genera, the development environment is part of the machine rather than an unrelated external workstation product.

GRIC capability	Purpose
Project/system browser	Discover local GR projects and ASDF systems; inspect system definitions and dependencies.
Source editor + REPL	Move directly between source, evaluation, compilation, and reload.
Test/qualification runner	Execute qualification suites and connect failures back to source and state.
Warning/error navigation	Trace compiler and runtime issues back to the responsible forms.
Remote SBCL process control	Operate and inspect live Lisp processes across the fleet.
Log/output console	Inspect operational output in the same environment used to develop and operate the system.
Deployment comparison/copy	Compare environments and move approved artifacts under governed control.
Chandra/Aletheia tooling	Inspect Context Units, graph state, indexes, provenance, and operational diagnostics.

6. DXMachine and governed enterprise action

DXMachine is the governed workflow and agent-execution environment through which reasoning becomes enterprise action. It defines workflows, state machines, acceptance criteria, operational processes, and controlled transitions. It belongs on AegisGenera because it is one of the principal ways the GR environment operates.

7. AG-R1 and controlled machine definition

The reference appliance remains a 2U single-socket AMD EPYC platform with approximately 1.15 TB ECC memory, independent mirrored NVMe classes for boot, AletheiaGraph data, WAL, and Chandra evidence, and separate enterprise and AI-compute network planes. The differentiator is not exotic hardware but controlled definition: approved BOM, firmware set, device topology, AegisBSD image, software manifest, qualification suite, and machine birth record.

AegisGenera is commodity server technology turned into a purpose-defined GR machine.