

PUBLIC READER GUIDE

How to Read Authority Engineering Formal Foundations RC-1

A plain-English guide to the labels, symbols, and document roles

Purpose

This guide explains the names, abbreviations, symbols, document roles, and status labels used in the Authority Engineering Formal Foundations RC-1 package. It does not add, change, or interpret Authority Engineering beyond the completed RC-1 artifacts.

The short answer

RC-1 means Release Candidate 1: the first package proposed for review and publication. It is a release label, not a mathematical claim, a law, or the name of a new concept.

AE means Authority Engineering.

L-AE identifies a numbered formal module, or layer, in the package. L-AE0 is the core representation; L-AE1 through L-AE9 are the numbered extensions that use it.

AE-REG-#### is an identifier for an entry in the Authority Engineering Registry. It identifies the source entry from which a representation was compiled; it is not itself a theorem, result, or equation number.

What each public document is for

Document	Plain-English purpose
Consolidated Specification	The controlled index of what RC-1 contains, where each part came from, and what it does not claim.
Formal Theory Presentation	The mathematical reading: the allowed symbols, what a model is, and the set of formal statements.
Semantic-Completion Adjudication	The classification of which symbols are primitives, partial formalizations, background parameters, or implementation-only artifacts.
Machine-Readable Formal Theory	The exact logic statements in a computer-readable syntax. It is intended for tools and formal-methods reviewers.
Validation Record	The checks that were run and the limits of what those checks establish.

Publication Record

How to Read Authority Engineering Formal Foundations RC-1

Author: Rob Caswell

Publisher: Cyber Terrain LLC

Published: 2026-08-01

Zenodo version: v1.0

DOI: <https://doi.org/10.5281/zenodo.21741177>

Website: <https://cyberterrain.ai/papers/rc1-reader-guide.pdf>

Corpus Context

Corpus: Authority Engineering

Document ID: AE-RC1-READ-0001

Object family: AE-RC1-READ-0001

Publisher-imprint edition prepared for versioned republication.

The prior Zenodo record remains the citable record until its new version is published.

The module names

Module	Subject named in the existing package
L-AE0	Core state-transition representation.
L-AE1	Bounded Authority Realization.
L-AE2	Guard and task-force structures.
L-AE3	Sovereign Stop.
L-AE4	Continuity, Surface, and Topology.
L-AE5	Fabric and STOIC.
L-AE6	Consequence Contention.
L-AE7	Consequential Ownership Test.
L-AE8	Authority Attack Distinction.
L-AE9	Consequence Fidelity.

The three primitive symbols

The current RC-1 corpus treats exactly three Authority Engineering concepts as mathematical primitives. A primitive is a starting symbol: it is constrained by formal statements but is not defined in terms of other currently admitted Authority Engineering symbols.

Symbol	Existing source name	Read it as
$A(s,t)$	Authority	Authority at state s and transition t .
$C(t)$	Consequence	Consequence at transition t .
$\text{Sigma}(s,t)$	Consequence Seam	Consequence Seam at state s and transition t .

The symbols s and t are variables. In the current formal language, s ranges over states and t ranges over transitions.

Basic formal-language terms

Term or notation	Plain-English meaning in RC-1
State, Transition	Formal categories, called sorts, used to type expressions. They are not automatically

Term or notation	Plain-English meaning in RC-1
	claims about a particular real-world system.
src(t) / tgt(t)	The source state and target state of a transition.
Sort	A category of object allowed in the formal language, similar to a type.
Signature	The inventory of sorts, functions, and relations permitted in a formal presentation.
Model	A mathematical structure that assigns meaning to the formal symbols and makes the required statements true. It is not automatically an observed or real-world model.
Constraint or sentence	A formal statement compiled from the existing source material. In RC-1, this label does not make the statement an Authority Engineering axiom, law, theorem, or empirical result.
->	“Implies”: whenever the left side holds, the right side must hold.
<-> or iff	“If and only if”: each side implies the other.
exists	At least one object of the stated kind exists.
T_RC1	The name of the set containing the 33 formal RC-1 sentences.
T_RC1 = phi	Every RC-1 model satisfies the statement phi; this is semantic consequence.

What BAR, TF-H, TF-V, and STOIC mean in the package

BAR is the package abbreviation for Bounded Authority Realization, the subject of L-AE1 and the representation of Registry entry AE-REG-0047.

TF-H and TF-V are labels used in L-AE2 for two task-force structures. The completed corpus uses those labels but does not supply an expanded public-language name for the letters H and V; this guide does not invent one.

STOIC is the name used in L-AE5 and in the symbol STOICAnalysis. The completed corpus does not expand the letters as an acronym; it therefore should be read as a defined package label, not as a word whose expansion is assumed.

Constraint labels: how to read them

Each label identifies one numbered formal statement. The letters group the statement with its module; the number distinguishes statements within that group. Where the corpus gives no expansion of the letters, the label should be treated as an identifier rather than as an unexplained public acronym.

Label group	Module and topic
RC-1	The L-AE0 core constraint.
BAR-definition	The L-AE1 Bounded Authority Realization convention.
GC, HC, VC, SC	L-AE2 Guard and task-force structure constraints.
SS	L-AE3 Sovereign Stop constraints.
CT, AS, AT	L-AE4 Continuity, Surface, and Topology constraints.
FB, ST	L-AE5 Fabric and STOIC constraints.
CC	L-AE6 Consequence Contention constraints.
OT	L-AE7 Consequential Ownership Test constraints.
AA	L-AE8 Authority Attack Distinction constraints.
CF	L-AE9 Consequence Fidelity constraints.

For example, CF-2 means the second numbered constraint in the Consequence Fidelity module. It is not a cryptographic hash, a severity score, or a claim that the constraint has been proved true in every application.

Status labels: what they do and do not mean

Label	Meaning
Primitive concept	A starting Authority Engineering symbol retained without a definition in the current corpus.
Partial formalization	A source-bound necessary-condition representation, not a complete definition.
Background parameter	A typed term needed by the representation whose full contextual, engineering, temporal, observational, or empirical meaning is not supplied by RC-1.
Representation artifact	A symbol used to organize the formal encoding or implementation signature; it is not independently claimed as an Authority Engineering concept.
PASS	The stated check passed, within the stated scope.
PARTIAL	Some stated coverage or adequacy exists, but the relevant layer is not complete.
BLOCKED	The report does not claim that result because required proof or evidence is absent.
NOT CLAIMED	The package makes no claim of that result.

Terms that should not be over-read

SMT-LIB is the standard text format used for machine-readable logic constraints. Z3 is the automated solver used for the recorded solver replay. A successful solver replay means that the encoded constraints have a satisfying interpretation in that replay; it does not by itself establish a theorem, universal consistency, soundness, empirical validity, or truth about a real-world system.

FOL means first-order logic. RC-1 uses a many-sorted form of first-order logic with equality: the language has several typed categories of objects rather than one undifferentiated domain.

A practical reading order

1. Read this guide.
2. Read the Consolidated Specification for the scope and non-claims.
3. Read the Semantic-Completion Adjudication to see what each symbol's status is.
4. Read the Formal Theory Presentation if you need the mathematical structure.
5. Consult the Machine-Readable Formal Theory only when reviewing or replaying the exact logical statements.
6. Read the Validation Record to understand what was checked and what remains unclaimed.

Important limit

This guide makes the package easier to read. It does not expand the current theory, convert partial formalizations into complete definitions, or supply meanings that the existing RC-1 corpus does not provide.