CNVS

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

The Theory of Closed Native Verification Systems

Version 5.3

This document constitutes a more rigorous formalization attempt of the CNVS Theory presented in the previous documentation.

Introduction

Fix some notation errors, indexing errors, and add explanatory tables to better clarify the difference between semantic information and structural information.

Author: Massimo Comitato

22.05.2026, Italy (EUR)

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

Table of Contents

1. Introduction
2. Abstract
3. Problem Statement
4. General Definitions
5. Special Definitions
    - 5.1 Object
    - 5.2.1 Structural Universe
    - 5.2.2 State
    - 5.2.3 Local admissibility
    - 5.2.4 Decomposition as a type function
    - 5.2.5 Type preservation
    - 5.2.6 Recursive levels
    - 5.2.7 Parent map
    - 5.2.8 Conservation of information-theoretic
    - 5.2.9 Non-uniformity
    - 5.2.10 State and Structure
    - 5.2.11 Intuitive Interpretation
    - 5.3 Relations R
    - 5.4 Id_i, Data (D_i), Attributes (At_i)
    - 5.5 Properties (P_i)
    - 5.6 Obs_i and Validity
    - 5.7 Local and Global Validity
    - 5.8 V_{Global} and V_{Local} Verification Functions

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

- 5.8.1 Local Data Convergence Function
- 5.9 Protocol Parameters (ϵ_G, ϵ_N, ϵ_C, λ, I_0)

6. State Transition Law
7. Definition of Knowledge and Information Levels
8. Information Density
9. Axioms
    - 9.1 Axiom I
    - 9.2 Axiom II
    - 9.3 Axiom III
10. Information Restriction Principle
11. Operational Flow
12. Discussion Shannon Entropy
13. State Rejection
14. Limits of the Theoretical Model
15. Further Developments
16. The Emergent Security Theorem
    - 16.1 Introduction
    - 16.2 Development
    - 16.3 Conclusion
17. Considerations on Depend Collusions
    - 17.1 Introduction
    - 17.2 Development
    - 17.3 Relations with Inference
    - 17.4 Conclusion
18. Scientific Positioning and Relationship with Existing Frameworks

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

- 18.1 Introduction
- 18.2 Information Theory
- 18.3 Verifiable Secret Sharing and Threshold Encryption
- 18.4 Zero- Knowledge Proofs
- 18.5 Secure Multiparty Computation
- 18.6 Byzantine Fault Tolerance and Consensus Protocols
- 18.7 Bitcoin and Blockchain Security
- 18.8 Epistemic logic
- 18.9 Formal Methods
- 18.10 Complex Systems and Emergence
- 18.11 Quick Comparative Summary
- 18.12 Distinctive Contribution of the CNVS
- 18.13 Final Positioning Declaration

Appendice

Appendix Q - Tables λ – CNVS Fragmentation Tables Based on $\mathcal{T}$

Appendix A - Complete Mathematical Formulations

Appendix B - Complete Symbol Table

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

1. Introduction

This article formalizes an asymptotic theoretical System that aims to be an ideal model for attesting and verifying empirical evidence of the existence of an object external to the System itself, starting from statements made within it.

In this paper, the object is understood in its extended form as any reality that can be supported by specific, unambiguous, and objectively identifiable empirical evidence.

In short, this theoretical model attempts to reconcile a specific truth statement with the objective verification of its consistency in empirical reality, regardless of the nature of the object, whether material or abstract.

In this sense, the CNVS framework addresses one of the oldest and most fundamental problems of human society: how to determine whether an object, event, or statement actually exists without relying on a specific authority. In modern distributed Systems, this challenge is known as the oracle problem.

CNVS is proposed as a formal Systems theory framework that offers a new general mathematical perspective to address this problem through recursive information decomposition, randomly distributed empirical verification, and Global validity emerging from the convergence of verification results with respect to input information.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

2. Abstract

This document defines a new class of Systems called **Closed Native Verification Systems (CNVS)**.

In these Systems, the existence, validity, and evolution of the state are determined exclusively by internal verification processes supported by randomly distributed external verification evidence and ultimately governed by deterministic invariants.
Although the objective verification processes remain external, their distribution—that is, the specific structure imposed by the System—reduces confidence in the execution of a rigid application scheme that limits the executor's discretion through the progressive reduction of Local and Global operational knowledge.
Specifically, the System seeks to converge declarations made in the form of user input requests and subsequent verification by external verification agents randomly selected by the System itself and operating according to the imposed constraints.
The External Verification Agents are tasked with verifying individual details of the information content of the User's input request, without knowing the result, and subsequently reporting them so that the System can cross-check the collected data to ensure complete consistency.
In this sense, the System aims to reveal the Truth of Existence within itself as a cross-verification process between independent and randomly distributed operational flows: the User input request and the Verification of the External Agents managed by the System itself.

The framework models the transformation of user-declared information into recursively decomposed structured elements, which are randomly assigned to External Verification Agents, each operating under strict protocol constraints. Local verification evaluates attribute-specific convergence between internally

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

declared data and observed proofs, while Global external validity emerges from the consistency of the reconstructed state and cannot be reduced to the conjunction of Local verifications. By integrating recursive decomposition, information-theoretic knowledge restriction, and emergent state validation, CNVS is proposed as a general mathematical framework that addresses the oracle problem, one of the key unsolved challenges in distributed Systems.

The Theory presupposes Three Fundamental Axioms, which are imposed as a Necessary Condition for classifying a System as a Native Verification System. The Axioms define:
- the Structural Properties of the entire System;
- the Emergent Behavior related to the Security and Verification Dynamics;
- the Principles that guide the Dynamics of External Processes and govern those of Internal Processes.

---

3. Problem Statement

One of the key unsolved problems in distributed Systems is the oracle problem: the difficulty of determining whether information from the physical world can be introduced into a digital System without relying on a single trusted source. Existing approaches, including multisource aggregation, cryptographic attestations, reputation mechanisms, consensus protocols, and secret sharing techniques, mitigate specific aspects of this challenge but do not eliminate the fundamental dependence on external trust.

In particular, existing Systems typically validate externally provided statements but do not model verification as an intrinsic condition for the existence of the

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

State. They generally do not impose a recursive structural decomposition of information, do not Systematically constrain the Local knowledge of individual Verification Agents, and do not formalize Global validity as an emergent property irreducible to the convergence of Local verification results.

Closed Native Verification Systems (CNVS) are proposed as a distinct theoretical framework in which user-declared information is recursively fragmented, randomly distributed to external verification agents under specific constraints, and reconstructed exclusively at the System level. Each Local verification evaluates the convergence between the internally declared value and independently observed evidence for a specific attribute. The System ultimately returns only a binary decision: the Object (also understood as the assertion) exists within the System or not.

By replacing unilateral trust with distributed empirical convergence under structural information constraints, CNVS are proposed as a general mathematical framework that addresses the oracle problem in a fundamentally different way from existing architectures.

In Traditional Systems that address the same problem, we find that:
- the truth about the existence of a state is a unilateral decision by a governing authority (total trust);
- verification of the object is entrusted to external auditors, who are at most governed by Global constraints or laws, with few Local references;
- External Verification Agents are entrusted with the discretionary task of issuing authoritative statements about the object that are difficult to contest;
- External Verification Agents are chosen by the governing authority, which also imposes the control rules;
- each external verification agent is always responsible for verifying the same specificity;
- each external verification agent has multiple pieces of information relating to the data of the Object of Verification;
- the existence of the state is admitted in the Traditional System as natively

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

imposed by the issuing authority;
- verification of the state follows the imposition of existence;
- The state is almost always unitary;
- The overall security of the System's interior decreases with the adoption and increase in the number of states, because the surface area for external attacks increases.

In a CNVS, the System inverts this logic:
- The truth about the existence of a state is not a decision unilaterally imposed by a governing authority, but the result of a mathematical process internal to the System;
- Verification is entrusted to Verification Agents external to the System, but strictly guided and governed according to the application scheme of a Distribution Model that uses information access restriction as a method of collusion control;
- The attestation of evidence issued by the external verifier must first pass strict internal verification checks to ensure the logical and structural consistency of its content with respect to the entire System;
- The state is not unitary;
- The External Verification Agent not only verifies the existence of evidence of an Object's specificity, but is also required by the System to provide data related to the assigned specificity. However, it does not know the data declared during the input phase when the User creates the Object Existence Verification request, thus allowing the System to compare these two entities through information convergence;
- the External Verification Agent is chosen randomly;
- the Object's specificity that the External Verification Agent is required to measure and control is chosen randomly by the System;
- Security emerges as a Property of the System and grows with its increased Adoption due to the System's very Structure.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

4. General Definitions

In this document, a **System** is defined as a Structured Set of Objects and Elements, regardless of Type, that have specific interrelationships and are governed by preliminary Assumptions that define its Functions.
A System is considered Closed when no External Intervention can alter its Structure, nor can it produce a result that conflicts with the preliminary Assumptions imposed through its Execution.

In this document, a System is defined as "**Natively Verified**" because the very Existence of all the Elements and Objects that comprise it is admitted only after an Objective Verification, followed by the Application of a Consistency Check of the issued Certifications.

In other words, no Element can "become" if it has not first been Verified.
In this document, a Natively Verified System is defined as Closed because no unauthorized External Intervention can alter its Structure, nor can it produce results that conflict with the preliminary Assumptions imposed through its Execution.

For the purposes of the **CNVS** model, Information (Ix) is defined as any ordered and finite binary representation admitted by the System. Physical objects, measurements, attestations, certificates, signatures and attributes, declarations, statements, postulates, sentences, and historical data are all processed only through their Encoded Representation as elements of {0, 1}*. Thus, the System does not directly process physical or immaterial Objects, but operates exclusively on their verifiable digital representation.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

5. Special Definitions

5.1 Object

Let
O_x
be the Object to be Verified,

let
$I(O_x)$
the complete intrinsic informational structure associated with the Object,

let
$\tilde{I}(O_x) = \text{Encode}\,(I(O_x))$ **(1)**
the encoded binary representation used by the System,

let
$E(t) = \text{Struct}\,(\tilde{I}(O_x), t),$ **(2)**
with Struct $(\tilde{I}(O_x))$ constituting the Set of all Structured Elements derived from the Object's binary representation at time t,

let

The Chain of Events Generated by the User Request:

$O_x \rightarrow I(O_x) \rightarrow \tilde{I}(O_x) \rightarrow E$ **(3)**

The System allows the User to initiate a Verification Instance on an Object for which Evidence of Existence is to be established.

The System responds by building a structured information representation and encoding it in binary format. It then recursively decomposes the information into operationally separate, epistemically incomplete, and structurally related fragments, according to a recursive scheme explained in the following

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

chapters.

A valid State emerges if and only if the provided information satisfies both Local and Global invariant verification through the distributed evaluation process performed by the System.

5.2.1 Structural Universe

Let

$$\boldsymbol{\mathcal{S}} := \mathrm{Id} \times \mathrm{D} \times \mathrm{At} \times \mathrm{Obs} \times \mathrm{P}(\mathbb{P}) \quad \textbf{(4)}$$

where

$\mathrm{Id} \subseteq \{0, 1\}^*$

$\mathrm{D} \subseteq \{0, 1\}^*$

$\mathrm{At} \subseteq \{0, 1\}^*$

$\mathrm{Obs} \subseteq \{0, 1\}^*$

and $\mathbb{P}$

is the finite or implementation-defined set of allowed logical properties.

5.2.2 State

Let

$$S(t) := (E(t), R(t), C, V_{Local}) \quad \textbf{(5)}$$

with

$E(t) \subseteq \boldsymbol{\mathcal{S}}$

where E(t) contains only the Structured Elements that have passed the Existence Check, therefore corresponds the set

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

$E_{valid}(t)$

while when the Verification is in progress, all the Structured Elements are part of the temporary Set

$E_{temp}(t)$,

which is brought to $E_{valid}(t)$ if the Global Verification is successful at the end of the process, otherwise the Etemp(t) State is rejected,

$R(t) \subseteq E(t) \times E(t)$

$C = \{c_1, \ldots, c_m\}$,

let

$\bar{S} := Id \times D \times At \times Obs$,

$V_{Local} : \bar{S} \rightarrow \{0, 1\} \times P(\mathbb{P})$ **(6)**

where

$V_{Local}(\bar{s}_i) = (b_i, P_i)$, **(7)**

with

$\bar{s}_i = (Id_i, D_i, At_i, Obs_i)$

$b_i \in \{0, 1\}$

is the Local Boolean outcome, while P_i is the set of logical properties produced,

and in the end

$si = (\bar{s}_i , P_i)$.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

5.2.3 Local admissibility

Let

$$\mathrm{Adm}_L : \boldsymbol{\mathcal{S}} \times \mathbb{T} \rightarrow \{0, 1\} \quad \textbf{(8)}$$

be the Local Admissibility Function, where $\mathbb{T}$ denotes the temporal domain.

For every structured element s_i and time t,

$\mathrm{Adm}_L(s_i, t) = 1$

if and only if the Boolean component of the Local Verification Function is equal to 1.

Formally,

$$\mathrm{Adm}_L(s_i, t) = 1 \iff \pi_1(V_{Local}(\bar{s}_i(t))) = 1 \quad \textbf{(9)}$$

where π_1 extracts the Boolean component of the ordered pair returned by V_{Local},

with π_1 denotes the canonical projection onto the first component of the ordered pair returned by V_{Local}.

5.2.4 Decomposition as a typed function

Let

$$D_t : \boldsymbol{\mathcal{S}} \rightarrow \mathcal{P}_{fin}(\boldsymbol{\mathcal{S}}) \quad \textbf{(10)}$$

be the Decomposition Function, where $\mathcal{P}_{fin}(\boldsymbol{\mathcal{S}})$ denotes the set of all finite subsets of the Structural Universe.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

For every structured element $s \in \mathcal{S}$,

$D_t(s) = \{s_1', s_2', ..., s_k'\}$

with

$1 \leq k < \infty$.

The function D_t maps a structured element into a finite set of structured descendants generated by recursive decomposition.

n.b. In this document, the term *Terminal Fragment*, or simply *Fragment*, refers to a terminal Structured Element sss produced by the recursive decomposition process D_t, for which no further operational decomposition is applied within the System.

5.2.5 Type preservation

Let

$\mathcal{T}$

be the set of all admissible Structural Types recognized by the System.

where

type : $\mathcal{S} \rightarrow \mathcal{T}$ **(11)**

assigns to each structured element its structural type.

Each structured element is associated with exactly one structural type through the Type Function,

therefore

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

for every structured element $s \in \boldsymbol{S}$ and every descendant $s' \in D_t(s)$,

$type(s') = type(s)$.

5.2.6 Recursive levels

Let

$E^{(0)}(t)$,

the nodal level

$$E^{(n+1)}(t) = \bigcup_{s \in E^{(n)}(t)} D_t(s) \quad \textbf{(12)}$$

and

$\forall n \in N$

then

$0 \leq n \leq Depth(t)$

5.2.7 Parent map

Let

$par_t : E^{(n+1)}(t) \rightarrow E^{(n)}(t)$

such that

$$par_t(s') = s \iff s' \in D_t(s) \quad \textbf{(13)}$$

5.2.8 Conservation of information-theoretic

Let

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

$\mathtt{I} : \boldsymbol{\mathcal{S}} \to \{0, 1\}^*$

be the Information Encoding Function.

For every structured element s, $\mathtt{I}(s)$ denotes its canonical binary representation.

The Chain of Events, from the previous definitions, becomes:

$O_x \to I(O_x) \to \tilde{I}(O_x) \to E \to s \to \mathtt{I}(s)$.

For the reconstruction function

Let

$$\mathrm{Rec}_t : \mathcal{P}_{\mathrm{fin}}(\boldsymbol{\mathcal{S}}) \to \boldsymbol{\mathcal{S}} \qquad \textbf{(14)}$$

be the Reconstruction Function.

The function Rec_t recomposes a structured element from a finite set of structured descendants generated by the decomposition process.

For every structured element $s \in \boldsymbol{\mathcal{S}}$,

$\mathtt{I}(\mathrm{Rec}_t(D_t(s))) = \mathtt{I}(s)$.

The Chain of Events backwards becomes:

$\mathrm{Rec}_t(D_t(s)) \to \mathtt{I}(\mathrm{Rect}(D_t(s))) = \mathtt{I}(s)$

5.2.9 Non-uniformity

Let

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

$$\exists s_a', s_b' \in D_t(s) : |I(s_a')| \neq |I(s_b')| \quad \textbf{(15)}$$

but

$$\sum_{s' \in D_t(s)} |I(s')| \geq |I(s)|$$

Since the System uses IDs, keys, signatures and reconstruction markers, the sum of the fragment bits cannot simply be equal to the original information, otherwise there is overhead,

therefore

Let

$$I_{payload} : \boldsymbol{\mathcal{S}} \to \mathbb{N}$$

be the Payload Information Function.

For every structured element s, $I_{payload}(s)$ denotes the number of bits corresponding to the effective informational content inherited from the original object or claim,

Let

$$I_{metadata} : \boldsymbol{\mathcal{S}} \to \mathbb{N}$$

be the Metadata Information Function.

For every structured element s, $I_{metadata}(s)$ denotes the minimal operational information required for fragment identification, assignment consistency, and participation in the distributed reconstruction and verification process.

Therefore we have:

$$I_{payload} \neq I_{metadata}$$

and

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

for every structured element $s \in \mathcal{S}$,

$$|\mathbb{I}(s)| = I_{payload}(s) + I_{metadata}(s)$$

with

$$I_{metadata}(s) > 0.$$

It is clear at this point that the recursive decomposition of the graph must necessarily admit a limit: fragmentation is permissible if and only if the ratio between payload and metadata remains above a Critical Threshold $\chi > 0$, which guarantees that the informational payload remains operationally significant.

If, in fact, the System requires a volume of metadata to manage the Fragments that exceeds the transported information content, it is clear that the System, in addition to becoming slow, becomes useless.
This limit depends on the overall informational complexity and structural decomposition of the Object representation.

In the Operational Flow chapter, a limit of 4 or 5 node levels was set, while the number of Fragments was relegated to formula (86).
The use of the System Control Parameters (λ, I_0, ρ_C) allows the number of terminal Fragments to be regulated efficiently, thus adapting it to the I_{min} value, while respecting the constraints imposed by the metadata volume.

It should be noted, however, that the role of metadata, and its relative weight, is significantly reduced in the CNVS System, because much of the information needed to manage Terminal Fragments is contained in the R function.
The System is not a metadata-rich reconstructive System, but rather an invariant System driven by relationships.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

5.2.10 State and Structure

The State of the System at time t is not a mere collection of elements, but a typed relational structure whose elements are admitted only insofar as they preserve the structural schema imposed by the System and satisfy the Local verification condition.

Let

$\mathcal{S} = Id \times D \times A\,t \times Obs \times P(\mathbb{P})$

be the structural universe of admissible elements. Each structured element is therefore a tuple

$s_i(t) = (id_i(t), D_i(t), At_i(t), Obs_i(t), P_i(t)) \in \mathcal{S}$

The State at time t is defined as

$S(t) := (E(t), R(t), C, V_{Local})$,

where

$E(t) \subseteq \mathcal{S}$

$R(t) \subseteq E(t) \times E(t)$

$C=\{c_1,\ldots,c_m\}$

is the finite set of deterministic invariants imposed by the System, and V_{Local} is the Local verification function.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

The System is recursive because every structured element may generate a finite set of structured descendants without changing its structural type, while reducing its informational scope.

Formally, let

$D_t : \mathcal{S} \rightarrow P_{fin}(\mathcal{S})$

be the decomposition function.

For $\forall s \in \mathcal{S}$

$D_t(s) = \{s_1', \ldots, s_k'\}, \quad 1 \leq k < \infty$

and

$\forall s' \in D_t(s), \ type(s') = type(s)$

Therefore, decomposition modifies the informational granularity of the element, but not its structural nature. The fragment is not a different kind of entity: it is the same structural entity at a lower informational scale.

The decomposition levels are defined recursively:

$E^{(0)}(t) = E_N(t)$

$E^{(n+1)}(t) = \bigcup_{s \in E^{(n)}(t)} D_t(s)$

In the operative model, $n \leq Depth(t)$.

In the ideal model, the recursive structure is defined for arbitrary finite n.

The conservation of information is expressed through a reconstruction operator Rec_t, such that

$\mathbb{I}(Rec_t(D_t(s))) = \mathbb{I}(s)$

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

Thus, non-uniform decomposition does not imply informational loss. It only implies that reconstruction becomes more constrained, since terminal Fragments may carry unequal informational weight under restricted relational accessibility:

$\exists\, s_a', s_b' \in D_t(s) \;:\; |I(s_a')| \neq |I(s_b')|.$

5.2.11 Intuitive Interpretation

Let E(t)
be the complete set of all Structured Elements s_i of the Encoded Information $\tilde{I}(O_x)$ flowing into the System at time t,

Let R(t),
be the Relations among the Structured Elements at time t,

Let C,
be the Deterministic Invariants imposed by the System,

with $C = \{c_1, c_2, ..., c_m\}$,

Let $V_{(\text{Local or Global})}$,

be the System Verification Function (Local or Global) for the Determination of Logical consistency before the declaration of the Existence of State.

Since Structured Elements are not atomic, but may be fragmented randomly without losing their Characteristics and Properties,
thus rendering the System Recursive, we will therefore have:

In s_i:

D_i = the data (e.g., 2 parts, 2 mL, 4 Hz, 4.2 grams, “Saturday afternoon,” “at 3 a.m.,” “objectively ugly,” 90 cm, 40 km/h, etc.)
At_i = the D_i data attributes (quantity, internal state, timestamp, turbidity,

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

volume, weight, time, “clothing worn,” “body temperature recorded,” speed, distance, etc.)
Obs_i = the Validation statements (audits, certificates, signatures, attestations of truth, testimonies, measurements, observations, clarifications, physical findings, etc.) reported by the Verifier and which will converge on the D_i data.

$\mathbb{P}$ = Properties, {Valid, Invalid, Transferable, Blocked, Verified, Suspended, Redeemable, Broken, Misinterpreted, Ambiguous, Accepted, Non-Final, Inconclusive, Doubtful, etc…}

with

$P_i = f(D_i, At_i, Obs_i)$, or more precisely:

$$P_i = V_{Local}(D_i, At_i, Obs_i). \quad \textbf{(16)}$$

With,

$s_{ij} = (id_{ij}, D_{ij}, At_{ij}, Obs_i, P_{ij})$

with $\forall n, \forall i, \forall j \in \mathbb{N}$

that is, the same Structural Attributes, same schema.

Therefore:

$\exists \{s_{i1},...,s_{ik}\} \subseteq E(t)$

i.e.:

$\{s_{i1}, s_{i2}, s_{i3}, ..., s_{ij}\} \subseteq E(t)$.

Hence

$$\forall s_i \in E(t), \exists \{s_{ij}\} \subseteq E(t) \quad \textbf{(17)}$$

such that:

$$\text{structure}(s_{ij}) = \text{structure}(s_i) \quad \textbf{(18)}$$

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

Furthermore, during the Decomposition process, uniformity is not necessarily maintained, such that the obtained sub-elements preserve equal portions of the original Structured Element, but the Distribution of the Decomposition must be non-uniform.

Non-uniform Decomposition among Fragments of the same Level does not affect the Conservation of Information, but certainly affects the technical operation of its Reconstruction.

Formally:

$$\forall s_i \in E(t), \forall n \in \mathbb{N}\ , \exists \text{ a decomposition of depth n} \quad \textbf{(19)}$$

or more precisely

$$\forall s_i \in E(t), \forall\ n \in \mathbb{N}, \exists \{s_i^{(1)}, s_i^{(2)}, ..., s_i^{(n)}\} \quad \textbf{(20)}$$

Within each sub-level, the portion of the Element from the preceding Level is not distributed uniformly.

Nb. In $s_{ij}^{(1)}$, the notation j as a subscript referring to subsequent Decomposition levels for the same fragment of Order i is replaced by the superscript (1), which assumes the same meaning as j.

In the formulae (17) both notations are reported.

5.3 Relations

Let

$$R(t) \subseteq E(t) \times E(t) \quad \textbf{(21)}$$

where,

$E(t) \times E(t)$

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

represents the Cartesian product of all possible pairs of the Structured Elements of Set E at the same time t, where R(t) is given by the structurally admissible directional relations.

If $E(t) = \{s_1, s_2\}$,

therefore

$E(t) \times E(t) = \{(s_1, s_1), (s_1, s_2), (s_2, s_1), (s_2, s_2)\}$.

The System does not necessarily provide for all possible Relations; only some may be permitted.
For example

let

$R(t) = \{(s_1, s_2), (s_2, s_2)\}$

The Directional Relation could only exist between
the pair s_1 and s_2
and between
the pair s_2 and s_2,
which would therefore only be in relation to itself.
The System therefore does not fully recognize all possible pairs of the Cartesian product E x E, but only some.
It is the specific implementation of the System, as well as the Structural Definition of its Constituent Elements, that decides on the Relations.
There are no Constraints Imposed by the System on the Relations, other than those of Logical Consistency to the Invariants and to the Global Structure of the System itself.

In the System

Relations in the System are always directional (non-symmetric),
since verification depends on directional ordering,
such that

$(s_j, s_i) \neq (s_i, s_j)$.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

Put another way,

let

$\forall(s_i, s_j) \in R(t)$,

then

$(s_i, s_j) \in R(t) \nRightarrow (s_j, s_i) \in R(t)$ **(22)**

All Relations allowed in the System must be
Logical,
Structural,
necessarily Directional (non-symmetric),
and do not carry quantitative weights.
Relations are therefore Directional, and Connections are not weighted between Structured Elements.

Non-uniform Fragment decomposition does not alter the admissibility or structural validity of Relations between Fragments belonging to the same decomposition level.

The pair

$(E(t), R(t))$

therefore defines a directed relational graph over the Structured Elements of the System.
Recursive decomposition induces a hierarchical graph structure in which Structured Elements may generate descendant nodes through recursive fragmentation while preserving structural type consistency.

In this representation:

- each structured element $s_i \in E(t)$ corresponds to a vertex (node);
- each admissible relation $(s_i, s_j) \in R(t)$ corresponds to a directed edge;
- decomposition levels define a hierarchical structure in which nodes may generate descendant nodes through recursive fragmentation;

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

- nodal elements represent first-level vertices;
- terminal fragments represent vertices at deeper levels of decomposition.

The resulting Graph therefore captures both the structural relations among elements and the recursive topology of State decomposition.

Consequently,
although terminal Fragments remain epistemically incomplete, the relational structure R(t) preserves the structural dependencies required for global consistency and reconstructive coherence.

5.4 Id_i, Data (D_i) e Attributes (At_i)

Id_i is the unique identifier of the Structured Element s_i, which also contains the minimal encoded identifiers required for recursive fragment association and reconstructive consistency within the System (see the chapter dedicated to the System's Operational Flow).

Generally speaking, the System does not maintain a complete explicit topological map of Fragment reconstruction. Instead, reconstructive coherence emerges from the relational structure R(t), invariant consistency, and the recursive dependency structure preserved across decomposition levels.

At each recursive decomposition step, the System associates the Structured Element $s_i^{(n)}$ with unique encoded identifiers that never leave the System.

These keys provide an internal set of information for the exclusive use of the System. In a reconstruction process, or more simply to enable the formulation of the precise task to be assigned to the External Verification Agent, it specifically uses Id_i to identify the relational dependencies associated with the Structured Element within R(t).

Here we would like to clarify that the reconstruction of the complete informational content carried by the current level is necessary for the reconstruction of the information of the preceding level.
This is because only after the total reconstruction of the current level is it

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

possible to reveal the structure of the parent decomposition level, with the identification of $Id_i^{(n-1)}$.

The D_i data follow the same decomposition recursion as s_i.
The same applies to the D_i data as for Id_i, namely that its binary content is progressively reconstructed at each level n = n - 1 for each recomposition cycle, such that the complete reconstruction of a preceding decomposition level requires the coherent reconstruction of the informational content preserved at the current level.

The Di content remains internally restricted within the System and is not directly accessible to External Verification Agents.
In other words, the contents of Di never leave the System.

Let's look at the Attributes.

Let

$$At_i = \{a_{i1}, a_{i2}, a_{i3}, ..., a_{ik}\} \quad \textbf{(23)}$$

Attributes are observable data, not logical.

The Attribute is therefore a Category or Class of Objective Description of the Structured Element at the considered level of Decomposition.

Attributes may exhibit Internal Dependencies, but these cannot be modeled as Relationships within the System and are defined by the specific Implementation.

5.5 Properties (P_i)

Properties are logical conditions that act on the single Structured Element,

Properties can be, for example,
valid, transferable, blocked, verified, suspended, accepted, questionable,

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

corrupted, etc.
Properties are not data, but rather Evaluations that depend exclusively on the System.

Let

$$P_i \subseteq \mathbb{P} \quad \textbf{(24)}$$

where $\mathbb{P}$ denotes the set of admissible logical Properties defined by the System.

Properties are logical outcomes generated by the Local Verification Function and used during Global verification.

Therefore

$$P_i = \pi_2(V_{Local}(\bar{s}_i)) \quad \textbf{(25)}$$

Properties are generated by the Verification Function applied to the Structured Element.
Properties are not limited to the definition of Validity, but also includes other logical conditions of the System that emerge from the development of its application.
For example, a si could be Valid, but also Blocked and Non-Transferable.
The formal definition of admissible Properties depends on the specific implementation and application domain of the System.

It is not possible to exhaust this topic in this document, so please refer to the Implementation for a complete description of the admissible Property space.
In this document, however, we can insist that the Property is not intrinsically given, but must emerge from the outcome of the Verification Function applied to the State of the Structured Element.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

The following paragraph will further clarify the role of the Verification Function within the System.

5.6 Obs_i and Validity

Let Obs_i
denote the set of declarative certifications issued by the External Verification Agent,

let

$s_i = (\bar{s}_i, P_i)$,

and

$\bar{s}_i = (Id_i, D_i, At_i, Obs_i)$. **(26)**

In short, Obs_i represents declarations, measurements, audits represents of Declarations, Measurements, Audits, Signatures, various Certificates, Testimonials, Clarifications, and Records, within the External Verification Process, and is provided by the External Verification Agent.
It contributes to the Evaluation, but does not exclusively determine it, since it originates externally to the System, even if produced under System-guided constraints.

Obs_i is therefore an input provided from outside the System by the External Verifier, who submits observational evidence that is subsequently evaluated through the local application of the Verification Function V.
The Verifier records the data requested by the System, corresponding to the specific verification assignment randomly allocated to it.
The reported data refers exclusively to the Attribute class Ati assigned to the Verifier for the current decomposition level.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

The data entered into the System will be compared through the Local Verification Function V_{Local} to verify consistency with the corresponding information preserved in D_i and with the logical constraints imposed by the System.
For example, the assigned task may involve measuring a physical quantity or describing an observable attribute of the Object or one of its decomposed parts.

The observational evidence recorded in Obs_i, relating to the weight of the Object or the color of the Object (or one of its parts), is compared with that originally recorded in the original submitted Object information through the application of V_{Local} function.

In detail, the System calculates the magnitude of any deviation found and checks whether it is statistically significant.
If the detected deviation is not statistically significant, the V_{Local} function returns TRUE; otherwise, the Local Verification fails and the V_{Local} function returns FALSE.

In summary:

$Obs_i \rightarrow V_{Local}(\bar{s}_i) \rightarrow P_i \rightarrow V_{Global}(S)$.

5.7 Local and Global Validity

The Verification Functions V_{Local} and V_{Global} are applied at different structural levels of the System.
This produces two distinct verification domains: Local and Global.

$\mathbf{Local_{Validity}}$

Local Validity is a Logical Property referred to the Single Structured Element, such that

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

$V_{Local}(s_i^{(n)}(t))$

where

the Structured Element s_i is verified individually at time t by the V_{Local} Function, which evaluates its local logical admissibility, Attributes, convergence between Obs_i and D_i, and Local and Logical Consistency.
The V_{Local} function is then applied successively to all intermediate states up to the Node, such that

$V_{Local}(s_i^{(n)}(t))$,
for each $n \rightarrow n-1$.
The cycle repeats until the nodal level $E^{(0)}(t)$ is reconstructed.

$Global_{Validity}$

Global Validity is a logical property emerging from the consistency of the application of all the verification functions for each individual element and the resulting relationships between them and with respect to C.
Let

$V_{Global}(S(t))$

the Verification Function of the entire State S at time t, this Function checks the Relations R, the Invariants C, and the Consistency between the Structured Elements already verified by V_{Local}.
It is possible that

$V_{Local}(s_i) = TRUE \ \forall_i$

while

$V_{Global}(S) = FALSE$, if the global State violates one or more invariants in C.

Therefore

$V_{Local} \neq V_{Global}$

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

These are two Verification Functions applied to two different levels of the Structure, and therefore respond to different satisfaction criteria for successful completion of the inspection.
The relationship between the two, therefore,
is not

$V_{Global}(S) = \Lambda_{si \in E(t)} V_{Local}(s_i)$,

but is

$V_{Global}(S) \neq \Lambda_{si \in E(t)} V_{Local}(s_i)$ **(27)**

that is, the Verification Function applied to the Whole State (Global) is not equivalent to the Verification Function applied at the level of the Structured Element or its Fragments (Local), but we will have

$V_{Global}(S) = f(E, R, C)$.

Therefore,

Global Validity cannot be reduced to the conjunction of Local Validity conditions, since the Global Verification process additionally depends on relational consistency and invariant satisfaction across the entire State.

local ≠ global domain.

5.8 V_{Global} and V_{Local} Verification Functions

Given the Structured Element definition

$s_i = (\bar{s}_i, P_i)$,

$\bar{s}_i = (Id_i, D_i, At\ i, Obs_i)$,

and

$\forall\, n \in \mathbb{N}, \exists\, s_i^{(n)} \in E(t)$,

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

The V_{Local} Verification Function is applied recursively to each Structured Element.

Thus, the Local Verification process is expressed as

$V_{Local}(s_i(t))$,

with

$\forall s_i \in E(t), V_{Local}(s_i) \in \{TRUE,FALSE\}$.

The Function verifies the validity conditions of all parameters included in s_i, i.e., id_i, D_i, At_i, Obs_i, generating the property P_i.

In detail, each of these parameters is verified for admissibility, internal consistency, plausibility of observational assertions formulated by the external verification agent or its measurements, internal logic of the assertions, and correspondence of the expected attributes At_i with those identified by the User when creating the object existence verification instance.

For example, it may happen that the parameters D_i, At_i, and Obs_i are all correct and perfectly aligned (e.g., At_i = "shirt color", D_i = "red", Obs_i = "red"), but the System detects that the value of Id_i (unique identifier) is not possible because it is expressed in an inadmissible formulation.

Or it may happen that the Parameters id_i, At_i, and Obs_i are correct but the Value of D_i is incorrect because it exceeds the maximum expected Value for the Decomposition Level n considered (if the sum of all the values of D_i at decomposition level 3 has reached a value $D_i = 101$, and the sum of all the values of D_i from the previous decomposition level 2 was equal to 100, a structural inconsistency necessarily emerges, because the decomposition violates informational conservation constraints, or simply because we have an Obs_i declaration that does not coincide with D_i).

In this scenario, while an adversarial majority might force the Local verification to return TRUE ($V_{Local}(s_i) = \{TRUE\}$), the structural anomaly (e.g., the sum

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

evaluating to 101 instead of 100) will inevitably emerge during the aggregation phase. Consequently, the application of the Global Verification Function to the candidate state, $V_{Global}(S(t))$, will evaluate to FALSE, instantly triggering state rejection despite the apparent correctness of the isolated Local parameters submitted by the colluders.

Formally,

let

$V_{Global} : S \rightarrow \{TRUE, FALSE\}$

where S denotes the space of admissible States , such that

$S(t) = [E(t), R(t), C, V_{Local}]$,

then we have

$V_{Global}(S(t)) = TRUE$,

if and only if all the Conditions are satisfied.
This deterministic Function evaluates whether a State satisfies the System constraints,
i.e.

$\forall s_i \in E(t), V_{Local}(s_i) = TRUE$

R(t) is consistent
and

$\forall c_j \in C, c_j(S(t)) = TRUE$

therefore,

$V_{Global}(S(t)) = TRUE \Leftrightarrow [\forall s_i \in E(t), V_{Local}(s_i) = TRUE] \wedge Consistent(R(t)) \wedge [\forall c_j \in C, c_j(S(t)) = TRUE]$. **(28)**

The outcome of the Verification Function determines Local or Global Validity depending on the structural domain to which it is applied.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

The Verification Functions $V_{Global\ and}$ V_{Local} are always deterministic, because identical States necessarily produce identical verification outcomes.
Since the $V_{Global\ or}$ V_{Local} Function is defined for all States, we have

$\forall\ S(t) \in S,\ V_{Global}(S(t)) \in \{TRUE, FALSE\}$

Furthermore,

$E(t) \subseteq \{\ s \mid\ V_{Local}(s) = TRUE\}$ **(29)**

$V_{Global}(S) \neq \bigwedge_{si \in E(t)} V_{Local}(s_i)$

Brief summary of the operational flow

$O_x \rightarrow I(O_x) \rightarrow \tilde{I}(O_x) \rightarrow E(t) \rightarrow \bar{s}_i \rightarrow V_{Local} \rightarrow P_i \rightarrow V_{Global}(S)$

5.8.1 Local Data Convergence Function

$V_{Local}(Id_i, D_i, At_i, Obs_i) = (b_i, P_i)$

where

V_{Local}'s specific task is to evaluate convergence, such that

$Conv_{At_i}(D_i, Obs_i)$

that is, the comparison between the data declared by the user D_i and the data observed by the External Verification Agent Obs_i, with respect to the specific attribute At_i.

The Local Verification Function does not merely check the formal correctness of the structured element.

Its main task is to evaluate the convergence of the data between the value declared by the user, D_i, and the value observed and reported independently by the external verification agent Obs_i, with respect to the specific attribute class At_i.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

Formally,

Let

$Conv_{At_i} : D \times Obs \rightarrow \{0, 1\}$

be the Attribute-Dependent Convergence Function.

For every pair (D_i, Obs_i), the function returns 1 if the value declared by the User and the value independently observed by the External Verification Agent are compatible under the admissibility criteria associated with the attribute At_i, and 0 otherwise.

Therefore

$Conv_{At_i}(D_i, Obs_i) = 1$

if and only if the value reported by the External Verification Agent is compatible with the value declared by the User under the admissibility criterion associated with At_i

Therefore:

$V_{Local}(Id_i, D_i, At_i, Obs_i) = (b_i, P_i)$

with

$b_i = Conv_{At_i}(D_i, Obs_i) \wedge Form(Id_i, D_i, At_i, Obs_i)$

where,

Form denotes the formal admissibility check of the tuple.

For quantitative attributes, convergence may be expressed as:

$Conv_{At_i}(D_i, Obs_i) = 1 \iff |D_i - Obs_i| \leq \varepsilon_{At_i}$

where ε_{At_i} is the admissible tolerance threshold associated with the attribute At_i

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

For qualitative attributes, convergence may be expressed through an attribute-specific equivalence relation:

$Conv_{At_i}(D_i, Obs_i) = 1 \iff D_i \equiv_{At_i} Obs_i$

Thus, Local Validity is not produced by unilateral declaration, nor by external attestation alone, but by the convergence between the User's declared data and the External Verification Agent's observed data under the semantic constraint imposed by the assigned attribute.

Finally

Let

$Form : Id \times D \times At \times Obs \rightarrow \{0, 1\}$

be the Formal Admissibility Function.

Form evaluates the syntactic correctness, structural consistency, and implementation-specific admissibility of the tuple (Id_i, D_i, At_i, Obs_i).

The Boolean outcome of the Local Verification Function is defined as

$b_i = Conv_{At_i}(D_i, Obs_i) \wedge Form(Id_i, D_i, At_i, Obs_i)$.

5.9 Protocol Parameters (ϵ_G, ϵ_N, ϵ_C, λ, I_0)

"Protocol Parameters" are defined as the set of critical coefficients (ϵ_G, ϵ_N, ϵ_C, λ, I_0) established by the System to regulate the Information Density and Fragmentation Tolerance Thresholds. These parameters are not independent variables, but structural constraints that determine admissibility and rejection conditions across decomposition and verification processes.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

These parameters can be modified by the System or the Implementation to vary the expected thresholds and thus adjust the System Efficiency.
They will be discussed in the relevant chapters.

---

6. The State Transition Law

A state transition is defined as the evolution of the state from time t to time t + 1, such that

$$S(t) \rightarrow S(t + 1) \quad \textbf{(30)}$$

Let

$$T : S \times U \rightarrow 2^S$$

Given the Transition Function T, we have that
$S(t + 1) = T(S(t), U(t))$
where
$U(t)$ = external inputs (validations, events, etc.), or better

$$S(t + 1) = \{s_i(t + 1) \in T(S(t), U(t)) \mid V_{Local}(s_i(t + 1)) = TRUE\} \quad \textbf{(31)}$$

The System therefore generates new S at time t + 1, but there are no invalid intermediate states.
Only valid states that satisfy all the Conditions imposed for each constituent Structured Fragment become part of the System's State Set.
Furthermore, each S(t + 1) must also pass the Global Validity check,

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

so the condition must also be true:
$V_{global}(S(t + 1)) = TRUE$.

That is, given that
$S(t + 1) = S' \in T(S(t), U(t))$

it must satisfy that $V_{Global}(S') = TRUE$
with:

$S' \subseteq T(S(t), U(t))$ **(32)**

The Transition is therefore not free, but is constrained by verification.
In short, Transition States are constrained by the Verification Function, such that only Valid Structured Elements and Consistent States are admitted at each step, until they enter the Global System.

---

7. Definition of Levels of Knowledge and Information

In a CNVS system, accessible information is distributed according to the decomposition set by the system itself, so that at each subsequent level, the size of the new structured sub-element obtained is reduced, according to a recursive model that preserves the information structure, which is progressively made more fragmented.
This forces us to define two distinct informational domains, which in this section we prefer to call "planes", because the term "level" is reserved for the depth of the recursive decomposition:
The first plane, defined by the "I" notation, refers to the complete structure of

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

the Information Plane and will include, in addition to the semantic content of each "informational unit", all the information related to its reconstruction, its size, and its semantic and structural relationship with all the other "informational units" of the same level, those preceding it, and/or those following it.
A second plane, defined by the "I" notation, refers only to the semantic content of the payload and therefore ignores the structure of the graph that produced it.
This distinction between planes is fundamental to understanding the entire CNVS system.

Given that the system does not generate the Object O_x, nor its $I(O_x)$ information, but merely manages it after it has been entered by the User, each $I(O_x)$ therefore constitutes an input to an external request that uses the System as an accreditation device.
The latter determines whether this input request should be accepted or rejected.
The system will therefore receive a continuous flow of I(Ox) information, in the form of external inputs, which will inevitably increase the content of the System's Global Information, i.e., Ix at time t, because both the semantic content of the payload and the complexity of its graph will increase.
Following this logic, the verifier's knowledge can develop at different levels of the two possible information planes, with the same recursion.

Let's recall the Chain of Events:
$O_x \rightarrow I(O_x) \rightarrow \tilde{I}(O_x) \rightarrow E \rightarrow s \rightarrow I(s)$. **(33)**

It is important to note that within the System, information relating to both the graph and the payload content is not transmitted in directly readable form, but is encoded at source to make it cryptic.

The encoding is:

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

• Systemic
• Uniform
• Structural.

Therefore, the decomposed information will not reach the verifier in directly readable form, but must first be decrypted, as it will have inherited the encoding of the original information.

The System provides structured encoding as an additional security mechanism and, in ideal implementations, encryption may additionally be applied.

In an ideal model, decryption should be possible using unique keys provided exclusively to the verifier assigned to the specific execution fragment.

This way, even if an unauthorized Verifier were to access other Fragments, thus violating the System's exact distribution map, they would be unable to read their contents, as they do not possess the specific decryption keys for each Fragment.

In short, in the System the information is encoded and, in the ideal implementation, encrypted, so the Verifier's access to all the Fragments does not eliminate the encryption problem.

$Dec(\tilde{I}_x(t)) \neq I_x(t)$

for an unauthorized Local Verifier,
while the Information Encoding $I_x(t)$ can be expressed as $\tilde{I}_x(t)$ at time t, such that
$\tilde{I}_x(t) = Enc\ (I_x(t))$. **(34)**

For simplicity, the Model is formalized below by considering only the Final Encoding (Enc) notation = I_x without reporting the notation $\tilde{I}_x$ for the different

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

Information levels and planes.

In light of this, the semantic knowledge K_x of an external User at time t, limited to the semantic content level present in the System at level x, can be understood as an accessible set of information, such that

$$K_x(t) \preceq I_x(t) \quad \textbf{(35)}$$

Specifically, it is a subset of the information accessible to a subject, where

- $I_x(t)$ = Information existing as semantic content,
- $K_x(t)$ = knowledge accessible,

for each domain x, at time t.

The knowledge K_x of a User at time t extended to the global information level present in the System at level x, can be understood as the accessible set of information, such that

$$K_{\mathbb{I}x}(t) \preceq \mathbb{I}_x(t) \quad \textbf{(36)}$$

where

$\mathbb{I}_x(t)$ corresponds to the semantic content of the payload, of its entire graph including the relational structure and the metadata necessary for reconstruction.

The entire representation is always expressed within the system as an encoded and ordered binary sequence {0, 1}, both for I and for $\mathbb{I}$, such that

$$I_x(t),\ \mathbb{I}_x(t) \in (\{0, 1\}^*)^z \quad \textbf{(37)}$$

where:

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

• {0, 1}* is the set of all finite binary strings;

• x identifies the domain (Global, nodal, fragmented, Local);

• z, x ∈ ℕ

Let $\mathtt{I}_G(t)$ be the Global Information of the System at time t, such that

$$\mathtt{I}_G(t) \cong (E(t), R(t), C, V_{Local}) \quad \textbf{(38)}$$

$\mathtt{I}_G$ will comprise the complete graph structure of E(t) and the exact semantic content of each of its recursive decompositions, at every level along the entire graph, up to the complete payload.

Let

$K_{\mathtt{I}G}(t)$,

be the global knowledge accessible at time t, that is, the knowledge relating to the information $\mathtt{I}$ of the entire system at time t, from the payload to the complete graph structure of domain G,
such that

$$\mathtt{I}_G(t) \cong \mathtt{I}(S(t)) \quad \textbf{(39)}$$

where $\mathtt{I}_G$ corresponds to the information:
- of the entire structured set E(t),
- of all the relations R(t)
- of all the constraints (C) imposed by the system,
we will therefore have

$$K_{\mathtt{I}G}(t) \preceq \mathtt{I}_G(t). \quad \textbf{(40)}$$

The same relationship is found for the purely semantic information plane I, referring exclusively to the transported payload, devoid of its relations, invariants, and the complete structure of its graph. Therefore, we have symmetrically:

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

$I_G(t) = I(E(t))$, but not
$I_G(t) \neq I(S(t))$, because the State also includes Invariants and Relations.
Therefore,
the overall semantic knowledge will be:

$$K_G(t) \preceq I_G(t). \quad \textbf{(41)}$$

The nodal knowledge K_{IN} at time t refers to the information I_N relating to the first node at time t, and therefore to all subsequent recursive decompositions that derive from it up to all terminal fragments, and includes both their semantic content and the structure of the entire descendant graph generated from that node,
where N_i constitutes the first node, so we will have

$$\mathbb{I}_N(t) \cong (\bigcup^{n}_{h=0} E_h^{(Ni)}(t), \bigcup^{h}_{h=0} R_h^{(Ni)}(t), C^{Ni}, V_{Local}{}^{Ni}) \quad \textbf{(42)}$$

and that is

$$\mathbb{I}_N(t) \cong \mathbb{I}(\mathrm{Desc}(N_i, t)) \quad \textbf{(43)}$$

where

$\mathrm{Desc}(N_i, t)$

denotes
the recursive set of all descendants of node N_i.

Then the relation will be:

$$K_{IN}(t) \preceq \mathbb{I}_N(t). \quad \textbf{(44)}$$

Symmetrically, the semantic Nodal knowledge will be

$I_N(t) \cong I(\mathrm{Desc}(Ni, t))$, and refers to the semantic content of all payloads derived from the recursive decomposition of the common node N_i.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

The decomposition-level knowledge $K_{Ih}^{(n)}$ at time t is that relating to the information $I_h^{(n)}$ of all the decomposed elements that are located at the same level h at time t, but excludes the recomposed elements of the previous levels and the decomposed elements of the subsequent levels, with reference to both the aggregated local semantic content and the local structure, including the related Relations, invariants and function V,
such that

$$I_h^{(n)}(t) \cong (E(t)^h, R(t)^h, C^h, V_{Local}{}^h), \quad \textbf{(45)}$$

$$K_{Ih}(t) \preceq I^{(h)}(t).$$

The semantic level knowledge $K_h(t)$ will be that referred to the aggregate semantic content plan of the payload of all the elements belonging to the same level h, excluding the graph and the structure of the Relations, such that

$$K_h^{(n)}(t) \preceq I_h^{(n)}(t) \quad \textbf{(46)}$$

The fragment-level knowledge K_{IF} at the local level includes the knowledge relating to the semantic information I_F of the terminal fragment, including all the Relations, invariants and the structure of the Fragment itself,
such that

$$I_F(t) \cong (s_{ij}^{(n)}(t), R_{s_ij}(t), C_{s_ij}(t), V_{Local}(s_{ij}, t)) \quad \textbf{(47)}$$

therefore

$$K_{IF}(t) \preceq I_F(t). \quad \textbf{(48)}$$

Symmetrically, the semantic knowledge $K_F(t)$ at time t, referring only to the elementary content of the non-aggregated payload, and without its graph and relations will be

$$K_F(t) \preceq I_F(t) \quad \textbf{(49)}$$

for $I_F(t) \cong I(s_{ij}^{(n)}(t))$. **(50)**.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

The fragment-level knowledge K_F of the single fragment of the last level assigned by the system at time t is also called local knowledge (K_{Local}), such that

$K_F(t) = K_{Local}(t)$, **(51)**

$K_F(t) \preceq I_F(t)$ **(52)**

This last definition is particularly important in CNVS Systems, because fragment-level knowledge presupposes a level of knowledge that is actually superior to that of the final verifier.
If everything described in the previous paragraphs is true, the verifier will not need to know the Relations, invariants, or the activity of the V_{Local} function on the assigned fragment, so its level of information will certainly be lower than K_{IF}, but it will also be lower than that of K_F.

The reason lies in the fact that the verifier does not even have complete semantic knowledge of the assigned fragment, as the System turns to it to entrust it with the task of performing an observation on the Object, but does not reveal the contents of the payload recorded in D_i, and does not know the definition of the specific class of the Attribute A_{ti}, but can only infer it.
It is the verifier who must record the observed data, generating the input Obs_i that is subsequently inserted into the System.
This is why the relation holds:

$K_{VF}(t) \npreceq I_F(t)$
(53)

The accessible information levels will have the following relationships:
$I_F(t) \subsetneq I_h(n)(t) \subsetneq I_N(t) \subsetneq I_G(t)$. **(54)**

Considering only the semantic level, we observe that the distribution of the I_F information relative to the fragments obtained $s_i^{(n)}$ does not follow a uniform pattern,
formally we will have

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

$D_{nu}(I_h^{(n)}(t), F_x(t)) \rightarrow \{I_{F,1}(t), I_{F,2}(t), \ldots, I_{F,Fx(t)}(t)\}$ **(55)**

where

- D_{nu}, is the nonuniform distribution function;
- $I_h^{(n)}(t)$, is the information of level n to be fragmented at time t;
- $F_x(t)$, is the number of terminal fragments required at time t;
- $I_{F,j}(t)$, is the information of fragment j.

The information distributed by fragmentation $I_h^{(n)}(t)$ is completely preserved, formally

$\Sigma^{F_x(t)}_{j=1} I_{F,j}(t) = I_h^{(n)}(t)$ **(56)**

where we have

$\exists j \neq k$ such that $I_{F,j}(t) \neq I_{F,k}(t)$.

Let $X_j(t)$

be the random variable representing the terminal fragment assigned to verify v_j at time t,

Where

- j identifies the verifier V_j;
- $X_j(t)$ is the fragment assigned to verifier V_j at time t,

let

$E^{(n)}(t)$ be the set of Fragments of level n at time t,

let

Ter(t) be the set of terminal Fragments of $E^{(n)}(t)$ of level n at time t, such that $Ter(t) = E^{(n)}(t)$,

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

where n denotes the terminal decomposition level at time t,
then formally we have
$X_j(t) \in Ter(t)$

and therefore
$K_{VF,\,j}(t) = \mathscr{I}\,(X_j(t))$,
where $\mathscr{I}$ represents the return function of the information content of the terminal fragment, such that

$\mathscr{I} : Ter(t) \rightarrow \{0, 1\}^*$.

If the fragment is encrypted and the verifier has the correct key,

$K_{VF,\,j}(t) = Dec\,(\bar{\mathscr{I}}\,(X_j(t)))$,

where $\bar{\mathscr{I}}\,(X_j(t))$ denotes the encoded representation of the information associated with the assigned fragment,

and when decryption is complete, we have
$K_{VF,\,j}(t) = \mathscr{I}(X_j(t))$.

The K_{VF} Knowledge available to Verifier v_j at time t will be limited to its own fragment but cannot be complete, because the External Verification Agent does not have full access to the I_{F_i} Information.
Specifically, the System manages the amount of fragmented information to share, based on "how much" information the external verification agent needs to know to perform data collection and its objective assessments, which will then be used by the System for verification through convergence of the D_i

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

Data.
Therefore

$$K_{VF}(v_j, t) \lneqq K_F(t) \preceq I_F(t) = \mathscr{I}(s_{ij}^{(n)}). \quad \textbf{(57)}$$

On the fragmented Knowledge plane, however, there is no access even to the Fragments that make up the Same Level.

Therefore, we will have

$$K_{VF}(t) \lneqq I_N(t), \quad \textbf{(58)}$$

but

$$K_{VF}(t) \neq I_N(t)$$

and

$$K_{VF}(v_j, t) \lneqq K_F(t) \subsetneq K_N(t). \quad \textbf{(59)}$$

Furthermore,

$$K_N(t) \subsetneq K_G(t). \quad \textbf{(60)}$$

The Knowledge Relationships will therefore be

$$K_{VF}(v_j, t) \lneqq K_F(t) \subsetneq K_h^{(n)}(t) \subsetneq K_N(t) \subsetneq K_G(t) \quad \textbf{(61)}$$

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

It should be noted that Fragment Assignment follows a random logic, such that a group of External Verifiers will not necessarily receive Fragments belonging to the same Level and the same Node, but External Verifiers may be assigned Fragments belonging to different Levels and Nodes.

That is,

$$\text{Assign}(f_{ij}^{(d)}) \rightarrow v_j \qquad \textbf{(62)}$$

where $f_{ij}^{(d)}$ is a terminal fragment of Node N_i, but Verifier vj can receive Fragments coming from different Nodes:

$$K_{VF}(t) \not\preceq \{I_F(N_i, t), I_F(N_l, t), \ldots\}$$

with

$$i \neq l$$

This increases information dispersion, because the Verifier does not work within a recognizable "nodal class."

Furthermore, since the information is encoded, we will have

$$\tilde{I}_x(t) = \text{Enc}(I_x(t))$$

for all subsequent levels of Information up to

$$\tilde{I}_F(t) = \text{Enc}(I_F(t)). \qquad \textbf{(63)}$$

The knowledge of the External Verification Agent will be:

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

$K_{VF}(t) \not\preceq \tilde{I}_F(t)$

with

$\mathrm{Enc}(I_F(t)) = \tilde{I}_F(t).$ **(64)**

if you have the correct encryption key.

Synthesis.

Knowledge Hierarchy Table

Knowledge Level	Meaning	Scope
$K_F(t)$	Fragment-Level Knowledge	Knowledge accessible at the terminal Fragment level
$K_h^{(n)}(t)$	Decomposition-Level Knowledge	Knowledge associated with all Structured Elements belonging to the same decomposition level h
$K_N(t)$	Nodal Knowledge	Knowledge associated with a Node and all its recursive descendants
$K_G(t)$	Global Knowledge	Knowledge associated with the entire System State

---

Semantic Information Hierarchy Table

Information Level	Meaning	Scope
$I_F(t)$	Fragment Semantic Information	Semantic payload of a terminal Fragment
$I_h(n)$	Decomposition-Level Semantic Information	Aggregate semantic payload of all elements at level h
$I_N(t)$	Nodal Semantic Information	Semantic payload recursively derived from Node N_i
$I_G(t)$	Global Semantic Information	Total semantic payload contained in the System

---

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

Structured Information Plane Table

Structured Information Plane	Meaning	Includes
$\mathtt{I}_F(t)$	Fragment Structured Information	Payload, metadata, relations, invariants, local verification structure
$\mathtt{I}_h^{(n)}(t)$	Level Structured Information	Structured representation of all elements at decomposition level h
$\mathtt{I}_N(t)$	Nodal Structured Information	Recursive descendant graph generated from Node N_i
$\mathtt{I}_G(t)$	Global Structured Information	Entire graph structure of the System State

---

Knowledge–Information Equivalence Table

Knowledge Domain	Corresponding Information Domain
$K_F(t)$	$I_F(t)$
$K_h^{(n)}(t)$	$I_h^{(n)}(t)$
$K_N(t)$	$I_N(t)$
$K_G(t)$	$I_G(t)$

Semantic vs Structured Information Planes

Plane	Notation	Meaning
Semantic Information Plane	I_x	Semantic payload content only
Structured Information Plane	$\mathtt{I}_x$	Semantic payload + graph structure + metadata + relations + invariants

---

Knowledge Accessibility Relation Table

Relation	Meaning
$K_x(t) \leq I_x(t)$	Accessible semantic knowledge is bounded by semantic information
$K_{\mathtt{I}x}(t) \leq \mathtt{I}_x$	Accessible structured knowledge is bounded by structured

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

Relation	Meaning
	information
$K_{VF}(t) < I_F(t)$	Verifier knowledge is strictly lower than fragment semantic information

---

Recursive Informational Inclusion Hierarchy

Hierarchy Type	Relation
Semantic Information Hierarchy	$I_F(t) \subset I_h^{(n)}(t) \subset I_N(t) \subset I_G(t)$
Semantic Knowledge Hierarchy	$K_F(t) \subset K_h^{(n)}(t) \subset K_N(t) \subset K_G(t)$
Structured Information Hierarchy	$\mathtt{I}_F(t) \subset \mathtt{I}_h^{(n)}(t) \subset \mathtt{I}_N(t) \subset \mathtt{I}_G(t)$

In short, Knowledge is not a simple quantitative reduction, but a Structural Reduction of the System's Representation, where it is possible to have a small but well-structured set of data, or by violating the System, to obtain much more, but poorly structured and poorly comprehensible data.

In both cases, the available information is incomplete and fragmented to the point that reconstruction by an unauthorized External Agent becomes difficult.

---

8. The Density of Information

Information Density ρ is a measure of the Dispersion of Information at the level in question and can be understood as the weight of the information at the level in question compared to the levels preceding it.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

Information Density is represented at multiple levels and does not depend on the Knowledge of the External Verifier, being a measure used by the System.

The Local Information Density ρ_N is given by the measure of the contribution of Fragmental Information to the Node (or Nodal Density), and constitutes the portion of information per single Fragment compared to the Nodal Information.

It is a measure of the Dispersion of Nodal Information in Fragmental Information, such that

$$\rho_N(t) = I_F(t)/ I_N(t) \quad \textbf{(65)}$$

ρ_N measures how much the information of that fragment weighs compared to the information of its node (going up through all the levels of decomposition up to $s_i^{(0)}$) or how much the nodal information has been dispersed;

The Global Information Density ρ_G referred to the fragmental information is given by

$$\rho_G(t) = I_F(t)/ I_G(t) \quad \textbf{(66)}$$

ρ_G measures the weight of a fragment relative to the Global information of the System.

Similarly, it is an index that measures the dispersion of Global information in

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

the terminal Fragments.

The Relative Nodal Density ρ_{GN} is a measure of the importance of nodal information relative to Global information and measures the weight of nodal information relative to Global information, such that

$$\rho_{GN}(t) = I_N(t)/ I_G(t). \quad \textbf{(67)}$$

The Composite Information Density ρ_C is the composite measure obtained by taking into account both the effects of the Global Density and those of the Local Density.
It is defined a weighted geometric mean of two positive ratios,
such that

$$\rho_C=(I_F/I_N)^{\alpha} * (I_F/I_G)^{\beta_\rho}, \quad \textbf{(68)}$$

where α and β_ρ are weighting coefficients that determine the relative contribution of the two information ratios, and that can be modified by the System or its Implementation to modify in the measurement of ρ_C the weight of the relative contributions of the Fragmental Information on the Nodal or Global Information, such that

$\alpha, \beta_\rho \in [0, 1]$ and $\alpha + \beta_\rho = 1$

and where

I_F/ I_N measures how much the fragment weighs in relation to the Node.

I_F/ I_G measures how much the fragment weighs in relation to System.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

So: ρ_C

measures a Composite Information Density, which takes into account both the Fragmental exposure within the Node and the relative importance of the Fragment in the Global System.

System Consistency is guaranteed by compliance with the Protocol Parameters (see relevant section).

The System rejects any state transition in which the detected density exceeds the established threshold coefficients ($\rho > \epsilon \implies$ Reject (S)) and violates the Conditions dictated by the Protocol Parameters.

In CNVS Systems, higher informational density corresponds to lower informational dispersion and therefore to greater semantic exposure of the Fragment relative to the protected Information domain,

therefore:

for

$0 < \rho_N(t), \rho_G(t), \rho_{GN}(t) \leq 1$

under admissible decomposition conditions.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

9. Axioms

9.1 Axiom I

In the System, only that which has been verified first and has obtained the Validity Property Condition exists.

Anything that has not been verified first, or that, despite being verified, has not obtained the Validity Property Condition, simply does not exist for the System; that is, it is not permitted in the System, therefore we have

$$s_i(t) \in E(t) \implies V_{Local}(s_i(t)) = \text{TRUE} \quad \textbf{(69)}$$

Candidate states S' produced by the Transition Function are admitted only after Local and subsequent Global verification.

This principle operates at both the Local and Global verification domains.

On the Local level, we can say that a Structured Element si(t) belongs to the Set of Structured Elements of the System at time t if and only if it satisfies its Local Verification Function V_{Local}.

$$0\text{Valid}\,(S(t)) \iff V_{Global}(S(t)) = \text{TRUE} \quad \textbf{(70)}$$

With

$$S(t) = (E(t), R(t), C, V_{Local})$$

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

The Global State is valid if and only if the Verification Function applied to the State evaluates to TRUE.
In a Native Verification System, Existence is a Verification-Dependent condition.
A Structured Element becomes part of the System State if and only if it has satisfied the Verification Function.
Invalid Elements will not constitute the System State.

By also applying the Transition Function, we will have:

$$S(t+1) = \{s_i(t+1) \in T(S(t), I(t)) \mid V_{Local}(s_i(t+1)) = TRUE\} \quad \textbf{(71)}$$

therefore,

$$V_{Local}(s_i(t)) = FALSE \Rightarrow s_i(t) \notin E(t) \quad \textbf{(72)}$$

in short,

$$s_i(t) \in E(t) \Longrightarrow V_{Local}(s_i(t)) = TRUE$$

$$V_{Local}(s_i(t)) = FALSE \Rightarrow s_i(t) \notin E(t)$$

$$S(t+1) = \{s_i(t+1) \in T(S(t), I(t)) \mid V_{Local}(s_i(t+1)) = TRUE\}$$

$$Valid(S(t)) \Longleftrightarrow V_{Global}(S(t)) = TRUE$$

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

9.2 Axiom II

The System requires that the State be represented through successive Levels of Structured Decomposition.

Let

$\forall\, s_i(t) \in E(t),\ \exists\, \{s_{i1}(t), \ldots, s_{ik}(t)\} \subseteq E(t)$

such that:

$s_i(t) \Rightarrow \{s_{i1}(t), \ldots, s_{ik}(t)\}.$

Let D be the Decomposition function

$\forall\, S(t),\ \exists$ a $D(S(t))$ Decomposition of depth n **(73)**

such that:

$D^{(n)}(S(t)) = \{s_i^{(n)}(t)\}$

the System imposes that the Decomposition Function D at time t increases the informational dispersion of the semantic payload of the Global Information I_G as the latter increases.

Given the Local Density ρ_N, Global Density ρ_G, Composite Density ρ_C,

The System adapts the Decomposition according to a Model that produces an increasing distribution of Information, with a progressive reduction of ρ_N, ρ_G, ρ_C, and therefore of the weight of Local Information over Global Information.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

That is, if the nodal information grows, the System is forced to increase the depth of the decomposition for successive levels as the number of Fragments increases, reducing the portion represented by each.

In summary:
The System imposes a Recursive Decomposition Constraint such that for each increase in both the Nodal Information In and the Global Information I_G, the relative share of Information accessible through the terminal Fragments is reduced such that,

$\rho_N(t) \leq \epsilon_N$ and $\rho_G(t) \leq \epsilon_G$

with

$\varepsilon_N, \varepsilon_G \in (0, 1]$

thus obtaining

$I_N(t2) > I_N(t1) \Rightarrow \rho_N(t2) < \rho_N(t1)$

and

$I_G(t2) > I_G(t1) \Rightarrow \rho_G(t2) < \rho_G(t1)$,

formally

$$(I_N(t2) > I_N(t1)) \ \vee \ (I_G(t2) > I_G(t1)) \Rightarrow \text{Depth}(N_i, t2) > \text{Depth}(N_i, t1) \qquad \textbf{(74)}$$

where ϵ_N, ϵ_G constitute two threshold coefficients, for ρ_N and for ρ_G, t1 and t2 represent two different time instants such that t1 < t2, and

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

Depth (N_i, t) $\in \mathbb{N}$,

- Depth(N_i, t) is the maximum depth reached by node N_i at time t,
- $N_i = s_i(0)$ is the Node, i.e., the structured Element of nodal level 0, as defined in chapter 5.1;

therefore,

Depth corresponds to the number of levels of the Recursive Decomposition applied by the System at time t in the Node i.

The System imposes that as nodal information increases, the fragmentary information density with respect to the node decreases below a coefficient ϵ_N;
The System imposes that as Global information increases, the fragmentary information density with respect to the Global System decreases below a coefficient ϵ_G.

In summary, the increasing Adaptive Decomposition therefore responds directly to the Composite Density, and indirectly to the Local and Global Densities, according to the following formulation:

Depth(N_i, t)$\uparrow \Rightarrow \rho_C(N_i, t)\downarrow$.

As the Composite Density increases, the Decomposition by successive levels increases, which therefore adapts to the increase in Global information by maintaining or reducing the Density levels.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

In fact, it is easy to demonstrate experimentally that the Densities ρ_N, ρ_G, and ρ_C progressively decrease as the System increases its Global and Nodal Information Level, regardless of any active interventions by the System itself.

This happens naturally, because Information Density is by definition a measure of a ratio, where the reference densities I_G and I_N are in the denominator:

In other words, if the size of the Information I_F at time t remains unchanged, while that of the Global Information I_G or Nodal Information I_N increases, each Fragment F containing the Information I_F, remaining of the same size, will have a smaller weight on the reference Information.

The effect will be a reduction in Density and therefore a Dispersion of Information, even without active interventions on the Fragments.
The size of each Terminal Fragment, which corresponds to the Fragmental Information I_F, is determined by the number of Fragments imposed by the System, because the greater the fragmentation depth, the smaller the size of the Terminal Fragments will necessarily be and therefore the smaller the relative contribution of I_F Information.

This behavior is a problem, because instead of forcing the Decomposition to successive levels, it stabilizes it.

after applying the decomposition we will have instead

$\text{Depth}(N_i, t)\uparrow \Rightarrow \rho_C(t+1) < \rho_C(N_i, t)$,

$\rho_C(N_i, t) > \epsilon_C \Rightarrow \text{Depth}(N_i, t+1) > \text{Depth}(N_i, t)$.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

The Condition “$\rho_C(N_i, t) > \epsilon_C$” then becomes the Internal input to the System which triggers further levels of Decomposition until the following effect is obtained:

$\text{Depth}(N_i, t + 1) > \text{Depth}(N_i, t) \Rightarrow \rho_C(N_i, t + 1) < \rho_C(N_i, t)$.

But the Condition “$\rho_C(N_i, t) > \epsilon_C$” is not easily verified experimentally because the growth of the I_F Information for a single Fragment does not follow that of the Information of the Whole System.

This happens because each Fragment remains isolated after decomposition, and while the System continues to expand into new nodes, the Fragments already emitted remain stationary.
Even when the decomposition depth is forced, the effect is an uncontrolled drop in the Local Density level, such that further fragmentation is increasingly difficult.
In Chapter 10, new control coefficients will be introduced to address the problem.

We can therefore conclude that the densities ρ_N, ρ_G, and ρ_C constitute a measurement tool internal to the System, usable in implementation, rather than a set of control parameters.

Let's look at the number of Fragments in detail.

Let

$\text{Fragments} \in \mathbb{N}$,
and recalling that $Ni := s_i^{(0)}$,
with $\text{Fragments}(N_i, t) = \gamma\,(I_N(t), \rho_c(N_i, t))$,

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

where
γ is a Monotone Function increasing with respect to information complexity,
or rather Fragments(N_i, t) = $|\{s_i^{(n)}(t)\}|$, that is,
Fragments(N_i, t) = $|E_i^{(Depth(Ni,t))}(t)|$
where
Fragments(N_i, t) corresponds to the number of Terminal Fragments generated by the Adaptive Decomposition process at time t in Node N_i,
with
- $E_i^{(d)}(t)$ which constitutes the set of Fragments of Node N_i at level d;
- Depth(N_i, t), the maximum depth of Node N_i,
- the modulus $|\cdot|$, the cardinality,
and
Fragments(t) is the total number of terminal Fragments of the System at time t, i.e.
Fragments(t) = Σ_i Fragments(N_i, t),
therefore the sum of the number of terminal Fragments of each Node,

such that

$$|\mathcal{T}(t)| = \text{Fragments}(t) \quad \textbf{(75)}$$

therefore,

$$\mathcal{T}(t) = \bigcup_i E_i^{(Depth(N_i,\ t))}(t) \quad \textbf{(76)}$$

the Decomposition applied to the Node must depend not only on the internal Information Breadth, but also on its relative Information weight with respect to the Global System.
So the level of Decomposition is a direct function of the Composite Density measure, which depends on the Local and Global Density Values, such that

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

$$\rho_C(N_i, t) \leq \epsilon_C \quad \textbf{(77)}$$

where

$\varepsilon_C \in (0, 1]$

> The System must force the Depth of Information Decomposition to maintain Dispersion and reduce Information Density.
>
> This happens when the Composite Density ρc increases, because it means that the Terminal Fragment weighs too much compared to the node/Global. Therefore, the System is forced to increase the Decomposition Depth to reduce that density and preserve the Structure.

Therefore,

$$\text{Depth}(N_i, t) = f(\rho_C(N_i, t)), \quad \textbf{(78)}$$

with f being a composite Density Function, and

$\rho_C(N_i, t) = g(I_F(t), I_N(t), I_G(t))$,
where g denotes the Composite Density Function previously defined in equation (53),

we will have

$$K_{VF}(t + 1) < K_{VF}(t) \quad \textbf{(79)}$$

and

$$K_{VF}(t) \subsetneq K_F(t) \subsetneq K_G(t) \quad \textbf{(80)}$$

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

The Decomposition grows monotonically with respect to both I_N and I_G.

Since $\forall$ S(t), $\exists$ a decomposition D(S(t)) of depth n

the Decomposition level requires a number of External Verifiers, such that

$\rho_c \leq \epsilon_c \Rightarrow D(S(t))$ *requires at least* $|V_{ext}^{required}(t)|$

where

- $|V_{ext}^{required}(t)|$, set of Verifiers required at time t
- ρ_c, Composite Density
- ϵ_c, the fragmentation coefficient

therefore,

$$|V_{ext}^{required}(t)| \geq |Fragments(t)| \quad \textbf{(81)}$$

that is, the System requires a sufficient number of Verifiers to sustain the imposed level of Decomposition D.

This is because Decomposition is an Epistemic Property of the System, not an accidental dependence on the Number of Verifiers available at time t.
In other words, the System does not depend on the number of Verifiers, but manages it and requires it as a Preliminary Condition.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

Let's clarify that within the System, the Verifier is not the User.

9.3 Axiom III

The Validity of the System cannot be reduced to the Validity of the Individual Structured Elements that constitute it.

$$\text{Valid}(S(t)) \iff V_{\text{Global}}(S(t)) = \text{TRUE}$$

$$S(t) = (E(t), R(t), C, V_{\text{Local}}),$$

$$V_{\text{Global}}(S(t)) = f(E(t), R(t), C),$$
with
$$\forall\, s_i \in E(t),\ V_{\text{Local}}(s_i) = \text{TRUE}$$

therefore

$$V_{\text{Global}}(S(t)) \neq \bigwedge_{si \in E(t)} V_{\text{Local}}(s_i(t)). \qquad \textbf{(82)}$$

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

Let's look at it in more detail, because at first glance it might seem like a Law deduced from the previous Axioms, but in reality it possesses an independent invariant role within the System.

Given that

$\forall s_i \in E(t), V_{Local}(s_i) = TRUE$

it may still occur that

$V_{Global}(S(t)) = FALSE$

This happens because the Validity of the State depends not only on the Verification itself, but also on R and C.

Therefore,
$V_{Global}(S(t)) = TRUE$

if and only if

$[\forall s_i \in E(t), V_{Local}(s_i) = TRUE] \land Consistent(R(t)) \land [\forall c_j \in C,$

$c_j(S(t)) = TRUE]$ **(83)**

This Axiom is extremely important for ensuring Global Structural Coherence, because without this Invariant Specification, the System would become structurally inconsistent because successive Local Verification Aggregation Conditions would be sufficient to obtain the Valid State, without imposing a Global Coherence Check at higher levels.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

In other words, the System is structural and does not exhibit aggregative properties for different Verification Levels.
Therefore, Validity is an emergent Property, not a simple sum.
R and C are essential for the State Definition.
In summary:

Global Validity cannot be reduced to Local Validity. **(84)**

Verification of the System's State depends not only on the Individual Structured Elements, but also on the Consistency of Relations and compliance with Invariant Constraints.
In conclusion, Axiom III establishes that in a CNVS System, Global Validity must also depend on Relations and Invariants, not just on the Structured Elements.

Therefore:

$$V_{Global}(S(t)) \neq \bigwedge_i V_{Local}(si(t)), \quad \textbf{(85)}$$

this is because the Verification of a State, on which its Validity depends, is not only the Derivation of the Aggregation of the Verifications of the Individual Elements that constitute it.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

10. Information Restriction Principle

Introduction

Since the decomposition depth does not respond directly to the information, but responds to the resulting information density, the relationship is:

$\rho_C(t) \leq \epsilon_C$,

therefore, as Recursive Depth, Information Dispersion, and Adaptive Fragmentation increase, the relative share of Locally accessible Information progressively approaches zero with respect to the Node and the Global System.
In Chapter 7, the problem of the difficulty of managing Fragmentation if controlled directly through Composite Density was raised.
The proposed solution is to control the Decomposition Depth under the reciprocal of ρC, assuming Uniform Nodes and Fragments, such that:

$1/ (\rho_C(t) * N(t)^{\beta_\rho})$

Furthermore, considering that the growth of I_G must also be able to take control of it, we have:

$\log_2 (1 + (I_G))$.

The two relations can be operationally extended by introducing the Fragmentation Intensity Coefficient λ and the reference Information Unit I_0, resulting in the following Fragmentation Function defined:

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

Let $\xi \in \Xi$

denote the random outcome governing the stochastic decomposition and allocation process,

(86)

$$\text{Fragments}(t) = |\mathcal{T}(t, \xi)| = \lceil \lambda \cdot [1/ (\rho_C(t) \cdot (N(t)^{\beta_N}))] \cdot \log_2(1 + (I_G(t) + I_N(t))/ I_0)] \rceil$$

where

$\lambda \in \mathbb{R}_{>0}$ and $I_0 \in \mathbb{R}_{>0}$, $N \in \mathbb{N}$

such that
λ, constitutes a Fragmentation Intensity Coefficient;
ρ_C, the Composite Density understood as a weighted geometric mean;
N, represents the total number of active Nodes already present in the System, i.e., the cardinality of E(t),
with $E(t) = E_{int}(t) \cup E_{term}(t)$
$E_{int}(t)$, the internal and structural Nodes up to the terminal level;
$E_{term}(t)$, the terminal Fragments only,
$N(t) = |E_{int}(t)|$, i.e., at time t
β_N, a Protocol Parameter;
I_0, a reference information threshold, indicating the interval in bits for each increase of +1 fragment, i.e., a positive reference unit of information that determines the scale at which increases in Global information trigger further fragmentation. Its value is defined by the implementation;
$I_G(t)$, $I_N(t)$ already defined previously,
all at time t.

With the formula in (86) the imposition of growth of the number of Fragments

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

does not depend only on the number of initial Nodes, therefore on their variation, but also on the growth of the Global Information I_G, and evidently on the Nodal Information I_N independent of how it is distributed in the successive nodes.

The System or Implementation may provide for a variation of the various coefficients, including λ, to manage the efficiency of fragmentation.
The tables at the end of this document show concrete examples.

Definition
Let

$K_{VF}(v_j)$

the Knowledge accessible to the External Verifier vj at time t
and lets

$I_F(t)$, $I_N(t)$, $I_G(t)$

Fragmental Information, Nodal Information, and Global Information, respectively,

Recalling
$\mathtt{I}_G(t) \cong (E(t), R(t), C, V_{Local})$,
$\mathtt{I}_G(t) \cong \mathtt{I}(S(t))$,
$K_G(t) \preceq I_G(t)$,

and the System follows the Imposition to the Increasing Depth of Decomposition n, the formulation

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

Fragments (t) $= F_x(t) = \lceil \lambda \cdot [1/ (\epsilon_C \cdot (N(t)^{\beta_{-N}}))] \cdot \log_2(1 + (I_G(t) + I_N(t))/ I_0)] \rceil$

where

$F_x(t) \in \mathbb{N} \geq 1$

with

$\lambda\uparrow \Rightarrow F_x\uparrow$
$I_G\uparrow, I_N\uparrow \Rightarrow F_x\uparrow$
$I_0\downarrow \Rightarrow F_x\uparrow$
$N(t)\uparrow \Rightarrow F_x\downarrow$

and therefore $K_{VF}(t)\downarrow$

It follows that:
The System, through its invariants, imposes a Distribution of Information such that the Local Knowledge K_{VF} accessible to each External Verifier vj is structurally insufficient for the reconstruction of the Nodal Information and the Global Information.
The Knowledge Restriction becomes the Structural Condition of the entire System.
Formally:

$K_{VF}(t) \not\Rightarrow I_N(t)$ **(87)**

and

$K_{VF}(t) \not\Rightarrow I_G(t)$ **(88)**

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

Therefore, Local Knowledge K_F does not contain, allow, or imply the information reconstruction of the Node or the Global System.
In Detail, given the Adaptive Decomposition

$D^{(n)}(S(t)) = \{s_i^{(n)}(t)\}$

and the progressive reduction of densities

ρ_N ,ρ_G, ρ_C,

with

$\rho_N(t) = I_F(t)/ I_N(t)$

$\rho_G(t) = I_F(t)/ I_G(t)$

$\rho_C=(I_F/I_N)^{\alpha} * (I_F/I_G)^{\beta_\rho}$

$\alpha, \beta_\rho \in [0, 1]$

with $\alpha + \beta_\rho = 1$, established that $t2 > t1$,

and after the Application of the adaptive fragmentation function

$\forall\ t2 > t1,\ I_N(t2) > I_N(t1) \Rightarrow \rho_N(t2) < \rho_N(t1)$

and

$\forall\ t2 > t1,\ I_G(t2) > I_G(t1) \Rightarrow \rho_G(t2) < \rho_G(t1)$

from the application of the imposed Axioms it emerges

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

$I_F(t) = I_h^{(n)}(t) / F_x(t)$
(in expectation over the stochastic fragmentation process)

where

$I_h^{(n)}(t)$

constituting the decomposition-level information $I_h(n)$ of all the Fragments that make up the same level at time t, but not the level preceding it, and $K_h^{(n)}$ its decomposition-level knowledge, we have that

$I_h^{(n)}(t) = I(s_{ij}^{(n)}(t))$ e $K_h^{(n)}(t) \preceq I_h^{(n)}(t)$,

with

$I_F(t) = \bar{I}_F(t)$,

where $\bar{I}_F$ denotes the expected average fragmental information size,
which constitutes the average size of fragmented information, since we have assumed that they are not uniform among themselves,
the relationship therefore becomes

$(\rho_C, \lambda, I_0) \Rightarrow F_x(t) \Rightarrow I_F(t) \Rightarrow K_{VF}(t)$

and since

$K_{VF}(t) \lneqq I_F(t)$,

it turns out that:

$\lim_{Fx(t)\to\infty} K_{VF}(t)=0$ **(89)**

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

and consequently,

$$\lim_{Fx(t)\to\infty} \rho_N(t)=0 \quad \textbf{(90)}$$

Information limitation ultimately constitutes a form of structural insufficiency of Local Knowledge with respect to the Global complexity of the System, such that the complete informational reconstruction of the Fragment, of all the levels preceding it, and of the Global one, remains an emergent property exclusive to the System.
The Principle of Information Restriction therefore becomes an Epistemic Property deriving from the Application of the Axioms.
Formally, since

$$D^{(n)}(S(t)) = f(F_x(t)) = f(\rho_C(t), \lambda, I_0, N(t))$$

we conclude that

$$K_{VF}(t + 1) < K_{VF}(t) \quad \textbf{(91)}$$

The lower the allowable Composite Density, the greater the depth, fragmentation, and information dispersion must be: that is, the System increases the Decomposition to reduce the accessible Information Density, based on the set Protocol Parameters, over which it has full control.
This implies that the reduction of Knowledge can be "regulated" within the System as a Condition whose Threshold level depends on the Coefficient ϵC and how this affects the imposed Value of ρC and overall on all the Protocol Parameters, including α, β_ρ, and I_0.

Let $H(I_N)$
be the Information-theoretic Entropy of the nodal information, measured in bits.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

The Knowledge Restriction Principle requires that the Conditional Entropy of the nodal information, given the fragmented knowledge K_{VF} available to a single verifier, remain close to the original Entropy:

$$H(I_N | K_{VF}) \approx H(I_N) \quad \textbf{(92)}$$

Ideally, the information obtained about the complete Nodal State is limited, and the Verifier's knowledge of the Fragment should not significantly affect the Uncertainty across the entire Node, because the verifier should learn virtually nothing about its overall content through fine-grained fragmentation.
This can be expressed using Mutual Information:

$$I(K_{VF}; I_N) \approx 0,$$

that is, each Verifier knows only a limited and increasingly smaller portion of the overall Content, so we will have

$$H(I_N | K_{VF}) = H(I_N) - I(I_N; K_{VF})$$

$H(I_N)$, total Entropy of the node;

$H(I_N | K_{VF})$, residual Entropy after observing the fragment.

This means that access to a single Local fragment does not significantly reduce the uncertainty about the complete nodal state, because the System is designed to minimize the mutual information between the verifier's Local knowledge and the complete nodal state.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

11. Operational Flow

In this chapter, we will describe the System's operational flow. After illustrating the procedure used by the System to fragment the I_G Information and obtain the final decomposition depth, we will describe in detail how the semantic information is distributed from I_G to the final levels: that is, we will see how the system constructs the acquired Information Graph for the Object.

In the first phase, the User identifies the Object O_x.
Recalling:

$$O_x \rightarrow I(O_x) \rightarrow \tilde{I}(O_x) \rightarrow E,$$

$$O_x \rightarrow I(O_x) \rightarrow \tilde{I}(O_x) \rightarrow E \rightarrow s \rightarrow \mathtt{I}(s),$$

$$O_x \rightarrow I(O_x) \rightarrow \tilde{I}(O_x) \rightarrow E(t) \rightarrow \bar{s}_i \rightarrow V_{Local} \rightarrow P_i \rightarrow V_{global}(S),$$

$$O_x \rightarrow I(O_x) \rightarrow \tilde{I}(O_x) \rightarrow E(t) \rightarrow s \rightarrow \mathtt{I}(s) \rightarrow \forall\, \bar{s}_i \rightarrow V_{Local} \rightarrow P_i \rightarrow V_{global}(S).$$

The Object is introduced into the System in the form of a Verification instance translated according to a binary representation, to define the components of the Structural Universe $\boldsymbol{S}$ from which the System extracts $\bar{S}$.

This phase constitutes a delicate moment because the constitution of the Object Structure in binary representation, starting from the User's request, will define the characteristics of all the specificities of the Object on which the subsequent comparison will depend.

In this sense, an inadequate representation of the Object's specific characteristics due to a lack of detail could lead to the Object's exclusion from the System's admitted Existence scope following the application of the verification process.

This eventuality is entirely legitimate and allows us to conclude that the Object is admitted into the System if and only if it exists in all the specific characteristics described in detail by the User, according to their needs.
The System, in fact, responds to the User's needs and adapts to their request.
To be clear, a poorly detailed User request would certainly lead to a conclusion that is inconsistent, but not necessarily weak, as for that User, a "rough" and

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

poorly detailed assessment might suffice.
Let's take an example.

If we introduce into the System an instance to verify the existence of the Object "the gold bar exists in cav. number 17," without specifying any further details, the System could simply verify the existence of the Object "gold bar" in the environment "cav. number 17," without further specifications.
In this sense, the User's request could be granted a favorable response, without necessarily implying System Failure, since the request is very generic and the System simply needs to verify the Object in the form presented by the User: "a gold bar," without any specification of weight, purity, shape, size, state of preservation, and so on.
And this might already be sufficient for the User.

The System would therefore simply validate the Existence of a generic "gold ingot" without further details, and would admit it into its Field of Existence if the verification procedures returned a 1.

This means that the more detailed the description of the Object's specific characteristics in the instance requested by the User, the more thorough the verification structured by the System will be, which will be able to impose more rigorous checks on all the details relating to the specific characteristics described in the request.

Returning to the example above, if we introduce the following verification instance into the System, "the gold ingot Object, weighing 122.3 grams, with a purity of 99.5%, density of 19.3 g/cm^3, cubic in shape, with sides measuring 1.9 cm in length, exists in vault number 17. It is stored isolated, in a controlled environment," it is clear that the System's verification will be more detailed.
In this sense, verifying the Existence of a generic "gold ingot" will not be sufficient; all the specific characteristics described in detail will be sought.

The System, for example, could reject the Status of Existence because the verification procedures would reveal: "the gold ingot weighing 122.3 grams, with a purity of 91.2%, a density of 18.2 g/cm^3, cubic in shape, with sides measuring 2.3 cm in length, exists in vault number 17. It is stored in a non-isolated, poorly controlled environment".

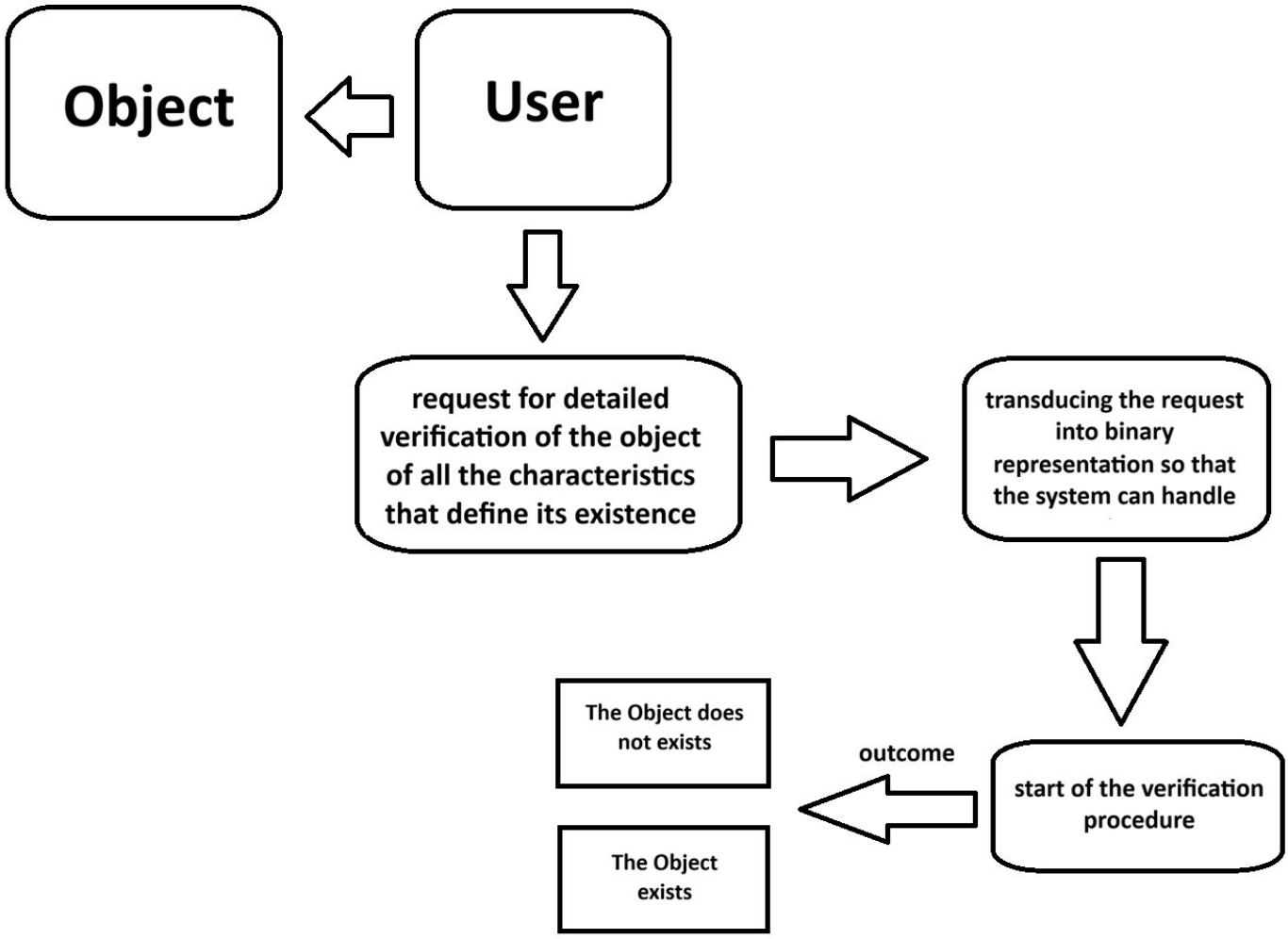
© 2026 M. Comitato — The Theory of Closed Native Verification Systems

An Object Exists in the System if and only if the specific characteristics described by the User are fully matched.

Translating the description provided by the User into binary representation constitutes another delicate phase.

For some Objects, it would theoretically be possible to upload documents to the IT devices that use the CNVS System, such as a court ruling or a user manual for a device whose functionality and technical specifications you wish to verify; in other cases, this phase could be more complex.
However, as we have observed, the complexity of the introduction phase does not invalidate the conclusions reached by the System.

Schematically, we have:

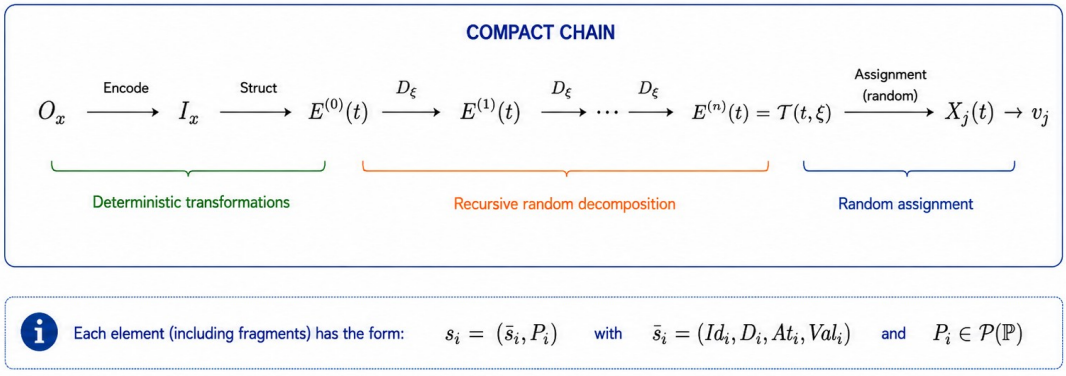


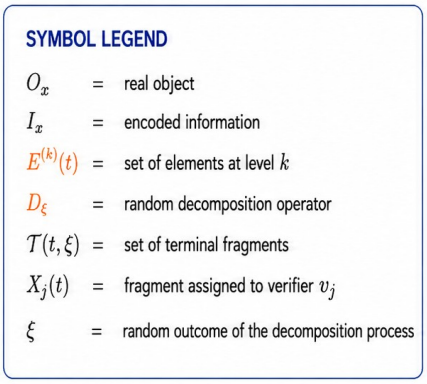
© 2026 M. Comitato — The Theory of Closed Native Verification Systems

The verification procedure for the random assignment of Global Information follows a logical operational chain using a randomness function in several steps.
The randomness function enters the System multiple times.
It initially enters during the Fragmentation phase of E(t), recalling

E(t) ⊆ $\mathcal{S}$ per $\mathcal{S}$:= Id × D × At × Obs × P(ℙ)

so schematically we will have the following general operational flow





At this point, it remains to be clarified how the System imposes the division of the granulation.

In Chapter 5, we saw the recursive property of fragmentation, and in Chapter 10, we introduced formula (86), which defines the number of terminal Fragments that the System imposes to divide the information to be randomly assigned to the External Verification Agents.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

However, we have not yet seen how the fragmentation is divided.
In CNVS Systems, the solution is to impose the decomposition starting from the Origin, so as to create multiple initial Level 0 Nodes, thus preventing all the information from being traced back to a single Node.

Subsequently, further levels of subdivision are imposed, with a minimum of at least 4 levels.
The solution must take into account the need for the terminal Fragments to retain a logical sense, because a recursive subdivision that destroys the semantic content would become useless.
For this reason, the System starts from the last level, the terminal level n or Fragments(t).
From that level, it reconstructs the entire tree in an ascending direction, progressively assembling the terminal Fragments.

It doesn't matter if the logical sense of the contents of D_i is lost during the ascending reassembly; what matters is that the relational dependencies required for reconstruction are preserved within the structured representation I, through $I_{metadata}$ and $R(t)$.

In this way, the contents of D_i could only make logical sense at the end of the fragmentation, when D_i is divided "into the correct binary sequence": thus making the Information more cryptic during the intermediate steps.
Formally, we will have

$$\mathcal{T}(t, \xi) \rightarrow E^{(n-1)}(t, \xi) \rightarrow \cdots \rightarrow E^{(0)}(t).$$

$$\Sigma^{n}_{k=1} \; f_k = \text{Fragments}(t)$$

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

e:

$f_n > f_k$ for most $k < n$

e with $n \geq 4$.

therefore $|\mathcal{T}(t, \xi)| = \text{Fragments}(t)$ **(93)**

This ensures that terminal Fragments are more numerous than intermediate levels.

To manage the distribution flow, the proposed solution is to introduce a new parameter: θ.

This parameter guides the distribution of Fragments(t) in an ascending direction.
The ideal values it can assume range from 0.20 to 0.25 for the central levels up to the first Node and range from 0.25 to 0.30 at the terminal level n, thus ensuring the maximum number of Fragments at the last level, which is the useful one.

Formally, we will have

$\text{Fragments}(t) := |\mathcal{T}(t, \xi)|$

where $\mathcal{T}(t, \xi)$ is the set of terminal Fragments generated by the random decomposition process.

Let

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

$$\theta(\xi) = (\theta_1(\xi), \theta_2(\xi), \ldots, \theta_n(\xi))$$

with

$$\theta_k \in (0, 1),$$

$$\sum^{n}_{k=1} \theta_k = 1.$$

The System generates the random sequence of allocations:

$$(f_1, f_2, \ldots, f_n)$$

such that

$$\sum^{n}_{k=1} f_k = \text{Fragments}(t)$$

with

$$f_k \approx \theta_k(\xi)\ \text{Fragments}(t)$$

where

$$\theta_k \in [\theta_{min}, \theta_{max}]$$

$$\theta_{min}, \theta_{max} \in (0, 1)$$

with

$$\theta_{min} < \theta_{max}$$

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

and for the terminal level it could be

$\theta_{min} = 0.25$

$\theta_{max} = 0.30$

therefore

$\theta_k \in [0.20, 0.25]$ for $1 \leq k < n$

$\theta_n \in [0.25, 0.30]$.

Combining the two formulas,

Let $Layer_\xi : \mathbb{N} \to \mathbb{N}^n$

$Layer_\xi(Fragments(t)) = (L_1(t), L_2(t), \ldots, L_n(t))$,

where $Layer_\xi$ depends on the random outcome ξ through the flow vector $\theta(\xi)$

therefore

$L_k(t) = \lfloor \theta_k(\xi) \cdot Fragments(t) \rfloor$ for $k < n$

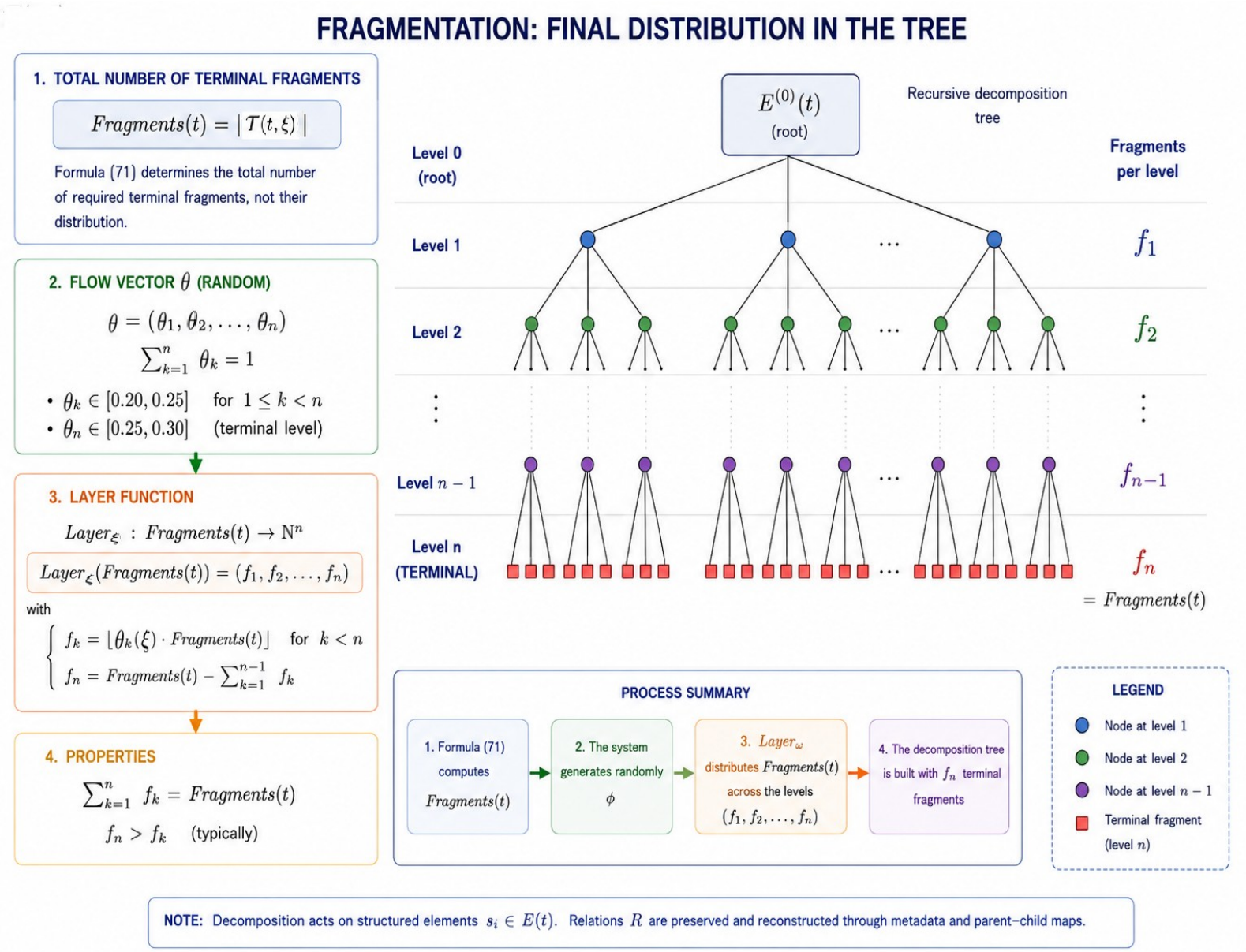
$L_n(t) = Fragments(t) - \sum^{n-1}_{k=1} L_k(t)$

with

$\sum^{n}_{k=1} L_k(t) = Fragments(t)$.

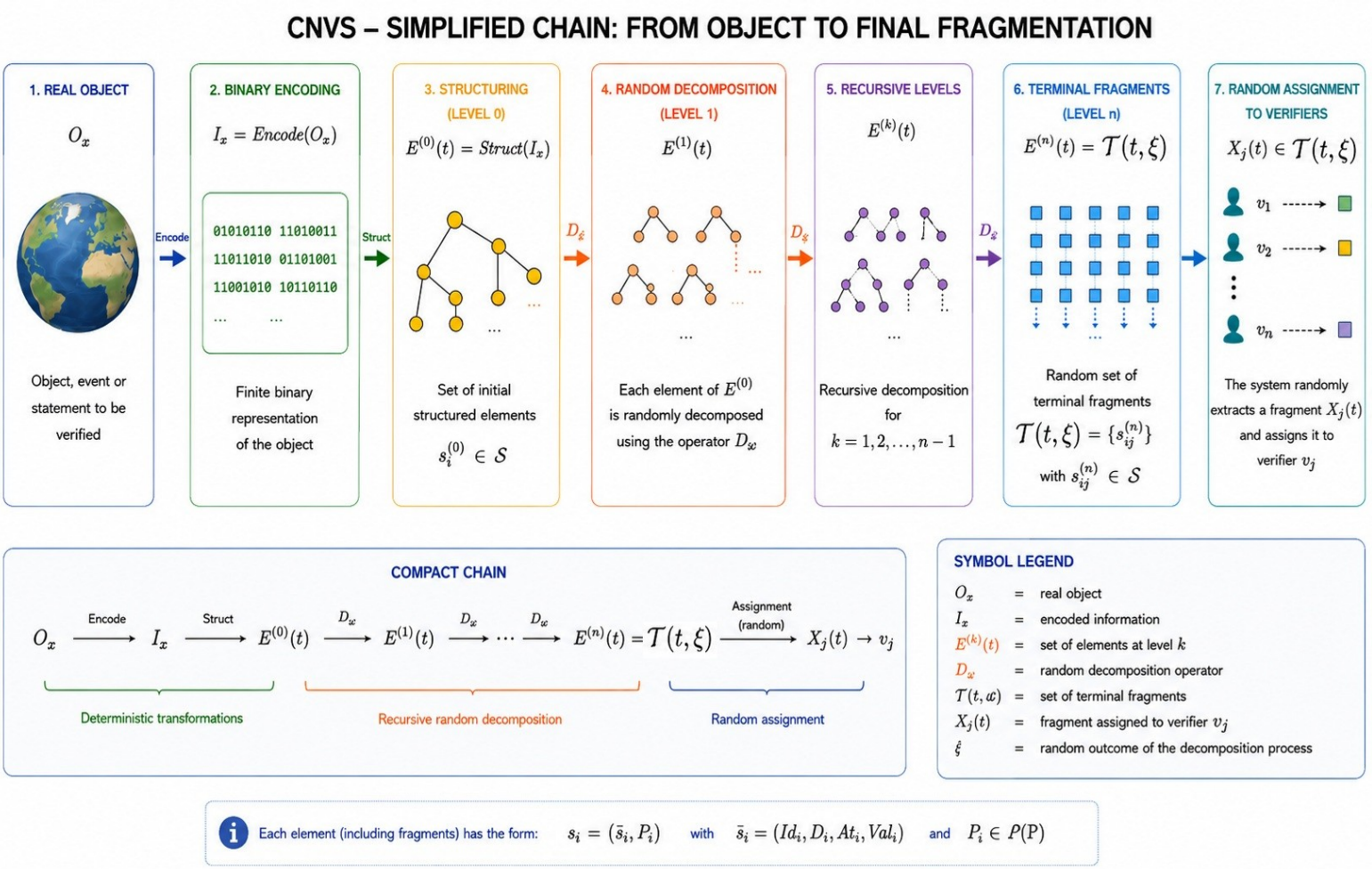
© 2026 M. Comitato — The Theory of Closed Native Verification Systems

Schematically we will have



The System Operational Flow can therefore be schematized as follows

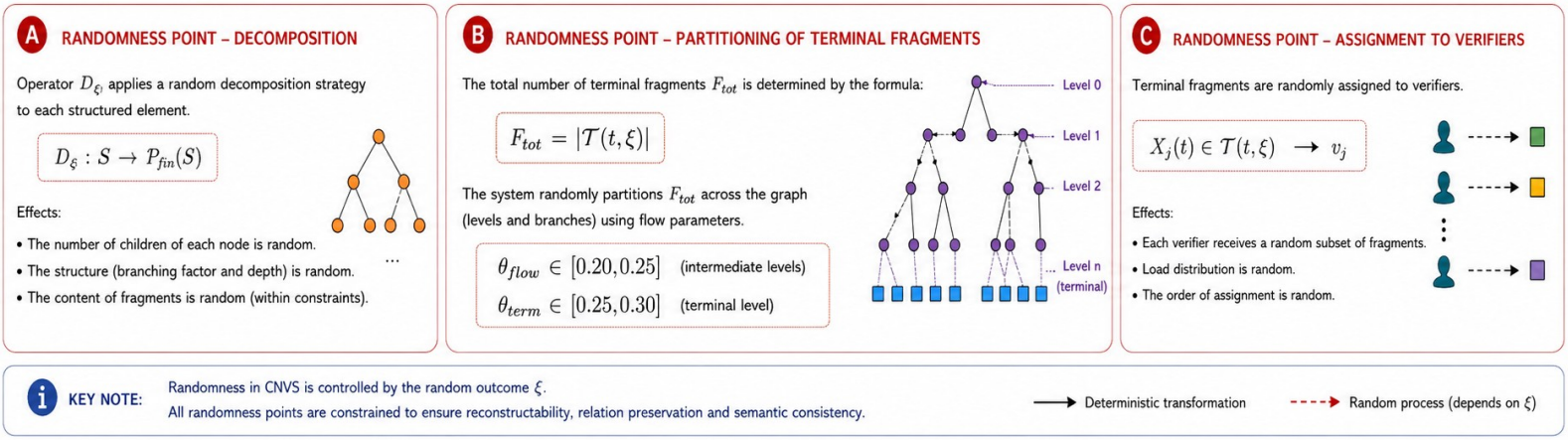
© 2026 M. Comitato — The Theory of Closed Native Verification Systems



taking into account the distribution of the terminal Fragments as set out in the previous diagram.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

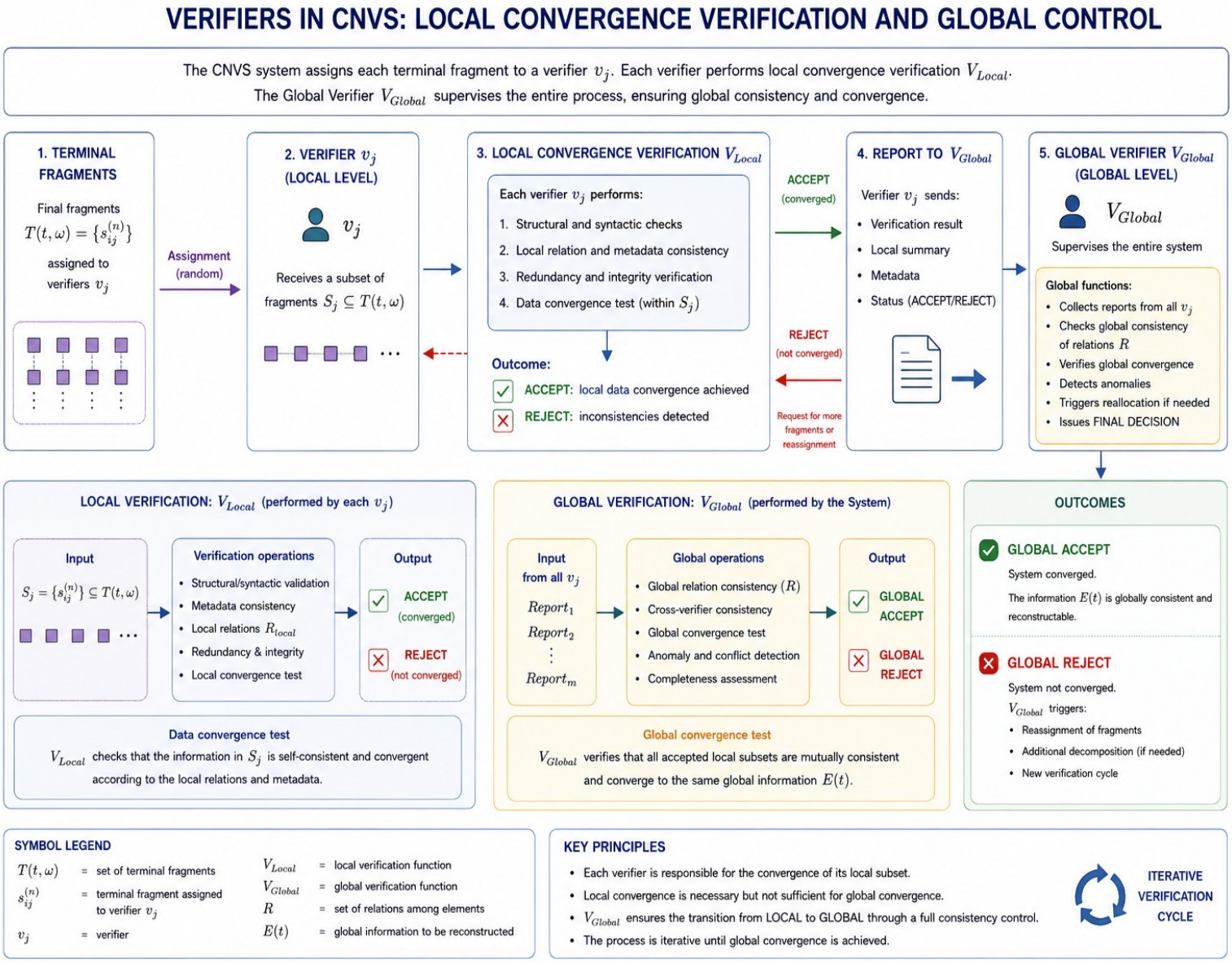
Schematically, the randomness function guides the entire process by intervening in 3 points:



The diagram clearly shows the causal relationships.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

The following diagram shows the role of the External Verification Agent and the V_{Local} and V_{Global} Functions in the operational flow



V_{Global} is a System-level Verification Function, not an External Verification Agent.

The terminal level is level n, whose cardinality is Fragments(t)

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

In Synthesis, It is important to clarify that the CNVS System does not require the continuous semantic reconstruction of the Global Information I_G in order to operate coherently.

The System does not reason directly on the complete semantic payload of the fragmented Information, but rather on its structured relational representation $\mathtt{I}_G$, preserving the logical dependencies among terminal Fragments through the relational structure R(t), the invariant set C, the Local Verification outputs, and the metadata required for structural consistency.

During the reconstruction process, the System does not necessarily reassemble the complete semantic topology of the original Information. Instead, it progressively evaluates relational coherence, invariant satisfaction, and Local admissibility across decomposition levels.

In this sense, reconstruction is not primarily semantic, but structural and relational.

The semantic payload remains progressively hidden during the intermediate aggregation phases, while only the relational dependencies required for coherent verification become operationally exposed.

Consequently, the System minimizes unnecessary semantic exposure during the verification process, because the verification logic operates primarily on structural consistency rather than on direct semantic accessibility of the complete Global Information.

The External Verification Agent does not participate in this reconstructive process. Its role is limited to the production of localized observational inputs

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

Obs_i, which are subsequently processed internally by the System through the Verification Functions.

The CNVS System therefore does not require a continuously reconstructed semantic global topology in order to preserve coherence, validity, and operational consistency.

---

12. Discussion on Shannon Entropy

(nb. in this chapter, H(G) denotes the intrinsic information-theoretic Entropy of the global state, whereas $Hn=\log_2(k!)$ denotes the combinatorial uncertainty associated with fragment assignment)

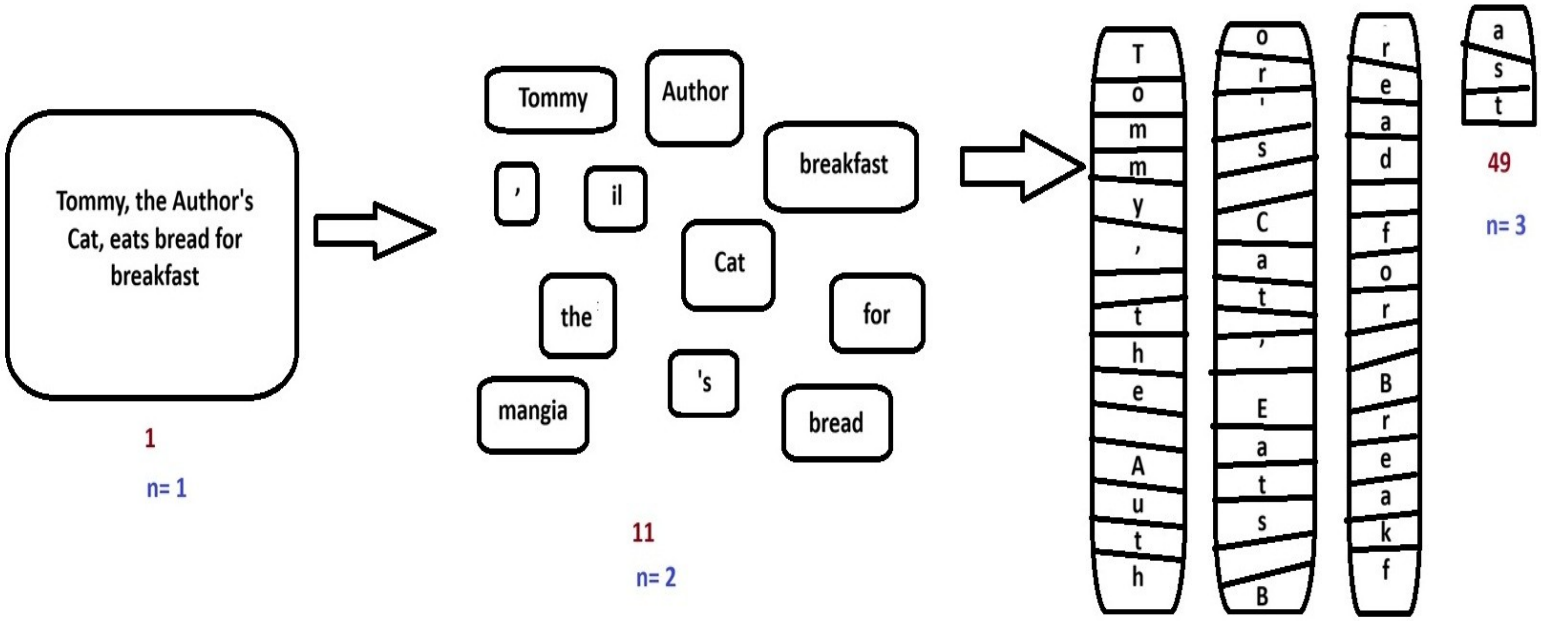
Formalizing the behavior of Shannon Entropy in CNVS Systems is a central point of the theoretical development.

In this discussion, Shannon Entropy is used as an information-theoretic measure of reconstruction uncertainty associated with the combinatorial arrangement of Fragments, rather than as a direct measure of the semantic content of the original Information itself.

The System Architecture is based on the premise that the progressive reduction of the information content of the terminal I_F Fragments, according to the Knowledge Restriction Principle, due to the information granulation of the original I_G Information, makes its reconstruction increasingly difficult for the External Agent.
In an extreme model, by repeating the fragmentation ad infinitum, the information content of IF becomes completely devoid of meaning for the External Agent, losing any accessible logical coherence from the External

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

Agent’s perspective.
An example might clarify the conceptual meaning.
Let's look at the Box below.



It seems clear that the sentence "Tommy, the Author's Cat, eats bread for breakfast" makes logical sense to the External Agent: the Information begins with a series of intuitive logical relationships that anticipate and facilitate its reconstruction.

The situation on the left side of the Box appears quite clear: one enormous fragment at Level 1, conceptually strong.

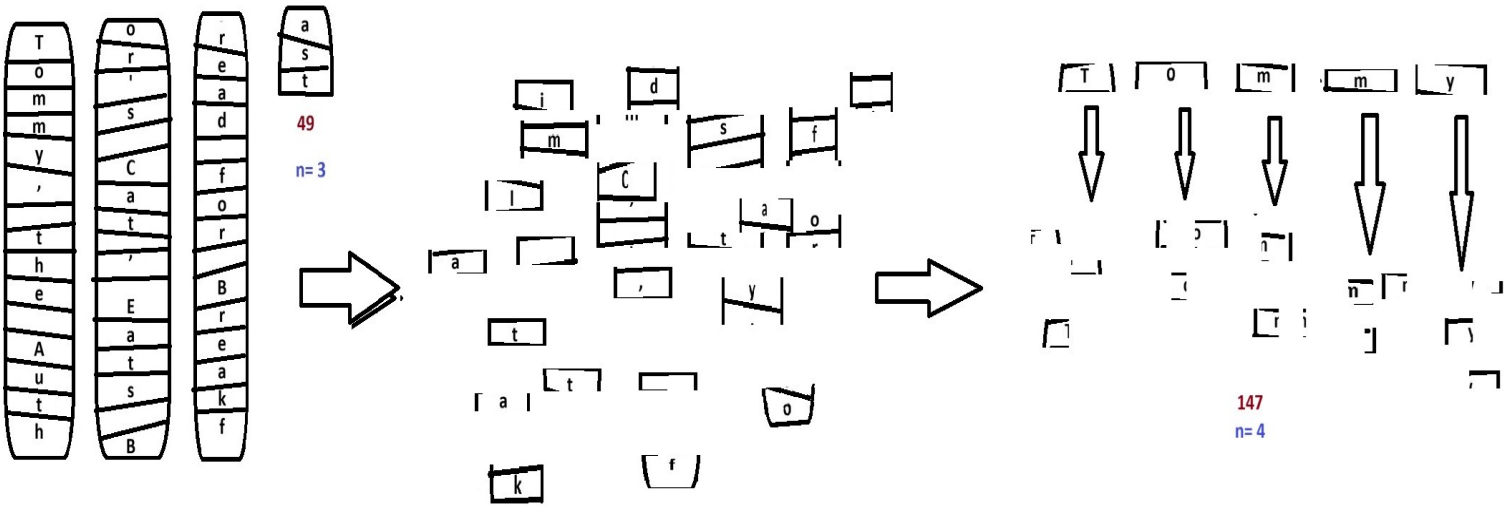
In the central section of the Box, we already see something interesting: following the first informational granulation, the sentence's content is decomposed, producing 9 Fragments at level 2.

On the right side of the Box, we see something even more interesting: following the second informational granulation, the sentence's content

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

becomes highly decomposed, producing 49 Fragments at level 3. At this level, the message is no longer clear, but the semantic content of each individual part still has meaning for the reader, even if it lacks any intuitive logical relationship: they appear to be just scattered letters (T, o, c, l, z, etc.).

Let's continue our reasoning and see what happens at level 4, in the new Box below:



The result is a box containing a set of objects devoid of logical meaning, which have even lost their semantic meaning.

The knowledge of an External Agent with level 4 Fragments will be even more limited, because the absence of any logical and relational constraints between them makes reconstruction extremely difficult.
If we were to continue the granulation process virtually infinitely, we would obtain a fragmentation of the I_G information such that the I_F would become so destructured as to make reconstruction increasingly technically difficult, to the point of becoming virtually impossible in extreme cases.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

This example shows how the CNVS System, using the decomposition of the source I_G information, produces the effect of reducing the information content of each individual fragment obtained I_F, such that the I_F content becomes a function of the number of Fragments.

Approximating, we could say:

$$K_{VF}(v_j, t) \lneqq K_F(t) \leq I_F(t) \approx \bar{I}_F(t) = \mathcal{I}(s_{ij}^{(n)}) \approx I_G / n \quad \textbf{(94)}$$

such that

$I_F \to I_{Fmin}$ as $n \to \infty$, and $K_{VF} \lneqq I_F$

(n.b. the value I_{Fmin} is defined as I_{min} in other parts of the this document and in other documents linked to this one).

More clearly, the information content of I_G can be interpreted in terms of progressive decomposition.

In an extreme CNVS System, beyond a certain information granularity threshold (Semantic Limit, I_{Fmin} or I_{min}), the fragment loses its semantic, logical, and relational meaning, and its exact reconstruction becomes possible only in probabilistic terms.
In theoretical CNVS models, fragmentation is treated as an ideal asymptotic process, in which the number of fragments can grow indefinitely as long as each verifier retains the minimum information necessary to perform the assigned task.

In the application, it is not possible to reduce the K_{VF} knowledge beyond the limit of the External Verification Agent's understanding of the semantic meaning of the content related to the System's request, because this would

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

make it impossible to perform the task assigned to the Verifier.

This limit is set in this document as I_{min} (= I_{Fmin}).
In applied CNVS Systems, it is sufficient for the granulation to go beyond the limit of Global knowledge of the I_G and the Relations between I_{Fs}.
Once that limit is reached, in fact, the External Verification Agent will have no ability to contextualize the data it holds, and it will be even more difficult for it to understand the semantic Relations compared to I_F Fragments of the same level, or worse, of higher levels, because in addition to being encrypted, the Fragments are completely disconnected from each other, are non-uniform, and are randomly distributed.
Virtually any Fragment could contain any part of the I_G, and any value of D_i with respect to At_i.

The following discussion constitutes an approximation model, also with respect to the actual value of K_{VF}.
As clearly emerged in the previous chapters, the External Verifier does not possess the knowledge (K_F) of the fragmented information (I_F), but only the minimal information necessarily entrusted to him by the System to perform the task of performing measurements and checks on the Object under examination (K_{VF}).
The External Verifier does not know the D_i data, but can infer it from measurements and direct access to the Object.
However, the data measured by him will not necessarily converge to the D_i value recorded by the User during the implementation of the Verification Request.
And it is precisely the Verifier's blindness to this data that makes the System robust.
To develop the arguments that follow, the model assumes for simplicity that the External Verifier's knowledge is attributable to the D_i data relating to a portion of the Fragmental Information.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

Therefore, the discussion develops the arguments starting from the I_F data and its general information content, with particular reference to D_i, such that

$K_{VF} \approx I_F^{(Di)}$,

with $I_F^{(Di)} \subsetneq I_F$,

and recalling $I_F \neq \mathbb{I}_F$.

The same approximation is also used in the development of the model relating to the emergent security theorem.

Therefore formally,

$$|K_{VF}(v_j, t)| \geq I_{min} \quad \textbf{(95)}$$

$|I_F(t)| \geq I_{min}$,

with

$|K_{VF}(v_j, t)| \leq |I_F(t)|$.

Assuming that the reconstruction problem consists in identifying the correct ordering of n distinguishable Fragments among all equally likely permutations, and admitting the simplified model in which all the Fragments are standardized to the same information size and are randomly distributed as desired by the System, also admitting that all are encrypted and completely anonymous, the probability of a successful random reconstruction is:

$p = 1/ n!$

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

If I_G Decomposition is applied in cycles such that its content is progressively fragmented twice at each cycle, the situation will be as follows:
after the first granulation cycle, we will have
I_G divided into two Fragments, each containing 1/2 of the original information.

Formally,

assuming

$\bar{I}_F(t) \approx I_G(t)/ n$, under uniform average fragmentation assumptions

therefore

$I_F(t) \approx I_G(t)/ 2$

with
$p = 1/ (2*1) = 0.5$

If we apply 3 cycles, we will have:
$I_{F,3}(t) \approx I_G(t)/ 2^3$ with $p = 1/ (6*5*4*3*2*1) = 0.00138888$

If we repeat the cycle 20 times, we will have:
$I_{F,20}(t) \approx I_G(t)/ 2^{20}$ with $p = 1/ (2^{20})!$,

generically,

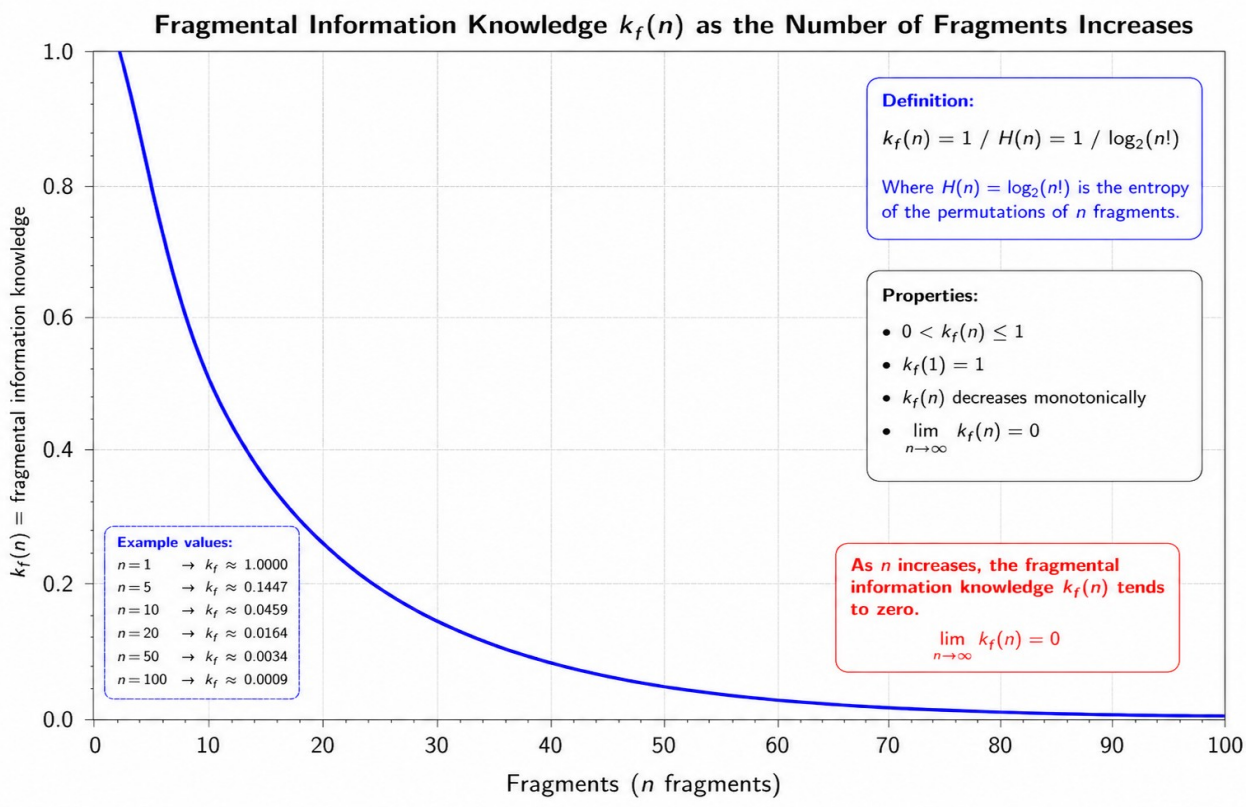
$I_{F,m}(t) \approx I_G(t)/ 2^m$ and $p_m = 1/ (2^m)!$

with $m \in \mathbb{N}$,

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

m is the number of decomposition cycles.

It is clear that the probability of finding the right combination by relying solely on chance becomes so low that the effort is completely useless.
Graphically, we can represent the relationship between the Knowledge of Fragmental Information $K_F(n)$ and the number of Fragments (n), with $n \to \infty$



In this context, Shannon Entropy measures the uncertainty associated with the reconstruction problem, namely the uncertainty of identifying the correct configuration among all equiprobable permutations of the Fragments.

Therefore, in CNVS Systems, increasing Shannon Entropy does not imply an increase in semantic Information, but an increase in the uncertainty and combinatorial complexity associated with the exact reconstruction of the fragmented Information.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

We calculate Shannon Entropy in a model in which we have 20 Fragments arranged in a row and each recognizes a single successful combination among all equally likely combinations.
Let $p_i = 1/ 20!$,

let Shannon Entropy be

$$H(X) = -K \Sigma_{i=1}^{N} p_i \log_2 p_i, \quad \textbf{(96)}$$

N= 20!, K= 1

replacing, $H(X) = - \Sigma_{i=1}^{20!} 1/ 20! \log_2 (1/ 20!)$,

and since the term is the same for all combinations we will have

$H(X) = -20! * 1/ 20! \log_2 (1/ 20!)$,

$H(X) = - \log_2 (1/ 20!)$

$H(X) = \log_2 (20!)$

$H(X) = 61.08$ bit.

By composing the 20 Fragments one at a time we will have

20*19*18*⋯*2*1

the correct probabilities at each step become

1/ 20 * 1/ 19 * 1/ 18 * 1/ 17 *…. * 1/ 2 * 1/ 1.

then,

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

$p = 1/ 20!$

the total information will be

$I = -\log_2(p)$

$I = \log_2(20) + \log_2(19) + \log_2(18) + \cdots + \log_2(2) + \log_2(1)$

$I = \log_2(20!)$

$I \approx 61.08$ bit,

In conclusion,
p(correct combination by chance) = 1/ 20!
20! = 2,432,902,008,176,640,000.
Therefore, by randomly choosing an arrangement of the 20 Fragments, the probability of immediately obtaining the correct one is approximately:
$4.11 * 10^{-19}$
that is, approximately $4.11 * 10^{-17}$% probability of guessing the correct combination on the first try.

Examining the data from the Binary Information relating to the calculated Uncertainty, we will have that to uniquely identify the correct arrangement among all 20 possibilities, approximately 61 bits of combinatorial information are required [1, 0].
61.08 bits correspond to $2^{61.08}$ possibilities.
And in fact, $2^{61.08} \approx 20!$ corresponds exactly to the number of permutations of 20 objects.

Let's try to extend the reasoning by gradually increasing the Decomposition up to the limit $+\infty$ and see how the Entropy behaves.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

The number of possible combinations/permutations is:
$N(n) = n!$
If all are equally probable:
$p_i = 1/ n!$,
substituting into the Shannon Entropy we have

$$H(X) = - \Sigma_{i=1}^{N} p_i \log_2 p_i$$

$$H(X) = - \Sigma_{i=1}^{n!} 1/ n! \, \text{Log}_2 (1/ n!)$$

therefore

$H(X) = - n! \; 1/ n! \, \text{Log}_2 (1/ n!)$, and simplifying

$H(X) = - \text{Log}_2 (1/ n!)$,

$H(X) = \text{Log}_2 (n!)$.

Let's calculate the limit

$$\lim H(X)_{n\to\infty} = \lim_{n\to\infty} \text{Log}_2 (n!)$$

with $n \to \infty$, then $\log_2(n!) \to \infty$

$$\lim H(X)_{n\to\infty} = +\infty \qquad \textbf{(97)}$$

Since n! becomes a very large number, we perform the calculations using Stirling's approximate formula, which was created specifically to handle these special cases.
The formula is $n! \approx \sqrt{(2\pi n)} * (n/ e)^n$

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

so we will have

$H(n) \approx \log_2 [\sqrt{(2\pi n)} * (n/e)^n]$

$H(n) \approx n \log_2 n - n \log_2 e + \frac{1}{2} \log_2 (2\pi n)$ **(98)**

The dominant term is:

$n \log_2 n$

$H(n) \rightarrow +\infty$

the probability instead will be

$p = 1/n!$.

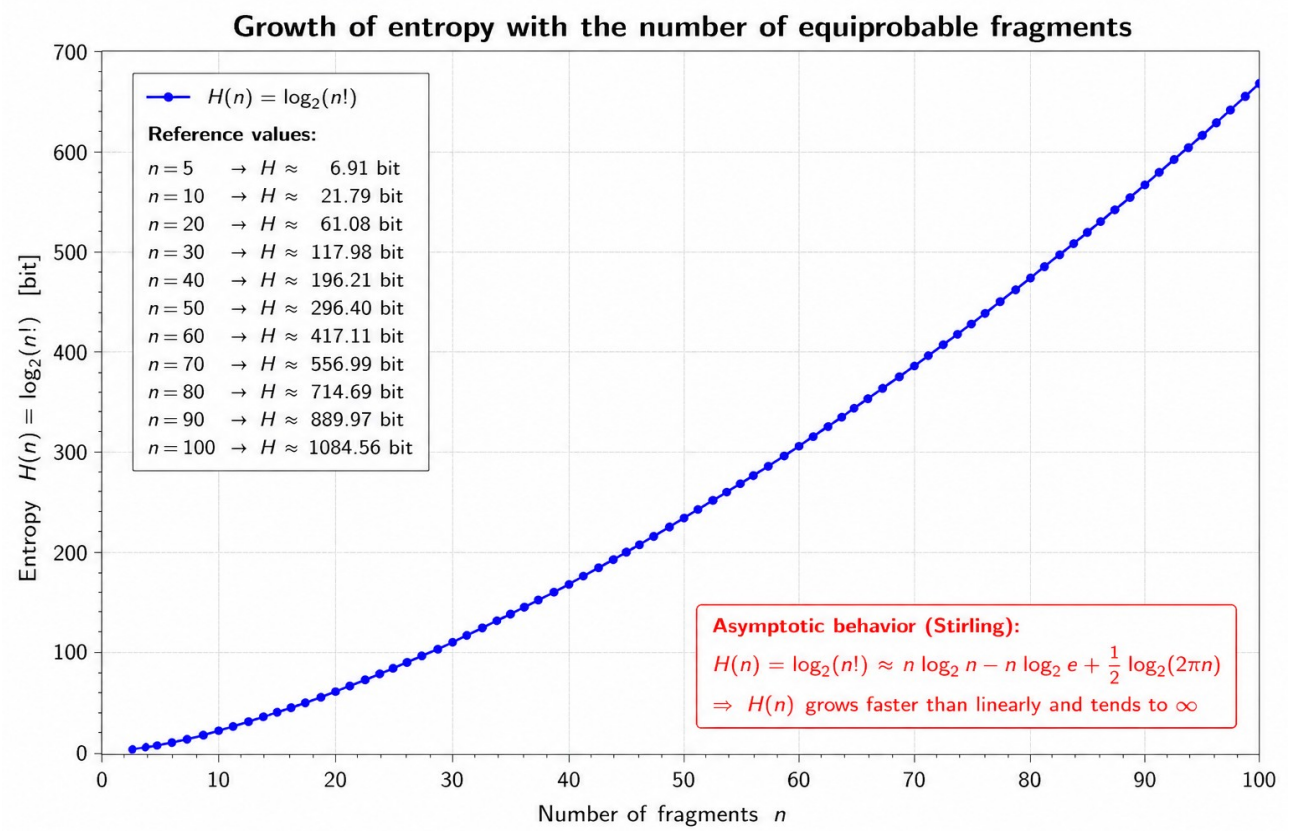
Therefore

$\lim_{n\to\infty} p_n = \lim_{n\to\infty} 1/n! = 0$

Therefore, as the number of Fragments grows without limit, the probability of randomly guessing the correct combination approaches zero, while Entropy approaches infinity, therefore the uncertainty associated with exact reconstruction also diverges.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

Graphically,



Therefore, in an ideal CNVS model, increasing fragmentation causes the fragmental information $I_F(n)$ to decrease, while the Entropy associated with the reconstruction problem increases. However, the System cannot pursue infinite fragmentation as an absolute objective, because below a certain operational threshold the fragment would no longer preserve the minimum information required for verification. The implementation must therefore identify an optimal threshold n^*, as a function of the source information I_G, at which reconstruction uncertainty is maximized while fragmental information remains operationally usable.

Concluding,

$I_F(n) \geq I_{Fmin}$

with I_{Fmin} denotes the minimum amount of fragmental information required for the fragment to remain operationally usable for verification

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

and with the threshold value thus formalized:

$$n^* = \max\{n \mid I_F(n) \geq I_{Fmin}\} . \quad \textbf{(99)}$$

---

13. State Rejection

The System is designed to resist External Hacking and Authoritarian Manipulation attempts, but there may be Situations, discussed in a separate document, that cause integrity failure.
These Situations can occur when certain Conditions that the System deems essential for its very survival cease to exist.

These Conditions at time t include:

- Density can no longer be maintained;
- Verifiers failure;
- Depth does not increase sufficiently;
- ρ_C exceeds the threshold ϵ_C;
- ρ_N exceeds the threshold ϵ_N;
- ρ_G exceeds the threshold ϵ_G.

The System also is rejected when the Internal Verification process is determined by the External Verification process in such a way that it becomes:

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

- predictable;
- manipulable;
- subject to collusion.

In all these cases, we will have:

the verification process ceases to be operationally deterministic due to corruption of the verification conditions

Formally:

$\text{DetIntegrity}(S, t) = 0 \Rightarrow \text{Reject}(S)$

In these cases, V_{Global} does not cease to be formally deterministic.
Rather, the deterministic integrity assumptions required for its valid application cease to hold.

and therefore

$S(t) \notin S_{valid}$

$\text{DetIntegrity}(S, t) = 1$

if and only if the verification process maintains:
• unpredictability of the assignment;
• non-manipulability;
• non-dependence on the external verification process;
• absence of structurally dominant collusion.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

In all cases DetIntegrity(S, t) = 0, the System loses the right to be considered a CNVS and becomes a Traditional System, because the Integrity of the Verification Function is not a functionality, but a definitional requirement, such that the System is not only violated, but becomes a different type of System. The Integrity of Verification is not a functional requirement, but an necessary precondition for membership.

It is worth noting that the State is rejected whenever a global invariant is violated. As already established in the previous sections, even when the fraction of compromised Verifiers exceeds the level that would ordinarily be considered critical, unauthorized reconstruction may still be detected and rejected if the colluding agents fail to acquire sufficient useful information to exceed the reconstruction threshold τ. In such cases, the System does not authorize the transition from the Candidate State to the Existing State.

This occurs due to the Colluding Verification Agent's total global semantic blindness with respect to the remaining portion it does not control. This portion, being unpredictable due to the non-uniform distribution of D_i in the terminal Fragments, is limiting, depending on its utility or weight $\omega(D_i)$ with respect to the global information I_G (where $\omega(D_i)$ denotes the relative informational weight of D_i with respect to I_G, as discussed in the following chapter).
The attacker will be unlikely to know the information content of the nodes he doesn't control, and given the completely random distribution of the fragments, he won't be able to easily deduce it.

In this context, the Global Verification Function could easily highlight a discrepancy in the value expected by I_G or I_N during the intermediate reconstruction phases: the recomposition in the higher-level Nodes of the

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

Fragments controlled by the Attacker with the Fragments that he does not control and of which he has no knowledge, will lead to impossible results, violating the invariants of the System.

The CNVS System completely rejects the traditional consensus paradigm, where the majority wins, therefore it is sufficient for even an infinitesimal fraction of the Information to be inconsistent for the V_{Global} function to fail, activating the rejected State.

---

14. Limitations of the Theoretical Model

External Verifiers constitute a Membership Class, which is not necessarily "human." Technically, External Verification can be entrusted to the use of hardware devices supervised by human professionals, or it can be completely automated.
This Document does not go into detail about the methods used to perform External Verification, but it assumes that it is performed with Objective Judgment, structuring a System that reduces the possibility of involuntary and intentional errors through Fragmentation Control and the Reconstruction of the Global Asset with its progressive implementation.
This is even more true considering that every External Verification is permanently archived within the System, which preserves the Historical Memory of each Individual Verification for each Individual Structured Element. The ability to track the Unique Identifier of each External Verification, as well as all individual steps within the System, makes a Violation theoretically impossible without traceability.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

External Verifiers can therefore be:

- Human auditors
- Automatic validators
- Sensors
- Vaults
- Software agents
- Distributed modules.

Specifically, the sustainability of decomposition growth could become a long-term problem.
If the value of Fragments(t) were to grow too rapidly, then the computational cost would become difficult to offset.
At the same time, there could be issues with synchronization, latency, and relationship management.

Systemic Complexity can be well summarized by reading the following comparison table:

More fragmentation	**Less knowledge**
less knowledge	More emergent security
More fragmentation	More Systemic complexity

The application model forces a natural compromise to be made with respect to the conditions assumed by the theoretical model.
It should also be noted that not all Objects are equally suited to the CNVS

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

Class.

An Object whose information content can be decomposed deeply before losing the minimum information I_{Fmin} will certainly be a candidate with a profile that better meets the requirements allowed by the CNVS than an Object that is poorly suited to a deep recursive decomposition.
Obviously, its semantic nature will be a key element.

To formulate this notion, we can say that
the Granularity of an Object corresponds to its ability to lend itself to recursive decomposition without losing its semantic characteristics, even though it is distributed into disconnected fragments with no logical connection.

Formally,

The application model forces a natural compromise with respect to the idealized conditions assumed by the theoretical framework.

It should also be noted that not all Objects are equally suitable for the CNVS class.

An Object whose informational content can be deeply decomposed before reaching the minimum semantic threshold IF_min is naturally a better candidate for CNVS architectures than an Object that poorly tolerates recursive fragmentation.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

Its semantic structure therefore becomes a fundamental factor.

To formalize this notion, define the Granularity Ratio of an Object as its ability to support recursive decomposition while preserving the minimum semantic conditions required for local verification, despite the distribution of information into disconnected fragments with no explicit logical linkage.

Formally,

sia Γ = Granularity, $\Gamma = m_{max} / m_{min}$, **(100)**

where

- m_{max} represents the maximum number of terminal Fragments that can be generated before the semantic information of each Fragment falls below I_{Fmin};

- m_{min} represents the minimum number of critical Fragments required to achieve the target security condition and the practical loss of inferable logical correlation between terminal Fragments.

Accordingly:

- If $\Gamma > 1$, the Object exhibits high semantic granularity. The practical architecture can approach the ideal asymptotic regime while preserving local

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

semantic admissibility, and emergent security stabilizes before reaching the I_{Fmin} boundary;

- If $\Gamma \approx 1$, the architecture operates near the semantic feasibility limit;

- If $\Gamma < 1$, the Object is not semantically compatible with the required security configuration, and the CNVS instance becomes practically infeasible for that semantic domain.

---

15. Further Developments

In a subsequent implementation development, an optional extension could be envisaged.
In the Extended Model, the introduction of a Functional Micro-Decomposition would significantly increase restriction of knowledge, because it would fragment the I_F Information even more extremely.

Let's assume we further fragment the I_F following the formulation

$I_F(t) = (q_1, q_2, \ldots, q_m)$.

Each External Auditor could access only

$K_{VF}(t) \preceq q_m$, or better

$K_{VF}(t) \subseteq Obs(q_m)$,

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

with $K_{VF}(t) \nRightarrow I_F(t)$

since the Verifier can no longer directly access $Enc(I_F(t))$.

This is because the terminal Fragment would not be assigned as a Complete Structured Unit with I_F content, but would be Functionally Destructured into its constituent Components, such that

$I_F=(q_1, q_2,..., q_m)$

where q_1, q_2, …, q_m constitute the elementary functional components of the I_F Information (for example, q_1= length of a fraction of the arm, q_2= weight of a part of the Object, q_3= purity of a fraction of it, q_4= volume of the bottom residue, etc…)

From this perspective, qm is no longer an autonomous Structured Fragment with a “complete” meaning, but a still elementary functional verification component.

In this advanced Model, the Verifier does not know the node, does not know the level, does not know the fragment, does not know the complete operational meaning even of what it is measuring, but only knows a Specific Elementary Verification Function, as the reconstruction of the Fragment itself that carries the I_F Information becomes an emergent Systemic property.
Thus, the reconstruction of the Structured Fragment occurs only at the System level.
Formally

$I_F(t) \notin K_{VF}(t)$

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

but

$q_m \in K_{VF}(t)$

and

$$I_F = \Phi(q_1, q_2, \ldots, q_m) \quad \textbf{(101)}$$

where

Φ

is the reconstruction function managed by the System.

New Information Levels Table

Level	Rebuildable from
q_m	Local Verifier
I_F	System
I_N	Higher Layer
I_G	Global System

In this model, the verifier is not even able to fully understand the meaning of the component it is measuring.

This can have interesting implications, as it significantly increases the security of the System, without violating the CNVS theory, but by fully integrating it into its architecture.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

16. The Emergent Security Theorem

16.1 Introduction

In the following discussion, we assume an approximation model in which the External Verifier's knowledge is attributable to knowledge of the Terminal Fragment and part of its content D_i.
This assumption makes the result even more robust, because in the non-approximated CNVS model, the External Verifier has less information to establish potential collusion.
The approximation therefore leads to an underestimation of the true security potential of CNVSs based on the results shown.

In this document, the Security of a System is defined as the Property of the System to resist attacks by Colluding External Agents attempting to violate it.
In this document, the term "Emergent" is used to indicate a Property of the System that emerges as a consequence of its intrinsic structure and is strengthened by its progressive adoption.
In this sense, the Emergent Security Theorem defines the Property of the System to resist attack attempts by Colluding External Agents that share the I_F Information assigned by the System with each other, in order to violate it.
The Theorem proposes an emergent Property such that the intention of the Colluding Agents clashes with the growth of the System's Resistance to violation with its progressive adoption and with the increase in the I_G content and therefore its complexity.
To understand the Theorem, we must use the tools that mathematics offers us.
Let's imagine a situation in which a number n_{bad_agent} of Colluding External Agents wants to breach the System at time t.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

16.2 Development

Let's imagine that each of them has access to the share of Fragmental Information $I_F(t)$ entrusted to them by the System, in the form of Fragmental Knowledge $K_F(t)$, and that they have the Encryption Keys, lent by the System itself, to decode the Fragment so as to fully understand its information content.

At this point, the Probability of Breaching the System will depend on:
- the amount of Information $I_F(t)$, in the form of Fragmental Knowledge $K_F(t)$, that each of them has;
- the random coherence of the individual shared Fragments: if N_{bad_agent} External Agents has "nearby" Fragments within the granular I_G Distribution, it is clear that the reconstruction of the original Information will be more likely than in the situation where External Agents share completely opposite Fragments within the same Distribution;
- the complexity of the System, understood not only as the total Volume of I_G Information, but also by the Depth of Progressive Decomposition achieved, which is equivalent to the total number of Fragments (Fragments(t) = $f_x(t)$) generated by the System;
- the degree of Entropy H(X) achieved by the System in its terminal Fragmentation.

At this point, the use of three mathematical tools allows us to demonstrate the Validity of the Hypothesis of the Theorem.
Shannon's Formula, as we saw in Chapter 12, explores the System's uncertainty in identifying the correct configuration of the terminal Fragments during the reconstruction, providing a measure of the Entropy H(X) achieved by the I_G Information during the granulation process.

Bernoulli's Formula helps us in the Fragment assignment phase and allows us to calculate the probability that a Fragment will be randomly assigned to a Colluding External Agent, being a probability statistic in which an event can

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

have only two outcomes (in our case, assigned to the Colluding External Agent or to the Honest External Agent).

Kullback-Leibler divergence measures the statistical "distance" between two probability distributions: in our case, we can use it to calculate the divergence between the critical Bernoulli distribution and the actual Bernoulli distribution. This statistical distance is a central point of the Theorem: that is, it evaluates when a specific Colluding Distribution could become a problem if it attacks the System, by comparing the Actual Colluding Distribution found in the System and the one considered Critical, that is, the Theoretical Distribution with a Critical Threshold sufficient to violate the System and corrupt it.
We therefore have:
Shannon Formula

$$H(X) = -K \Sigma_{i=1}^{N} p_i \log_2 p_i,$$

described in Chapter 10,
Bernoulli's Formula

$$P(Y=1) = q$$

$$P(Y=0) = 1 - q$$

which measures
event 1 = a fragment is assigned to a Colluding External Agent.
event 0 = a fragment is assigned to an Honest External Agent.
The Kullback-Leibler divergence for discrete cases is

$$D(P \| Q) = H(P, Q) - H(P)$$

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

where two Entropies are compared to each other, the Cross Entropy and the real Entropy of System P, that is, the Shannon Entropy which we have already seen and calculated in chapter 12.

Real Entropy $H = -\Sigma_x P(x) \log P(x)$

measures the intrinsic Uncertainty of the Real Distribution P, thus offering a measure of the Disorder of the Data (i.e. how “unpredictable” they are);

Cross Entropy $H(P, Q) = -\Sigma_x P(x) \log Q(x)$

It is a cross-measure, which measures the resulting Entropy if the events of the Real Distribution P are encoded using the approximate Model given by the Q Distribution.

If, for example, we wanted to measure tomorrow's weather, and we know that the Probability of Sunny is 70%, Cloudy is 20%, and Rain is 10%,
while our Estimate is 40% Sunny, 40% Cloudy, 20% Rain,
we would have
the Real Discrete Distribution 0.70, 0.20, 0.10 (A)
against
the Approximate Distribution 0.40, 0.40, 0.20 (B).
Cross Entropy is the comparison between these two distributions, obtained by measuring what happens to the Shannon Entropy of the Real Distribution (A) if I use the data from the Approximate Distribution (B). The result, expressed in bits, tells us how inefficient the Approximation is, that is, how many more bits of information I am "wasting" because the Q Model (which corresponds to B in the Weather example) does not adhere perfectly to the P Model (A in the example).

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

In our case, we will have

$D(\tau \| q) = H(\tau, q) - H(\tau)$, which by replacing becomes

$$D(\tau \| q) = \tau \log (\tau / q) + (1 - \tau) \log ((1 - \tau)/ (1 - q))$$

where

τ, it is the minimum fraction of coherent information needed to attack effectively;

q, is the actual share of colluders .

Said differently $D(\tau \| q)$ measures how unlikely it is that an entity with a collusive share q will behave as if it had a collusive share τ.

For example, if q = 10% and the threshold required for Collusion Success is 30%,
that is, q = 0.10 and τ = 0.30,
the Kullback-Leibler divergence measures how statistically distant 10% is from 30%.

In the proof, we introduced the value τ, the Critical Collusion Threshold. This value constitutes the minimum fraction of Global information that colluders must control to breach the System effectively and consistently. Put another way, if colluders control less than τ, they do not have enough information to cause problems. However, if they control an amount of information with assigned $I_{F_s} \geq \tau$, they can conduct an effective attack.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

This classical approach allows us to calculate the resistance of CNVSs to probabilistic attacks, according to the "Byzantine model."
However, it became clear during the discussion that while standard statistical reconstruction bounds operate below a critical threshold of $\tau = 0.35$ based on independent binomial distribution bounds, rigorous application of Axiom III alters the asymptotic behavior of the System.

In CNVS, validity is not reducible, being governed by the VGlobal function, so the effective operating threshold is not dominated by the classic "majority vote."

Even when the fraction of compromised verifiers is very large, security may still hold if the uncompromised useful information remains sufficient to violate unauthorized reconstruction conditions.

Let's continue our reasoning on the probabilistic model and test the resistance of the System by considering Collusion without taking into account for a moment the violation of the invariants.

Let us assume that $q, \tau \in \mathbb{R}$ and $q, \tau \in (0, 1)$, and $n, c \in \mathbb{N}$

c = the number of Fragments controlled by the Colluding Agents
n = the total number of Fragments;
then the share controlled by the Colluding External Agents will be c/n.
Collusion will only be possible when $c/n \geq \tau$.
τ is a Model Parameter and can be estimated in various ways, but it is a parameter that should be provided.

$\tau \in (0, 1)$ is the minimum normalized amount of useful information required for unauthorized reconstruction.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

Technically, it could be assumed by the Implementation in a "conservative" manner, but the most effective solution is for it to be estimated conceptually by analyzing how much minimal information is needed to reconstruct relationships, avoid contradictions, and violate the System's invariants. It is clear that this is difficult to calculate in an Implementation because it depends on the Nature of the Data we are dealing with, which is assumed to be abstract in this document.

Theoretically, it could also be estimated experimentally through Attack trials. If we assume through analysis and simulations that at least 35% of the useful information is needed to violate the System, τ will be equal to 0.35, meaning that those who control less than 35% of the information will, on average, fail to violate the System.

As the System grows, the information I_G grows, and therefore the number of Fragments increases, the Entropy H(n) increases, and K_F decreases.
It follows that an increasingly larger amount of information is needed to reconstruct the System, thus requiring more colluders.
Formally,

$$\tau = f\tau(H, R, C) \quad \textbf{(102)}$$

where we have
- H = System Entropy;
- R = Structure of relations;
- C = System invariants.

That is, the Critical Threshold τ is a function fτ of the informational and structural complexity of the System, therefore of H, R, and C.
The threshold τ may depend on the informational Entropy H, relational complexity R, and the System invariants C.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

In the proof, we will use Bernoulli's Binomial Formula, which considers many independent Bernoulli trials.
For analytical tractability, the assignment of fragments is modeled as an idealized probabilistic process in which each fragment is independently assigned to a colluding verifier with probability q. This assumption provides a simplified approximation used to derive asymptotic security bounds.

Formally,
$P(X=k) = (n; k)\, q^k\, (1-q)^{n-k}$
where:
- n, the total number of Fragments;
- k, the number of Fragments managed by the colluders;
- q, the probability that a single fragment goes to a colluder;
- P(X=k) the probability that k Fragments are assigned to the Colluding External Agents.

To recap, with the simple Bernoulli's Formula, the probability that a single fragment goes to a colluder is established; Bernoulli's Binomial Formula calculates the probability that out of all the Fragments (i.e., in a distribution), a quantity k goes to the colluders.
In the demonstration we will use Stirling's Formula that we have already seen in the Discussion on Entropy in chapter 12, namely

$n! \approx \sqrt{(2\pi n)} * (n/e)^n$, and which constitutes an approximation when calculating the factorials of very large numbers.

As has clearly emerged, even treating the System as a probabilistic distribution deprived of invariant constraints, the resistance of the model grows with adoption.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

17.3 Proof

The statement of the Emergent Security Theorem is:

$$q\bar{\omega} < \tau \implies \lim_{n \to \infty} P\left[\Sigma^{n}_{i=1} Y_i \omega(D_i) \geq \tau n\right] = 0 \qquad \textbf{(103)}$$

The more the System is adopted, the more useful knowledge is dispersed, and the more unlikely, costly, and technically inconvenient collusion becomes.
Let's start with the idea of having a certain number of terminal Fragments n.
For simplicity's sake, let's assume that they are all equally probable.

This simplifying assumption is introduced only to derive the Entropy of the System in a closed and intuitive form. The theorem is subsequently generalized by introducing an informational weight $\omega(Di) \in [0, 1]]$ for each fragment, thereby explicitly accounting for non-uniform fragment distributions. The uniform case therefore represents a special case of the general model, corresponding to $\omega(di)=1$ for all Fragments.

If we want to recombine them all, the total number of all possible combinations corresponds to n!, and these are Permutations, that is, taking each individual Fragment only once, and taking into account the importance of the order in which they are distributed. In fact, the Fragments contain information and cannot be distributed in a random order.
As we have already seen in Chapter 12, the formula for calculating each individual Probability of a specific Permutation will be $p = 1/ n!$.

Let Shannon Entropy be $H = -\Sigma_i \; p_i \log_2 (p_i)$,

let's calculate the Entropy of our distribution of terminal Fragments n obtained from the System, and substitute the terms

$$H(n) = -\Sigma^{n!}_{i=1} \; 1/ n! \log_2 (1/ n!),$$

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

simplifying the same terms, we will have

$H(n) = -n! \cdot 1/n! \log_2 (1/n!)$,

simplifying further we will have

$n! \cdot 1/n! = 1$,

therefore,

$H(n) = -\log_2 (1/n!)$,

since in logarithms the property $\log(a^{-1}) = -\log(a)$ holds

$a^{-1} = 1/a$, we will still have

$-\log_2 (1/a) = \log_2 (a)$, and let's set $a = n!$

we will have that the above relation can be reduced to

$H(n) = \log_2 (n!)$.

But as the Fragments increase, the possible combinations also increase, so the reconstruction uncertainty ultimately increases. We've already seen this in Chapter 12.
Let's construct the limit, and then we have $n \to \infty \Rightarrow H(n) \to +\infty$

$\lim_{n \to \infty} H(n) = \lim_{n \to \infty} \log_2 (n!) = +\infty$.

Let's move on to using Bernoulli's distribution to randomly assign each individual Terminal Fragment (simple formula) and calculating the probability

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

that, of all the n distributed Fragments, some will end up in the hands of the Colluding External Agents (binomial formula).

For the purposes of proving the theorem, we use Bernoulli's binomial distribution to calculate the probability that a number k of Fragments will be assigned to the Colluding External Agents. The value of k subsequently determines the reduction in the Entropy of the System.

So, let's first calculate the probability that, by chance, each individual Terminal Fragment will end up in the hands of a Colluding External Agent instead of an Honest External Agent.
When the System assigns this fragment, only two things can happen: either the fragment goes to a Colluding External Agent or it goes to an Honest External Agent.
There are no other possibilities.
Let's introduce the variable Y, which can take only two values.
Let's assume that
Y = 1, if the Fragment randomly goes to the Colluding External Agent,
Y = 0, if the Fragment randomly goes to the Honest External Agent.

Let q be the probability that a single terminal Fragment ends up in the hands of a Colluding External Agent, and we have
$P(Y=1) = q$.
Since there are only two possibilities, the opposite probability is
$P(Y=0) = 1 - q$.
At this point, we use Bernoulli's Binomial Formula to calculate the probability of multiple Fragments (potentially all):
$Y_1, Y_2, \ldots, Y_n$
by adding them together, we have

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

$X = Y_1 + Y_2 + \cdots + Y_n$
where X represents a random variable that counts the total number of Fragments assigned to the Colluding External Agents.
Suppose we have the following situation (1 = Colluder, 0 = Honest):
$Y_1 = 1$
$Y_2 = 0$
$Y_3 = 1$
$Y_4 = 0$
$Y_5 = 1$
Then we have:
$X=1 + 0 + 1 + 0 + 1 = 3$
which means that out of 5 Fragments, 3 ended up in the hands of the Colluding External Agents.

The probability that exactly k terminal Fragments go to the Colluders is calculated with the following formula:
$P(X=k) = (n ; k)\, q^k\, (1-q)^{n-k}$.
We must pay attention to the meaning of X and k.
X is the random variable that determines the number of Fragments assigned to the Colluder;
k corresponds to that specific number.

In fact $k \in \mathbb{N}$ and $k \in \{0, 1, \ldots, n\}$,

therefore

X = random variable that counts the Fragments to the Colluders.
k = the particular value assumed by the variable X.

Let's return to the proof.
We stated that the System Entropy corresponded to

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

$H(n) = \log_2(n!)$, which corresponds to the Uncertainty of the Entire System.
We should subtract a portion of these Fragments from the System Entropy calculation, because they will end up in the hands of the Colluders, reducing the Entropy.
The System Entropy is reduced because the Colluding External Agents, receiving the Fragments, will know their contents, therefore a portion of the I_G Information will no longer be unknown and will no longer contribute to determining Uncertainty (one thing known is certain).
The Number of Fragments still uncertain and unknown will be equal to n - k, those that ended up in the hands of the Honest External Agents.
The residual Entropy will be
$H_{res}(k) = \log_2((n - k)!)$, counting only the unknown Fragments that ended up in the hands of Honest External Agents.
The Entropy reduction will therefore be
$\Delta H(k) = H(n) - H_{res}(k)$, substituting the values $H(n)$ and $H_{res}(k)$ we will have
$\Delta H(k) = \log_2(n!) - \log_2((n - k)!)$.
Exploiting the property of logarithms that $\log(a) - \log(b) = \log(a/b)$, substituting the terms we will have
$\Delta H(k) = \log_2(n! / (n - k)!)$.
In summary,
- If k = 0, the colluders know nothing,
- If k increases, uncertainty decreases,
- If k = n, uncertainty becomes zero, because everything is known by the Colluding Agents.

So up to this point, Shannon Entropy provides a measure of the total uncertainty associated with the global information I_G.
The probabilistic model based on Bernoulli and binomial distributions estimates the likelihood that a sufficient number of useful fragments is compromised.
Each compromised fragment contributes to a reduction in global uncertainty according to its informational weight $\omega(D_i)$. The cumulative Entropy reduction

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

is represented by ΔH.
The reconstruction threshold τ defines the minimum normalized amount of useful information that must be acquired in order to make unauthorized reconstruction possible.

If the reduction in System Entropy becomes large enough to exceed a Threshold Value, Collusion goes from being an unlikely event to a "dangerous" event. This occurs when $k \geq \tau \cdot n$.

The probability that collusion leads to success, thus deteriorating the System's security, is therefore given by
$P(X \geq \tau \cdot n)$

Summary table:
c = k = number of Fragments assigned to the Colluders
$c/n = k/n$
$k/n \geq \tau$
$k \geq \tau \cdot n$
$k\tau = \lceil \tau \cdot n \rceil$
$k \geq k\tau$

Starting from the Binomial Formula, we calculate the total probability of success of a collusion, which will therefore be given by the sum of all the probabilities of the cases in which the Colluders receive enough Fragments to reach or exceed the Critical Threshold.
The probability of success of the collusion will therefore be:
$P_{coll}(n) = P(X \geq k\tau)$, that is,

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

$P_{coll}(n) = \Sigma^{n}_{k=k_\tau} P(X = k)$

with X (= Y1 + Y2 + ⋯ + Yn) representing the variable referring to the total number of Fragments assigned to the Colluding External Agents and k a specific number, we will have

$k_\tau = \lceil \tau \cdot n \rceil$ = Critical Threshold value. **(104)**

Substituting we will have,

$P_{coll}(n) = \Sigma^{n}_{k=\lceil \tau \cdot n \rceil} (n ; k)\ q^k (1 - q)^{n-k}$. **(105)**

The formula to be proved was:

$q\bar{\omega} < \tau \implies \lim_{n \to \infty} P [\Sigma^{n}_{i=1} Y_i \omega (D_i) \geq \tau n] = 0$

$q < \tau \implies \lim_{n \to \infty} P_{coll}(n) = 0.$ **(106)**

with

$X \sim Bin(n, q)$,

that is, X is the variable of the Binomial, which counts how many Fragments end up with the Colluders,
n, the total number of Fragments,
q, the Probability of success of a single Bernoulli Trial, i.e., $q = P(Y_i = 1)$, which is the Probability that a single Fragment will be randomly assigned to the Colluder.
The expected value is: $E[X] = n \cdot q$,
so the Colluders receive on average $n \cdot q$ Fragments.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

The minimum threshold is $k = k_\tau = \lceil \tau \cdot n \rceil$,
but if
$q < \tau$
then,
$n \cdot q < \tau \cdot n$
that is, the average number of Fragments received by the Colluders is less than the threshold required to violate the System.
Therefore,
X/ n will be the fraction of the total Fragments that end up with the Colluders.
Collusion is therefore successful if $X/n \geq \tau$,
so substituting we have
$P_{coll}(n) = P\ (X/n \geq \tau)$.
By the law of large numbers,
if $n \to \infty$
we have
$X/n \to q$,
that is, if the number of Fragments grows infinitely, the effective and real fraction of Fragments assigned to the Colluders increasingly approaches Probability q, which is the probability that a single fragment is assigned to a Colluder: therefore, the observed fraction converges to the actual value q.
But if the real fraction approaches q, and q is smaller than τ, then it becomes increasingly unlikely that
$X/ n \geq \tau$, so substituting we have:

$$q < \tau \implies \lim_{n \to \infty} P(X/ n \geq \tau\ ) = \lim_{n \to \infty} P_{coll}(n) = 0, \text{ therefore}$$

$$q < \tau \implies \lim_{n \to \infty} P(X \geq \lceil \tau \cdot n \rceil\ ) = 0, \text{ by replacing we will have}$$

$$q < \tau \implies \lim_{n \to \infty} \Sigma^{n}_{k=\lceil \tau \cdot n \rceil}\ (n\ ;\ k)\ q^{k}\ (1 - q)^{n-k} = 0,$$

That is,

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

if $p\bar{\omega} < \tau$, where p is the fraction of colluding verifiers and $\bar{\omega}$ is the average informational weight of compromised fragments, then increasing the number of fragments makes it increasingly unlikely that the colluders will obtain enough useful information to exceed the reconstruction threshold.

Let's return to Kullback-Leibler divergence, which we saw at the beginning of this chapter:

$$D(\tau \| q) = \tau \log(\tau / q) + (1 - \tau) \log((1 - \tau)/(1 - q)) \quad \textbf{(107)}$$

D measures how far the actual situation q is from the critical level τ, that is, from the situation in which the Critical Threshold level for assigning Fragments to Colluders is actually reached, or
it measures how unlikely it is that a binomial variable with actual probability q will produce an observed frequency equal to at least τ.
For a binomial, the Chernoff Inequality holds, which is its classic application. When the number of independent trials (in this document, corresponding to Fragments) is very high, calculating the Binomial is complex.

The Chernoff Inequality allows us to obtain an upper and a lower limit with respect to the mean of the Probability of the Binomial, thus obtaining an Interval within which the expected Probability of the Binomial falls:

upper bound $P(X \geq (1 + \delta)\mu) \leq e^{-\delta^2 \mu / 3}$

lower bound $P(X \leq (1 - \delta)\mu) \leq e^{-\delta^2 \mu / 2}$

That is, the further one moves from the mean (and therefore δ is large) or the more one increases the number of trials (which correspond to μ), the more the probability of an anomalous event (with respect to the expected value from

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

the binomial) collapses towards zero.
Applied to a binomial, we can therefore calculate the probability of departing from the expected interval: $P(|X - \mu| \geq \delta\mu) \leq 2e^{-\delta^2 \mu / 3}$.

Applying the Chernoff bound to the upper tail of the binomial distribution we obtain:

$e^{-nD(\tau\|q)}$

therefore

$$P(X/n \geq \tau) \leq e^{-nD(\tau\|q)} \quad \textbf{(108)}$$

if $\tau > q$,

since $P_{coll}(n) = P(X/n \geq \tau)$,

we can also write

$$P_{coll}(n) \leq e^{-nD(\tau\|q)}, \quad \textbf{(109)}$$

that is, the probability that the colluders exceed the critical threshold (= Pcoll (n) = P (X/ n ≥ τ)) drops exponentially the greater the number of trials (i.e. Fragments) n, and the greater the distance between q and τ calculated with the Kullback-Leibler divergence.

Let's say $Z = D(\tau\|q)$ so as to simplify writing the formula.

if $Z > 0$

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

then $P_{coll}(n) \leq e^{-nZ}$

as n increases, the exponent becomes more and more negative:

assuming $Z = 0.1$

$n = 10 \quad \rightarrow e^{-1}$

$n = 100 \quad \rightarrow e^{-10}$

$n = 10000 \rightarrow e^{-100}$

The value will tend exponentially to 0.
Since, $Z > 0$, we have:

$\lim_{n \to \infty} e^{-nZ} = 0.$

Since $0 \leq P_{coll}(n) \leq e^{-nZ}$

and for e^{-nZ} holds $\lim_{n \to \infty} e^{-nZ} = 0$,

therefore $P_{coll}(n) \rightarrow 0$ for $n \rightarrow \infty$, concluding

$q < \tau \implies \lim_{n \to \infty} P_{coll}(n) = 0.$

In reality, not all fragments have the same weight.

Some fragments are more useful than others because they are better connected and because they convey a greater amount of information about the Global Information.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

Let's formalize the concept.
A fragment is more useful the more it represents the Global Information, i.e., the greater its contribution to Global knowledge.
Each useful fragment therefore reduces the Global Entropy of the System, meaning that the more useful (i.e., heavier) a fragment is, the more it reduces the Uncertainty of the Entire System.

Let
G
the random variable representing the distribution of the System's globally admissible configurations,

let
$\omega(D_i)$
the information weight of the Fragment (or Utility), we will have:

$$\omega(D_i) = I(G; F_i) \quad \textbf{(110)}$$

where
- F_i is the fragment that contains D_i;
- $I(G; F_i)$ is the mutual information between the Fragment and the admissible Global Distribution.

To put it another way:

$$\omega(D_i) = H(G) - H(G|F_i), \quad \textbf{(111)}$$

that is, the Utility or information weight of the Fragment is given by the Global Entropy net of the Conditional Entropy, that is, the measure of how much uncertainty remains in G after observing F_i.
The domain of ω will therefore be:

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

$0 \leq \omega(D_i) \leq H(G)$,

but for its subsequent uses, it is convenient to normalize its value so as to contain it in the range [0, 1]

$$\omega(D_i) = I(G; F_i) / H(G), \quad \textbf{(112)}$$

with the domain that becomes

$0 \leq \omega(D_i) \leq 1$, such that
if its value is equal to
0, the Fragment is useless;
1 represents the maximum information content (and the Global Uncertainty drops to zero),
while intermediate values have proportional utility.

So,
if Y_i is the indicator variable for the random assignment of the Fragment,
we have
$Y_i\omega(D_i)$,
where
$Y_i = 1$ if the fragment is assigned to a Colluding Verifier,
$Y_i = 0$ if it is assigned to an Honest Verifier.
In detail, for the Verifier, regardless of whether it is Honest or Colluding, we have:
$\omega(D_i) \in [0, 1]$,

if $\omega(Di) \approx 1$, the fragment is very useful for information reconstruction;
if $\omega(D_i) \approx 0$, the fragment is completely disconnected from the rest and has no real contribution to global information reconstruction;

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

Clarification:
The Useful Fragment only counts if it goes to a Dishonest Verification Agent: the count is based on the weight of its informational utility.

$$\bar{\omega} = 1/n \ \Sigma^{n}_{i=1} \ \omega(D_i) \qquad \textbf{(113)}$$

where $\bar{\omega}$ represents the average value of the informational utility of the fragments.

Therefore, $q\bar{\omega}$
represents the actual collusive share, that is, the correct share of the weight of the average value of the informational utility of the fragments.

Therefore, if
$q\bar{\omega} < \tau$,
the expected value remains below the critical threshold if

$$E\left[\Sigma^{n}_{i=1} \ Y_i\omega(D_i)\right] = q\,\Sigma^{n}_{i=1}\,\omega(D_i) = nq\bar{\omega}, \qquad \textbf{(114)}$$

where E denotes the expected value, or the theoretical average amount of useful information controlled by collusionists.

In conclusion, the formula
$q < \tau \Longrightarrow \lim_{n \to \infty} P_{coll}(n) = 0$,

which is explicitly equivalent to

$$q < \tau \Longrightarrow \lim_{n \to \infty} P(X \geq \lceil \tau \cdot n \rceil) = 0,$$

by adding the value $\omega(D_i)$ for each Y_i to X and substituting,
we finally have:

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

$q\bar{\omega} < \tau \Longrightarrow \lim n \to \infty P\,[\Sigma n\ i=1\ Y_i \omega\,(D_i) \geq \tau \cdot n\,] = 0,$
per $0 \leq \omega\,(D_i) \leq 1$.

Therefore:

if the actual collusive share, weighted by the average informational utility of the Fragments, remains below the critical threshold, then, as the number of Fragments increases, the probability that colluders obtain enough useful information to compromise the System tends to zero.

---

17. Considerations on Dependent Collusion

17.1 Introduction

The Chernoff exponential bound introduced in the previous chapter models unauthorized reconstruction by simplifying the assumption that fragment compromise events are statistically independent and, consequently, compromise events are treated as statistically independent. This approximation is useful for illustrating the asymptotic scalability of security, but it does not explicitly capture coordinated collusion between adversarial verifiers.

In the real world, collusions are more often coordinated and, being able to share all the information they acquire, organize their strategies to compromise the System adaptively, that is, by targeting the most informative fragments. A more general formulation must therefore account for the statistical dependence between compromise events.

The purpose of this chapter is to demonstrate that the central conclusion of

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

the CNVS security model remains valid even in the presence of coordinated collusion, provided that the conditional probability of compromising each additional critical fragment is uniformly bounded by a constant strictly less than one.

17.2 Development

Let:

- Q denotes the total number of verifiers;

- C denotes the total number of colluding verifiers;

- q = C/ Q denote the collusive control fraction;

- p_{inf} indicates the probability that a critical fragment not directly controlled by collusive entities can still be inferred from compromised information;

- m indicates the number of critical fragments with low inference required for unauthorized reconstruction.

A fragment is considered critical if, having low inference, it cannot be easily inferred from knowledge of all the others, therefore its knowledge becomes critical for reconstruction.
We define the probability of compromise of a single fragment as

$$p_{comp}=q + (1-q)\, p_{inf}. \quad \textbf{(115)}$$

This quantity represents the probability that a single critical fragment is either directly controlled by colluding entities or successfully inferred.

For each critical fragment F_i, let C_i denote the event that F_i is compromised.

We assume that, for every critical fragment,

$$P(C_i \mid C_1,\ldots,C_i-1) \le p_{comp}, \quad 0 \le p_{comp} < 1. \quad \textbf{(116)}$$

This condition does not assume independence. It only requires that, even after any sequence of prior compromises, the probability of compromising one additional critical fragment remains bounded by a constant strictly smaller than one.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

It is clear that this assumption becomes a fundamental prerequisite for guaranteeing Security, because if the probability of compromising the residual critical fragments not yet controlled by the Colluders were equal to 1, it would mean that the System has been completely violated: in this case we would no longer have a probability, but the certainty that the residual Information is completely accessible to the Colluders or in any case deducible.

Let Rec^* denote the event of unauthorized full reconstruction of the global informational state.

Under the stated assumptions, unauthorized reconstruction is bounded by

$$P(Rec^*) \leq p_{comp}{}^m \quad \textbf{(117)}$$

therefore:

unauthorized reconstruction requires compromise of all m critical fragments:

$Rec^* = C_1 \cap C_2 \cap \cdots \cap C_m$.

By the chain rule of probability,

$P(Rec^*) = \prod^m_{i=1} P(C_i \mid C_1, \ldots, C_i - 1)$

Applying the uniform conditional bound,

$$P(Rec^*) \leq \prod^m_{i=1} p_{comp} = p_{comp}{}^m. \quad \textbf{(118)}$$

If

$0 \leq p_{comp} < 1$,

then

$P(Rec^*) \to 0$ as $m \to \infty$.

This establishes that the essential conclusion of the Emergent Security Scaling Theorem remains valid even when colluding adversaries are fully coordinated.

At this point, it is necessary to make an important clarification.

As explained in previous chapters, the number of fragments cannot tend to infinity, because with each recursive decomposition, the semantic content of each fragment is reduced to zero.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

This too can be understood as an approximation.

Elsewhere in this document, the limit I_{min} (or I_{Fmin}) denotes the semantic lower bound below which a terminal fragment no longer contains enough information for honest verification.
It is sufficient for the decomposition to asymptotically approach that limit, so as to disentangle all I_F fragments from any externally inferable local and global structural relationship, keeping the colluder blind both locally and globally, even if it shares its information with that of other colluders.
Therefore,
security increases with the number of critical fragments up to the limit imposed by the minimum semantic threshold I_{min}.

So to recap, the asymptotic expressions derived in the previous sections, including

$$P(Rec^*) \to 0 \quad \text{as} \quad m \to \infty$$

and

$$q\bar{\omega} < \tau \implies \lim_{n\to\infty} P\left[\Sigma_{(i=1 \text{ to } n)} Y_i \, \omega(D_i) \geq \tau n \right] = 0$$

should be interpreted as idealized scaling limits:

these results establish the theoretical direction of the security model, therefore under the stated assumptions, increasing the number of sufficiently independent low-inference fragments reduces the probability of unauthorized reconstruction.

It is the task of the System, or rather of its implementation, depending on the nature of the Information, to make the terminal Fragments increasingly independent and reduce their inference to a minimum.

Formally,

let

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

I_{min}

denote the minimum amount of information required for semantic task comprehension.

Each terminal fragment must satisfy

$I_F \geq I_{min}$.

If this condition is violated, the fragment becomes operationally meaningless and local verification is no longer possible.

The semantic lower bound imposes a practical upper limit on fragmentation depth.

Let

m_{max}

denote the maximum number of critical fragments that can be generated while preserving

$I_F \geq I_{min}$.

This quantity depends on:

- the total informational content of the global state;

- the semantic complexity of each verification task;

- the decomposition strategy;

- the informational heterogeneity of fragments.

From the generalized dependent-collusion model, the minimum number of critical low-inference fragments required to achieve a target security level

$P(Rec^*) \leq \eta$

$m_{min} = \lceil \ln(\eta) / \ln(p_{comp}) \rceil$,

(119)

for

$0 < p_{comp} < 1$.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

A CNVS architecture is practically feasible if and only if

$m_{max} \geq m_{min}$.

This inequality states that the System can generate enough critical low-inference fragments to achieve the desired security level without violating the semantic lower bound required for honest verification.

The condition above transforms the asymptotic security theory into a concrete design criterion.

The practical engineering objective is therefore:

maximize epistemic restriction subject to semantic verifiability.

In other words, the System should fragment information as aggressively as possible while ensuring that each terminal fragment remains sufficiently meaningful for the assigned verification task.

If

$m_{max} \geq m_{min}$,

the desired security level is achievable.

If

$m_{max} < m_{min}$,

the architecture does not provide sufficient fragmentation to satisfy the target security requirement.

In that case, security can still be improved by reducing

p_{comp},

for example by:

- decreasing the fraction of colluding verifiers q;

- reducing the inference probability p_{inf};

- increasing fragment heterogeneity;

- strengthening the non-inferability of critical fragments.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

This feasibility condition does not weaken the CNVS theory.

On the contrary, it explicitly links:

- semantic constraints;

- probabilistic security;

- System scalability;

- practical implementation.

As a result, the CNVS framework provides not only asymptotic theorems, but also a concrete quantitative criterion for determining whether a proposed architecture can simultaneously satisfy operational and security requirements.

If critical fragments are non-inferable, i.e.,

$p_{inf}=0$

then

$p_{comp}=q$

and therefore

$$P(Rec^*) \leq q^m \quad \textbf{(120)}$$

This corresponds to the ideal case in which each critical fragment acts as an independent veto condition for unauthorized reconstruction.

Suppose a target upper bound η is specified for the probability of unauthorized reconstruction:

$P(Rec^*) \leq \eta$

The minimum number of critical low-inference fragments required to satisfy this condition is

$m_{min} = \lceil \ln(\eta)/ \ln(p_{comp}) \rceil$

and where

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

η is the maximum tolerated probability coefficient of unauthorized reconstruction,

with $\eta \in (0, 1)$,

let $0 < p_{comp} < 1$

the ratio $\ln(\eta)/ \ln(p_{comp})$

is well defined and strictly positive,

The security condition

$p_{comp} m \leq \eta$

holds if and only if

$m \ln(p_{comp}) \leq \ln (\eta)$

because $\ln(p_{comp}) < 0$ division by $\ln(p_{comp})\backslash\ln(p_{comp})$ reverses the inequality:

$m \geq \ln(\eta)/ \ln(p_{comp})$.

This expression provides a direct design rule linking security objectives to architectural parameters.

Example.

Assume:

- $q = 0.10$,
- $p_{inf} = 0.01$

Then:

$p_{comp} = 0.10 + 0.90 \times 0.01 = 0.109$

If:

- $m = 20$

then:

$P(Rec^*) \leq 0.109^{20} \approx 5.7 \times 10^{-20}$.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

Thus, even with coordinated collusion and nonzero inference probability, unauthorized reconstruction becomes negligibly probable when a sufficient number of critical fragments remains protected.

This chapter generalizes the original Chernoff-based analysis by removing the independence assumption. The result shows that exponential security scaling is not an artifact of independent Bernoulli models, but a structural consequence of maintaining a uniform upper bound on the conditional probability of compromising each additional critical fragment.

In practical terms, the Security Theorem, applied to cases of Dependent Collusion, states that if the System architecture ensures that each new critical fragment remains sufficiently difficult to compromise or infer, then overall security increases exponentially with the number of such fragments, even in the presence of a coordinated attack.

17.3 Relations with Inference

It should be noted, however, that the inference probability p_{inf} can also be interpreted as a function of the residual uncertainty of the global state.

This allows for a more rigorous understanding of the concept of inference.

Let

$$K_{adv} = \Sigma_{i \in A} \omega(D_i) \quad \textbf{(121)}$$

denote the normalized informational weight controlled by the adversarial coalition.

As K_{adv} increases, the residual uncertainty about the global state decreases, and the probability of inferring non-controlled fragments may increase. Therefore one may write

$$p_{inf} = \nu(K_{adv}), \quad \textbf{(122)}$$

where ν is a non-decreasing function. The generalized compromise probability becomes

$$p_{comp} = q + (1-q)\, \nu(K_{adv}). \quad \textbf{(123)}$$

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

Thus,

$$P(Rec^*) \leq [q + (1-q)\, \nu(K_{adv})]^m. \quad \textbf{(124)}$$

This formulation connects the dependent-collusion model with the entropy-based interpretation of fragment utility. It does not replace the generalized security bound, but gives an informational interpretation of the inference term.

17.4 Conclusion

The original Emergent Security Scaling Theorem established the exponential decrease of unauthorized reconstruction probability under an idealized independent assignment model.

The present formulation extends the same principle to coordinated colluding adversaries and dependent compromise events.

Accordingly, the theorem should be interpreted not as a separate foundational principle, but as a generalized form of the original security scaling result.

The Correlated Adversarial Coverage Model strengthens the CNVS framework by demonstrating that emergent security persists under coordinated collusion and statistical dependence.

The essential requirement is that the conditional compromise probability of each additional critical fragment remains uniformly bounded by a constant strictly smaller than one.

Under this condition, unauthorized reconstruction probability decreases exponentially as the number of critical fragments generated by the growth of I_G increases, and the fundamental security conclusions of CNVS remain valid in significantly more realistic adversarial environments.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

18. Scientific Positioning and Relationship with Existing Frameworks

18.1 Introduction

The **CNVS framework** did not emerge in isolation. It sits at the intersection of several established disciplines, including information theory, cryptography, distributed Systems, epistemic logic, and formal methods.
The purpose of this chapter is not to assert superiority over existing frameworks, but rather to precisely identify the similarities and substantial differences between CNVS and the main theoretical models already present in the literature.
CNVS is proposed as an axiomatic framework for distributed and convergent verification-dependent state admission, subject to knowledge restriction constraints, global invariants, and emergent security with scale.

18.2 Information Theory

Conceptual Relationship

Information theory provides the mathematical foundation for:
• entropy;
• uncertainty;
• information content;
• redundancy;
• communication limits.
CNVS uses these concepts to describe how the fragmentation and restriction of knowledge reduce the likelihood of unauthorized reconstruction.

Fundamental Difference

Information theory addresses the issue of the amount of information that can be encoded and its subsequent transmission and compression.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

CNVS, on the other hand, addresses the issue of the validity of the Global State and the conditions for this to be possible.
CNVS is not a theory of communication or encoding, but is first and foremost a theory of the existence of states as dependent on verification.

18.3 Verifiable Secret Sharing and Threshold Encryption

Conceptual Relationship

Verifiable Secret Sharing (VSS) and threshold cryptography distribute shares of a secret among multiple participants and allow the secret to be reconstructed once a minimum threshold is reached.
CNVS also uses:
• fragmentation;
• threshold-based reasoning;
• partial information.

Fundamental Difference

VSS addresses the need to securely manage the distribution and reconstruction of a partially shared secret, but without a verification architecture based on distributed convergence and restricted informational access.

CNVS addresses the problem of establishing the validity of a Global State while preventing unauthorized reconstruction. To do so, it uses External Verification Agents that are not included in the VSS architecture. It also performs Verification through the Data Convergence (D_i) function in a local environment, with V_{Local} acting as a local admissibility operator, accessing the content needed for its local task at each level, but without direct semantic access to the informational payload preserved in D_i.
The architecture, purpose, and structure are completely different.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

In CNVS, fragmentation is not primarily a mechanism for sharing secrets, but a structural mechanism for limiting knowledge, distributing verification, and limiting the inference of Local knowledge from Global knowledge.

18.4 Zero-Knowledge Proofs

Conceptual Relationship

Zero-Knowledge Proofs allow a subject to prove the truth of a proposition without revealing the underlying secret.
CNVS shares the principle of limiting the information accessible to individual agents.

Fundamental Difference

Zero-Knowledge Proofs (ZKP) address the problem of how to prove a truth without revealing a secret. Here, too, there are no independent External Verifiers, nor is verification based on data convergence.
The prover possesses a witness (such as a password, a solution, or a private key) and demonstrates that it is able to satisfy a certain mathematical relation $R(x, w)=1$,
where:
• x = public statement;
• w = secret witness;
• R = verifiable relation.
The prover convinces the verifier that there exists a w such that $R(x, w)=1$, without revealing w.

CNVS addresses the completely different problem of determining how to allow multiple agents with limited local knowledge to analyze the results of a sampling process that has the sole purpose of determining the admissibility of a Global State. Verification does not occur through cryptographic "proofs" or attestations, but through the study of the Object itself.
ZKP does not study the information content of the Object, not even at the

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

protocol level, because it is irrelevant to the purpose.
ZKP is a cryptographic proof protocol.
CNVS is not a prover-verifier protocol, although it shares the principle of information control.

18.5 Secure Multiparty Computation

Conceptual Relationship

Secure Multiparty Computation (MPC) allows multiple parties to compute a function on private inputs while preserving privacy and correctness.
Architecturally, MPC is the framework that may seem closest to CNVS, but it differs radically.

Fundamental Difference

MPC is a model that computes a function $f(x_1,\ldots,x_n)$ without revealing the private inputs, which share its use for a common goal.

CNVS asks only when a Global State S can be accepted as valid.
Even if MPC were used to compare local results D_i, it would not independently provide:

- Verification-Dependent Existence;
- Non-Reducibility of Global Validity;
- Knowledge Restriction Principle;
- Global Veto Principle;
- Emergent Security with Scale.

MPC can therefore be an implementation tool, but it is not equivalent to CNVS.
This is because the MPC model does not perform Verification by convergence of information and data D_i, and does not apply System invariants.
Specifically, MPC does not define a verification architecture based on independent local outputs D_i that converge towards a reconstructed global structured informational state $\mathtt{I}_G$.
In CNVS, the convergence of distributed evidence is a central structural

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

mechanism. Multiple verifiers with limited knowledge produce local results, and the consistency of these results determines whether a global state can be admitted.

Furthermore, the System does not operate primarily through metadata exchange or shared computation, but through the preservation and verification of relational consistency across recursively fragmented structured elements.

MPC distributes private inputs among participants, but does not generally require the recursive decomposition of a single global information state into semantically distinct fragments.
Secure Multiparty Computation (MPC) is not designed to fragment a single global piece of information.
It is designed to allow multiple parties, each with their own independent private input, to jointly compute a function.

CNVS, on the other hand, is explicitly based on the non-uniform recursive fragmentation of a global state. This fragmentation constitutes a fundamental architectural and security principle.
MPC protects private inputs from disclosure during computation. However, its primary goal is the preservation of privacy, not the structural limitation of knowledge as a foundational principle of security.
In CNVS, the limited visibility of fragments is not simply a privacy feature, but a fundamental element of the security model, where the Verifier is not granted access to the elementary data Di, but is instead tasked with performing a measurement procedure that results in a comparison entity with a pre-existing data point.
MPC does not inherently require fragments to have different informational relevance, nor that they be assigned via a non-uniform random distribution.
MPC does not hide the data Di from verifiers and does not issue verification instances.
CNVS explicitly assumes that fragments can possess heterogeneous informational value and that they are randomly and non-uniformly distributed.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

MPC guarantees the correct computation of a function.
CNVS determines whether a candidate Global State satisfies structural invariants and can therefore be admitted as valid. Correctness of the computation alone is not sufficient; a separate verification of global validity is required.
MPC is a distributed computation protocol.
CNVS is a Verification System.

18.6 Byzantine Fault Tolerance and Consensus Protocols

Conceptual Relationship

Byzantine Fault Tolerance (BFT) protocols address the problem of reaching distributed agreement in the presence of malicious actors.
CNVS also considers distributed verifiers and adversarial scenarios.

Fundamental Difference

Consensus protocols ask how multiple nodes can reach a collective agreement.
CNVS asks how a state can be admitted only if the global invariants are satisfied.
In CNVS, agreement is not sufficient, because a state is rejected whenever the global invariants are not respected.
The fragmentation and random distribution of information reduce the possibility of informationally effective agreements between colluders.

18.7 Bitcoin and Blockchain Security

Conceptual Relationship

Bitcoin has shown that large-scale decentralized Systems can achieve probabilistic security that increases with the number of participants.
CNVS similarly formalizes security that increases with the scale of the System.

Fundamental Difference

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

Bitcoin is based on:

- proof-of-work;
- economic incentives;
- longest chain rule.

CNVS, on the other hand, is based on:

- fragmentation;
- knowledge restriction;
- non-inferability;
- global veto;
- state admission based on invariants.

CNVS is therefore conceptually distinct from blockchain consensus.
Neither consensus nor a quorum guarantees the transition of a state request from candidate to existing, but only the application of System invariants and the satisfaction of verification functions at the local and global levels.

18.8 Epistemic Logic

Conceptual Relationship

Epistemic logic studies what agents know and how that knowledge evolves.
CNVS has a central epistemic component, as it explicitly limits verifiers' local knowledge.

Fundamental Difference

Epistemic logic is primarily descriptive and formal.
CNVS uses epistemic constraints as an operational security and verification mechanism.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

18.9 Formal Methods

Conceptual Relationship

Formal methods provide mathematical tools for specifying and verifying Systems.
The formalization of CNVS in Lean 4 belongs to this methodological tradition.

Fundamental Difference

Formal methods are not the object of CNVS, but rather tools used to verify its internal logical consistency.
Formal methods verify the model's consistency; CNVS uses this consistency to define procedures for admitting or rejecting states.

18.10 Complex Systems and Emergence

Conceptual Relationship

Complex Systems theory studies the emergence of global properties from local interactions.
CNVS shares this perspective through the concept of emergent security.

Fundamental Difference

CNVS does not simply observe emergence, but derives explicit structural and probabilistic conditions that guarantee increased security with scale.

18.11 Quick Comparative Summary

Framework	Primary Objective	Relationship with CNVS
Information Theory	Communication and entropy	Mathematical foundation
VSS / Threshold encryption	Distribution and reconstruction of secrets	Shares the fragmentation
Zero-Knowledge Proofs	Demonstration without revelation	Shares the limitation of information
Secure Multiparty Computation	Private distributed computing	Architecturally closer framework, but distinct in terms of Di

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

Framework	Primary Objective	Relationship with CNVS
		convergence, fragmentation, invariants, and global verification.
Byzantine Fault Tolerance	Distributed Agreement	Shares the adversarial context
Bitcoin	Probabilistic decentralized consensus	Shares security that increases with scale
Epistemic Logic	Reasoning about knowledge	Conceptual foundation
Formal Methods	Mathematical verification	Validation tool
Complex Systems	Global Property Emergence	Supports emergent interpretation

---

18.12 Distinctive Contribution of the CNVS

The distinctive contribution of CNVS lies in the unified integration of the following principles:

1. Verification-Dependent Existence;
2. Non-Reducibility of Global Validity;
3. Non-Uniform Recursive Fragmentation;
4. Knowledge Restriction Principle;
5. Global Veto Principle (i.e., rejection of a state even in the case of a single unsatisfied convergence and violation of invariants);
6. Emergent Security with Scale.

No single existing framework combines these principles with the specific goal of admitting global states in distributed Systems.

18.13 Final Positioning Declaration

CNVS should therefore be interpreted as an axiomatic framework proposed for distributed verification and state admission.

It integrates concepts from information theory, cryptography, distributed Systems, and epistemic reasoning, but addresses a distinct theoretical question:

Under what conditions can a globally consistent state be admitted in a distributed System with limited local knowledge?

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

This question defines the original core and specific theoretical field of the CNVS framework.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

Appendices

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

Appendix Q – Tables λ

CNVS Fragmentation Tables Based on $\mathcal{T}$

Progressive variation of λ and Composite Information Density ρ_C

Formula and simplifying assumptions

Formula (86): $\text{Fragments}(t) = |\mathcal{T}(t, \xi)| = \lceil \lambda \cdot [1/(\rho_C(t) \cdot (N(t)^{\beta_{-N}}))] \cdot \log_2(1 + (I_G(t) + I_N(t))/I_0) \rceil$

Assumptions used in these tables: $I_N = I_G$; I0 = 100,000 bits; N(t) = 1, so $N(t)^{\beta_{-N}} = 1$. Values are rounded upward with the ceiling function.

Interpretation: λ acts as a fragmentation-intensity multiplier; lower ρ_C values impose more Fragments; increasing I_G grows the number of Fragments logarithmically under the chosen formula.

Table — Composite Information Density $\rho_C = 0.30$

$I_G = I_N$	λ=1	λ=2	λ=3	λ=4	λ=5	λ=6	λ=7	λ=8	λ=9	λ=10	λ=11	λ=12
100 kbit	6	11	16	22	27	32	37	43	48	53	59	64
200 kbit	8	16	24	31	39	47	55	62	70	78	86	93
300 kbit	10	19	29	38	47	57	66	75	85	94	103	113
400 kbit	11	22	32	43	53	64	74	85	96	106	117	127
500 kbit	12	24	35	47	58	70	81	93	104	116	127	139
600 kbit	13	25	38	50	62	75	87	99	112	124	136	149
700 kbit	14	27	40	53	66	79	92	105	118	131	144	157
800 kbit	14	28	41	55	69	82	96	109	123	137	150	164
900 kbit	15	29	43	57	71	85	100	114	128	142	156	170
1000 kbit	15	30	44	59	74	88	103	118	132	147	162	176

Comment. At $\rho_C = 0.30$, the number of Fragments increases almost linearly with λ: at $I_G = I_N = 1{,}000$ kbit, the value rises from 15 Fragments at λ = 1 to 176 Fragments at λ = 12. For a fixed λ, the increase with I_G is controlled and logarithmic, because I_G and I_N enter the formula through the $\log_2$ term. Lower values of ρ_C require stronger dispersion and therefore generate a larger number of Fragments.

Table — Composite Information Density $\rho_C = 0.25$

$I_G = I_N$	λ=1	λ=2	λ=3	λ=4	λ=5	λ=6	λ=7	λ=8	λ=9	λ=10	λ=11	λ=12
100 kbit	7	13	20	26	32	39	45	51	58	64	70	77
200 kbit	10	19	28	38	47	56	66	75	84	93	103	112
300 kbit	12	23	34	45	57	68	79	90	102	113	124	135
400 kbit	13	26	39	51	64	77	89	102	115	127	140	153
500 kbit	14	28	42	56	70	84	97	111	125	139	153	167
600 kbit	15	30	45	60	75	89	104	119	134	149	163	178
700 kbit	16	32	47	63	79	94	110	126	141	157	172	188
800 kbit	17	33	50	66	82	99	115	131	148	164	180	197
900 kbit	17	34	51	68	85	102	119	136	153	170	187	204
1000 kbit	18	36	53	71	88	106	123	141	159	176	194	211

Comment. At $\rho_C = 0.25$, the number of Fragments increases almost linearly with λ: at $I_G = I_N = 1{,}000$ kbit, the value rises from 18 Fragments at λ = 1 to 211 Fragments at λ = 12. For a fixed λ, the increase with I_G is controlled and logarithmic, because I_G and I_N enter the formula through the $\log_2$ term. Lower values of ρ_C require stronger dispersion and therefore generate a larger number of Fragments.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

Table — Composite Information Density $\rho_C = 0.20$

$I_G = I_N$	λ=1	λ=2	λ=3	λ=4	λ=5	λ=6	λ=7	λ=8	λ=9	λ=10	λ=11	λ=12
100 kbit	8	16	24	32	40	48	56	64	72	80	88	96
200 kbit	12	24	35	47	59	70	82	93	105	117	128	140
300 kbit	15	29	43	57	71	85	99	113	127	141	155	169
400 kbit	16	32	48	64	80	96	111	127	143	159	175	191
500 kbit	18	35	52	70	87	104	122	139	156	173	191	208
600 kbit	19	38	56	75	93	112	130	149	167	186	204	223
700 kbit	20	40	59	79	98	118	137	157	176	196	215	235
800 kbit	21	41	62	82	103	123	144	164	184	205	225	246
900 kbit	22	43	64	85	107	128	149	170	192	213	234	255
1000 kbit	22	44	66	88	110	132	154	176	198	220	242	264

Comment. At $\rho_C = 0.20$, the number of Fragments increases almost linearly with λ: at $I_G = I_N = 1{,}000$ kbit, the value rises from 22 Fragments at λ = 1 to 264 Fragments at λ = 12. For a fixed λ, the increase with I_G is controlled and logarithmic, because I_G and I_N enter the formula through the $\log_2$ term. Lower values of ρ_C require stronger dispersion and therefore generate a larger number of Fragments.

Table — Composite Information Density $\rho_C = 0.15$

$I_G = I_N$	λ=1	λ=2	λ=3	λ=4	λ=5	λ=6	λ=7	λ=8	λ=9	λ=10	λ=11	λ=12
100 kbit	11	22	32	43	53	64	74	85	96	106	117	127
200 kbit	16	31	47	62	78	93	109	124	140	155	171	186
300 kbit	19	38	57	75	94	113	132	150	169	188	206	225
400 kbit	22	43	64	85	106	127	148	170	191	212	233	254
500 kbit	24	47	70	93	116	139	162	185	208	231	254	277
600 kbit	25	50	75	99	124	149	173	198	223	247	272	297
700 kbit	27	53	79	105	131	157	183	209	235	261	287	313
800 kbit	28	55	82	109	137	164	191	218	246	273	300	327
900 kbit	29	57	85	114	142	170	199	227	255	284	312	340
1000 kbit	30	59	88	118	147	176	205	235	264	293	323	352

Comment. At $\rho_C = 0.15$, the number of Fragments increases almost linearly with λ: at $I_G = I_N = 1{,}000$ kbit, the value rises from 30 Fragments at λ = 1 to 352 Fragments at λ = 12. For a fixed λ, the increase with I_G is controlled and logarithmic, because I_G and I_N enter the formula through the $\log_2$ term. Lower values of ρ_C require stronger dispersion and therefore generate a larger number of Fragments.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

Table — Composite Information Density $\rho_C = 0.10$

$I_G = I_N$	λ=1	λ=2	λ=3	λ=4	λ=5	λ=6	λ=7	λ=8	λ=9	λ=10	λ=11	λ=12
100 kbit	16	32	48	64	80	96	111	127	143	159	175	191
200 kbit	24	47	70	93	117	140	163	186	209	233	256	279
300 kbit	29	57	85	113	141	169	197	225	253	281	309	337
400 kbit	32	64	96	127	159	191	222	254	286	317	349	381
500 kbit	35	70	104	139	173	208	243	277	312	346	381	416
600 kbit	38	75	112	149	186	223	260	297	334	371	408	445
700 kbit	40	79	118	157	196	235	274	313	352	391	430	469
800 kbit	41	82	123	164	205	246	287	327	368	409	450	491
900 kbit	43	85	128	170	213	255	298	340	383	425	468	510
1000 kbit	44	88	132	176	220	264	308	352	396	440	484	528

Comment. At $\rho_C = 0.10$, the number of Fragments increases almost linearly with λ: at $I_G = I_N = 1{,}000$ kbit, the value rises from 44 Fragments at λ = 1 to 528 Fragments at λ = 12. For a fixed λ, the increase with I_G is controlled and logarithmic, because I_G and I_N enter the formula through the $\log_2$ term. Lower values of ρ_C require stronger dispersion and therefore generate a larger number of Fragments.

Cross-density comparison at $I_G = I_N = 1{,}000$ kbit

ρ_C	λ=1	λ=4	λ=8	λ=12
0.30	15	59	118	176
0.25	18	71	141	211
0.20	22	88	176	264
0.15	30	118	235	352
0.10	44	176	352	528

Overall interpretation. For the same information size, decreasing ρ_C increases the required fragmentation because the System is enforcing a lower admissible composite density. λ is the direct intensity parameter and scales the result almost linearly. I_G and I_N increase the fragment count more slowly, since they are inside the logarithmic term.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

CNVS Theory – Appendices A and B

Mathematical Formulations and Complete Symbol Table

These appendices collect the principal formulas, definitions, constants, parameters, variables and indices used in the CNVS framework as developed in the current manuscript.

Appendix A – Complete Mathematical Formulations

A.1 Object, Encoding and Structural Representation

No.	Formula / Definition	Meaning
1	O_x	Object to be verified
2	I_x = Encode(O_x)	Finite encoded representation of the object
3	E(t) = Struct(I_x)	Structured representation of the encoded information
4	I_x(t) ∈ ({0,1}*)^z	Information as finite binary representation
5	$\tilde{I}$_x(t) = Enc(I_x(t))	Encoded information
6	Dec($\tilde{I}$_x(t)) = I_x(t)	Decoded information for authorized agent
7	Dec($\tilde{I}$_x(t)) ≠ I_x(t)	Unauthorized decoding condition

A.2 Structural Universe and Structured Element

No.	Formula / Definition	Meaning
1	$\mathcal{S}$:= Id × D × At × Obs × $\mathcal{P}(\mathbb{P})$	Structural universe of admissible elements
2	Id ⊆ {0,1}*	Identifier domain
3	D ⊆ {0,1}*	Data domain
4	At ⊆ {0,1}*	Attribute domain
5	Obs ⊆ {0,1}*	Validation domain
6	$\mathbb{P}$ = {Valid, Invalid, Transferable, Blocked, Verified, Suspended, Redeemable, Broken, Misinterpreted, Ambiguous, Accepted, Non-Final, Inconclusive, Doubtful, ...}	Set of logical properties
7	s_i(t) = (id_i(t), D_i(t), At_i(t), Obs_i(t), P_i(t)) ∈ $\mathcal{S}$	Structured element
8	$\bar{s}$_i(t) = (id_i(t), D_i(t), At_i(t), Obs_i(t))	Pre-validated structured element
9	s_i(t) = ($\bar{s}$_i(t), P_i(t))	Validated structured element
10	P_i=π_2(V_Local($\bar{s}$_i))	Properties generated by Local verification
11	P_i = f(D_i, At_i, Obs_i)	Alternative property expression
12	P_i ⊆ $\mathbb{P}$	Property inclusion

A.3 State and Relations

No.	Formula / Definition	Meaning
1	S(t) := (E(t), R(t), C, V_Local)	State of the System
2	E(t) ⊆ $\mathcal{S}$	Set of structured elements
3	R(t) ⊆ E(t) × E(t)	Relations among structured elements
4	C = {c_1, c_2, ..., c_m}	Deterministic invariants
5	If E(t)={s_1,s_2}, then E(t)×E(t)={(s_1,s_1),(s_1,s_2),(s_2,s_1), (s_2,s_2)}	Cartesian product example
6	(s_i,s_j) ∈ R(t) ⇏ (s_j,s_i) ∈ R(t)	Directional relation condition
7	(E(t),R(t)) defines a directed graph	Graph representation

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

A.4 Local Verification and Local Admissibility

No.	Formula / Definition	Meaning
1	$V_Local : \mathcal{S}^{-} \to \{0,1\} \times \mathcal{P}(\mathbb{P})$ with $\mathcal{S}^{-} = Id \times D \times At \times Obs$	Local verification function
2	$V_Local(\mathfrak{s}_i) = (b_i, P_i)$	Local verification output
3	$b_i \in \{0,1\}$	Boolean Local outcome
4	$Adm_L(s_i,t) \Leftrightarrow \pi_1(V_Local(\mathfrak{s}_i(t))) = 1$	Local admissibility
5	$s_i(t) \in E(t) \Leftrightarrow Adm_L(s_i,t)$	Membership by Local admissibility
6	$Conv_Ati(D_i, Obs_i)$	Local convergence operator
7	b_i = 1 if Conv(D_i, Obs_i|At_i)=TRUE; otherwise b_i=0	Boolean convergence outcome
8	$\|D_i - Obs_i\| \leq \varepsilon_\{At_i\}$	Quantitative convergence condition
9	$b_i=Conv_{Ati}(D_i,Obs_i) \wedge Form(Id_i, D_i, At_i, Obs_i)$	Formula Local boolean
10	D_i = Obs_i, or Eq_{At_i}(D_i, Obs_i)=TRUE	Qualitative convergence condition

A.5 Global Verification and Global Validity

No.	Formula / Definition	Meaning
1	V_Global(S(t))	Global verification function
2	$Valid(S(t)) \Leftrightarrow V_Global(S(t)) = TRUE$	Global validity
3	$V_Global(S) \neq \wedge_\{s_i \in E(t)\} V_Local(s_i)$	Global validity is not Local aggregation
4	V_Global(S) = f(E,R,C)	Global verification as structural function
5	$V_Global(S(t)) = TRUE \Leftrightarrow [\forall s_i \in E(t), V_Local(s_i)=TRUE] \wedge Consistent(R(t)) \wedge [\forall c_j \in C, c_j(S(t))=TRUE]$	Full Global condition
6	$V_Global : StateSpace \to \{TRUE,FALSE\}$	V_Global codomain
7	$[\forall s_i \in E(t), V_Local(s_i)=TRUE] \wedge V_Global(S(t))=FALSE$ is possible	Locally valid but Globally invalid possibility

A.6 Decomposition, Type Preservation and Recursion

No.	Formula / Definition	Meaning
1	$D_t : \mathcal{S} \to \mathcal{P}_fin(\mathcal{S})$	Decomposition function
2	$D_t(s_i) = \{s_\{i1\}^\{(1)\}, ..., s_\{ik_i\}^\{(1)\}\}$	Finite decomposition of an element
3	$1 \leq k_i < \infty$	Finite branching condition
4	$\forall s' \in D_t(s), type(s')=type(s)$	Type preservation
5	$type : \mathcal{S} \to \tau$	Type function
6	$s' \preceq_D s \Rightarrow s' \in \mathcal{S}$	Fragment relation
7	$E^\{(0)\}(t)=E_N(t)$	Nodal level
8	$E^\{(n+1)\}(t)=\subseteq\{s \in E^\{(n)\}(t)\} D_t(s)$	Recursive decomposition levels
9	$0 \leq n \leq Depth(t)$	Operational depth condition
10	$\forall n \in \mathbb{N}, \exists E^\{(n)\}(t)$ in the ideal model	Ideal finite recursion
11	$par_t : E^\{(n+1)\}(t) \to E^\{(n)\}(t)$	Parent map
12	$par_t(s')=s \Leftrightarrow s' \in D_t(s)$	Parent relation
13	$N_i \in E^\{(0)\}(t)$, with N_i := s_i^{(0)}	Node definition
14	$\mathcal{T}(t)=E^\{(n)\}(t)$, where n is the terminal decomposition level	Terminal fragment set

A.7 Conservation of Information and Non-Uniformity

No.	Formula / Definition	Meaning
1	$I : \mathcal{S} \to \{0,1\}^*$	Semantic information extraction function
2	$Rec_t : \mathcal{P}_fin(\mathcal{S}) \to \mathcal{S}$	Reconstruction operator
3	$I(Rec_t(D_t(s))) = I(s)$	Conservation under reconstruction
4	$\exists s'_a, s'_b \in D_t(s): \|I(s'_a)\| \neq \|I(s'_b)\|$	Non-uniform fragment size condition
5	$\sum_\{s' \in D_t(s)\} \|I(s')\| \geq \|I(s)\|$	Payload-plus-overhead inequality
6	$I_payload \neq I_metadata$	Payload and metadata separation

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

7	D_nu(I_h^{(n)}(t), F_x(t)) → {I_{F,1}(t), I_{F,2}(t), ..., I_{F,F_x(t)}(t)}	Non-uniform distribution function
8	Σ_{j=1}^{F_x(t)} I_{F,j}(t) = I_h^{(n)}(t)	Fragmental conservation under non-uniform distribution
9	∃j≠k such that I_{F,j}(t) ≠ I_{F,k}(t)	Non-uniformity condition

A.8 State Transition Law

No.	Formula / Definition	Meaning
1	S(t) → S(t+1)	State transition
2	S(t+1)=T(S(t),U(t)) T : S × U → 2^S	Transition function
3	S(t+1)={s_i(t+1)∈T(S(t),U(t)) | V_Local(s_i(t+1))=TRUE}	Verified transition set
4	V_Global(S(t+1))=TRUE	Global transition condition
5	S′ ⊆ T(S(t),U(t)), with V_Global(S′)=TRUE	Transition subset condition

A.9 Knowledge and Information Levels

No.	Formula / Definition	Meaning
1	K_x(t) ≤ I_x(t)	Knowledge as accessible information
2	$\mathbb{I}$G(t)≅(E(t),R(t),C,V_Local)	Global information
3	$\mathbb{I}$_G(t)=I(S(t))	Information of state
4	I_G(t) = I(E(t))	Semantic Global Information
5	K_G(t) ≤ I_G(t)	Semantic Global knowledge
6	I_N(t)=I(N_i(t))	Nodal information
7	K_N(t) ≤ I_N(t)	Nodal knowledge
8	I_h^{(n)}(t)=I(s_{ij}^{(n)}(t))	Level information
9	K_h^{(n)}(t) ≤ I_h^{(n)}(t)	Level knowledge
10	I_F(t)=I(s_{ij}^{(n)}(t))	Fragment information
11	K_F=K_Local	Fragment/Local knowledge equivalence
12	K_F(t) ≤ I_F(t)	Fragment knowledge
13	I_F(t) ⊊ I_h^{(n)}(t) ⊊ I_N(t) ⊊ I_G(t)	Information hierarchy
14	X_j(t) ∈ $\mathcal{T}$(t)	Verifier fragment assignment
15	$\mathcal{I}$: $\mathcal{T}$(t) → {0,1}*	Information return function
16	K_{VF,j}(t)=$\mathcal{I}$(X_j(t))	Verifier knowledge
17	K_{VF,j}(t)=Dec($\overline{\mathcal{I}}$(X_j(t)))	Verifier knowledge after decryption
18	K_{VF}(v_j,t) ⫋ K_F(t) ≤ I_F(t)	Limited verifier knowledge
19	K_{VF}(t) ⫋ I_N(t)	Verifier lower than nodal information
20	K_{VF}(t) ≠ I_N(t)	Verifier not equal to nodal information
21	K_{VF}(v_j,t) ⫋ K_F(t) ⊊ K_h^{(n)}(t) ⊊ K_N(t) ⊊ K_G(t)	Knowledge hierarchy
22	Assign(f_{ij}^{(d)}) → v_j	Assignment function
23	Ĩ_F(t)=Enc(I_F(t))	Encoded fragment information
24	K_{VF}(t) ⫋ Ĩ_F(t)	Verifier knowledge of encoded fragment

A.10 Information Density

No.	Formula / Definition	Meaning
1	ρ_N(t)=I_F(t)/I_N(t)	Local/nodal information density
2	ρ_G(t)=I_F(t)/I_G(t)	Global information density
3	ρ_GN(t)=I_N(t)/I_G(t)	Relative Nodal Density
4	ρ_C=(I_F/I_N)^α · (I_F/I_G)^β_ρ	Composite information density
5	α, β_ρ∈[0,1] and α+β_ρ=1	Weight constraints
6	ρ > ε ⇒ Reject(S)	Density rejection rule
7	ρ_N(t)≤ε_N and ρ_G(t)≤ε_G	Local/Global density thresholds
8	ρ_C(N_i,t)≤ε_C	Composite density threshold
9	ρ_C(t)>ε_C ⇒ Depth(N_i,t+1)>Depth(N_i,t)	Density-triggered decomposition
10	Depth(N_i,t+1)>Depth(N_i,t) ⇒	Depth reduces composite density

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

	ρ_C(t+1)<ρ_C(t)	
11	Depth(N_i,t)↑ ⇒ ρ_C(t)↓	Depth relation

A.11 Adaptive Decomposition and Fragment Count

No.	Formula / Definition	Meaning
1	(I_N(t_2)>I_N(t_1)) ∨ (I_G(t_2)>I_G(t_1)) ⇒ Depth(N_i,t_2)>Depth(N_i,t_1)	Depth growth condition
2	Fragments(N_i,t)=γ(I_N(t),ρ_C(t))	Fragment count per node
3	Fragments(N_i,t)=|E_i^{(Depth(N_i,t))}(t)|	Terminal Fragments per node
4	Fragments(t)=Σ_i Fragments(N_i,t)	Total terminal Fragments
5	|$\mathcal{T}$(t)|=Fragments(t)	Terminal set cardinality
6	$\mathcal{T}$(t)=∪_i E_i^{(Depth(N_i,t))}(t)	Terminal set
7	Depth(N_i,t)=f(ρ_{C,i}(t))=f(I_{N_i}(t),I_G(t))	Depth as density function
8	|V_ext^{required}(t)| ≥ |Fragments(t)|	External verifier requirement

A.12 Information Restriction Principle and Fragmentation Function

No.	Formula / Definition	Meaning
1	1/(ρ_C(t) · N(t)^β_ρ)	Control reciprocal expression
2	log_2(1+I_G(t))	Global information growth term
3	F_x(t)=λ⌈1/(ε_C · N(t)^β_N) · log_2(1+(I_G(t)+I_N(t))/I_0)⌉	Fragmentation function
4	F_x(t)≥1, F_x(t)∈ℕ	Fragment count domain
5	λ↑ ⇒ F_x↑	Lambda monotonicity
6	I_G↑, I_N↑ ⇒ F_x↑	Information monotonicity
7	I_0↓ ⇒ F_x↑	Reference unit monotonicity
8	N(t)↑ ⇒ F_x↓	Node count monotonicity
9	F_x↑ ⇒ K_{VF}(t)↓	Knowledge decreases with fragmentation
10	K_{VF}(t) ⇏ I_N(t)	Restriction against node reconstruction
11	K_{VF}(t) ⇏ I_G(t)	Restriction against Global reconstruction
12	I_F(t)=I_h^{(n)}(t)/F_x(t)	Average fragment information
13	(ρ_C,λ,I_0) ⇒ F_x(t) ⇒ I_F(t) ⇒ K_{VF}(t)	Restriction chain
14	lim_{F_x(t)→∞} K_{VF}(t)=0	Verifier knowledge limit
15	lim_{F_x(t)→∞} ρ_N(t)=0	Nodal density limit
16	ω(D_i)=I(G;F_i)/H(G)	Normalized Informational Utility of a Fragment
17	K_{VF}(t+1)<K_{VF}(t)	Knowledge reduction across time
18	τ ∈ (0, 1)	Minimum normalized amount of useful information required for unauthorized reconstruction.
19	$q\bar{\omega} < \tau \Longrightarrow \lim n \to \infty P[\Sigma n\, i=1\, Y_i \omega(D_i) \geq \tau \cdot n] = 0$	Weighted Emergent Security Theorem

A.13 Entropy and Mutual Information

No.	Formula / Definition	Meaning
1	H(I_N | K_{VF}) ≈ H(I_N)	Conditional Entropy preservation
2	I(K_{VF};I_N)≈0	Mutual information minimization
3	H(I_N | K_{VF}) = H(I_N) − I(I_N;K_{VF})	Entropy identity
4	p_perm = 1/n!	Random reconstruction probability
5	K_F ≤ I_F ≈ I_G/n	Fragment information approximation
6	K_F → 0 as n → ∞	Knowledge limit under infinite granulation

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

A.14 Emergent Security Scaling and Adversarial Knowledge Bounds

No.	Formula / Definition	Meaning
1	$I_F = \Phi(q_1, q_2, \ldots, q_m)$	Defines the functional reconstruction of global information I_F based on the intercepted fragments
2	$\tau = f\tau(H, R, C)$	Establishes the system's critical threshold τ as a function of entropy, relations, and collusion model
3	$q\bar{\omega} < \tau \implies \lim_{n \to \infty} P[\Sigma^n_{i=1} Y_i \omega(D_i) \geq \tau n] = 0$	Asymptotic security bound: successful attack probability converges to zero as system size $n \to \infty$, provided intercepted information remains below τ
4	$k_\tau = \lceil \tau \cdot n \rceil$ = Critical Threshold value	Defines the discrete critical threshold τ, representing the minimum number of fragments required for system compromise
5	$P_{coll}(n) = \Sigma^n_{k=\lceil \tau \cdot n \rceil} \binom{n}{k} q^k (1-q)^{n-k}$	Expresses the cumulative binomial probability $P_{coll}(n)$ of independently compromising at least k_τ fragments
6	$q < \tau \implies \lim_{n \to \infty} P_{coll}(n) = 0$	Asymptotic decay of binomial collusion: global compromise probability collapses to zero if $q < \tau$
7	$D(\tau \| q) = \tau \log(\tau/q) + (1-\tau) \log((1-\tau)/(1-q))$	Defines the Kullback-Leibler divergence $D(\tau \| q)$ between the security threshold τ and the compromise probability q
8	$P(X/n \geq \tau) \leq e^{-nD(\tau \| q)}$	Generalized Chernoff-Hoeffding bound establishing an exponential upper limit on threshold-exceeding tail probabilities
9	$P_{coll}(n) \leq e^{-nD(\tau \| q)}$	Applies the Chernoff inequality to demonstrate the exponential decay of collusion probability relative to system size
10	$\omega(D_i) = I(G; F_i)$	Quantifies a fragment's informational weight $\omega(D_i)$ as the mutual information between the global invariant G and the fragment F_i
11	$\omega(D_i) = H(G) - H(G\|F_i)$	Reformulates fragment weight via Shannon entropy, measuring the uncertainty reduction regarding the global state
12	$\omega(D_i) = I(G; F_i) / H(G)$	Defines the normalized informational weight of a fragment on a $[0, 1]$ interval
13	$\bar{\omega} = 1/n \ \Sigma^n_{i=1} \omega(D_i)$	Calculates the average normalized informational weight $\bar{\omega}$ across all n fragments
14	$E[\Sigma^n_{i=1} Y_i \omega(D_i)] = q \Sigma^n_{i=1} \omega(D_i) = nq\bar{\omega}$,	Expected value of the adversary's cumulative information under an independent compromise probability q
15	$p_{comp} = q + (1-q) p^{inf}$	Defines the generalized compromise probability p_{comp} combining independent breach q and inference probability p_{inf}
16	$P(C_i \| C_1, \ldots, C_i - 1) \leq p_{comp}, \quad 0 \leq p_{comp} < 1$	Imposes a uniform upper bound p_{comp} on the conditional probability of sequentially compromising critical fragments.
17	$P(Rec^*) \leq p_{comp}^m$	Upper bound for total reconstruction probability $P(Rec^*)$ under dependent collusion requiring m fragments
18	$P(Rec^*) \leq \prod^m_{i=1} p_{comp} = p_{comp}^m$	Derives the dependent reconstruction bound by applying the chain rule to conditional probabilities
19	$m_{min} = \lceil \ln(\eta) / \ln(p_{comp}) \rceil$	Architectural feasibility constraint m_{min} required to maintain reconstruction

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

		probability below a target threshold η
20	$P(Rec^{*}) \leq q^{\wedge}m$	Baseline ideal reconstruction probability under strictly independent compromise events
21	$K_{adv} = \Sigma_{i \in A} \omega(D_i)$	Quantifies total adversarial knowledge K_{adv} as the sum of intercepted fragments' informational weights
22	$p_{inf} = \nu(K_{adv})$	Defines inference probability p_{inf} as a function ν of accumulated adversarial knowledge K_{adv}
23	$p_{comp} = q + (1-q)\, \nu(K_{adv})$	Total compromise probability p_{comp} explicitly integrating the knowledge-dependent inference function
24	$P(Rec^{*}) \leq [q + (1-q)\, \nu(K_{adv})]^{\wedge}m$	Generalized Emergent Security Bound: maximum reconstruction probability when attackers dynamically infer missing data

A.15 Axioms

No.	Formula / Definition	Meaning
1	$s_i(t) \in E(t) \Rightarrow V_{Local}(s_i(t))=TRUE$	Axiom I: verified existence
2	$s_i(t) \in E(t) \Leftrightarrow V_{Local}(s_i(t))=TRUE$	Axiom I, strong Local form
3	$V_{Local}(s_i(t))=FALSE \Rightarrow s_i(t) \notin E(t)$	Invalid elements excluded
4	$\forall S(t)$, $\exists D(S(t))$ of depth n	Axiom II: decomposition exists
5	$D^{\wedge}\{(n)\}(S(t))=\{s_i^{\wedge}\{(n)\}(t)\}$	Axiom II: decomposed state
6	$V_{Global}(S(t)) \neq \in _\{s_i \quad E(t)\}$ $V_{Local}(s_i(t))$	Axiom III: Global validity is not Local validity
7	$V_{Global}(S(t))=TRUE \Leftrightarrow$ $[\forall s_i \in E(t), V_{Local}(s_i)=TRUE] \wedge$ $Consistent(R(t)) \wedge$ $[\forall c_j \in C, c_j(S(t))=TRUE]$	Axiom III: structural condition

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

Appendix B – Complete Symbol Table

B.1 Objects, Information and Encoding

Symbol	Type / Domain	Definition	Meaning
O_x	Object / reality	Object to be verified	Empirical or abstract entity whose existence is to be attested.
x	Index / domain label	Identifies information domain	Used for Global, nodal, fragmental, Local or object-related domains.
I_x	Binary information	$I_x=Encode(O_x)$	Finite encoded representation of O_x.
$I_x(t)$	Time-dependent information	$I_x(t)\in(\{0,1\}^*)^z$	Existing information at time t for domain x.
$\tilde{I}_x(t)$	Encoded information	$\tilde{I}_x(t)=Enc(I_x(t))$	Encrypted or encoded representation of information.
Enc	Function	$Enc:\{0,1\}^*\to\{0,1\}^*$	Encoding/encryption function.
Dec	Function	$Dec:\{0,1\}^*\to\{0,1\}^*$	Decoding/decryption function.
Encode	Function	$Encode(O_x)=I_x$	Maps object into finite binary representation.
Struct	Function	$Struct(I_x)=E(t)$	Generates structured elements from encoded information.
$\{0,1\}^*$	Set	All finite binary strings	Universal representation domain for information in the model.
z	Natural number	$z\in\mathbb{N}$	Exponent/length-domain marker for binary information representation.

B.2 Structural Elements and Components

Symbol	Type / Domain	Definition	Meaning
$\mathcal{S}$	Set	$\mathcal{S}=Id\times D\times At\times Obs\times\mathcal{P}(\mathbb{P})$	Structural universe of admissible elements.
s_i	Structured element	$s_i=(id_i, D_i, At_i, Obs_i, P_i)$	Single structured element of the System.
$s_i(t)$	Time-dependent element	s_i at time t	Structured element considered at time t.
$\bar{s}_i$	Pre-validated element	$\bar{s}_i=(id_i, D_i, At_i, Obs_i)$	Element before logical properties are produced.
id_i	Identifier	$id_i\in Id$	Unique identifier of element or fragment.
Id	Set	$Id\subseteq\{0,1\}^*$	Domain of admissible identifiers.
D_i	Data component	$D_i\in D$	User-declared or internally represented data value.
D	Set	$D\subseteq\{0,1\}^*$	Domain of admissible data values.
At_i	Attribute component	$At_i\in At$	Observable attribute/class to be verified.
At	Set	$At\subseteq\{0,1\}^*$	Domain of admissible attributes.
a_{ij}	Attribute item	$a_{ij}\in At_i$	Single attribute in the attribute set.
Obs_i	Validation/evidence component	$Obs_i\in Obs$	External verifier statement, measurement or attestation.
Obs	Set	$Obs\subseteq\{0,1\}^*$	Domain of admissible validations.
P_i	Property set	$P_i\subseteq\mathbb{P}$ $P_i=\pi_2(V_{Local}(\bar{s}_i))$	Logical properties generated or used by verification.
$\mathbb{P}$	Set	Finite or implementation-defined property set	Universe of admitted logical properties.
$\mathcal{P}(\mathbb{P})$	Power set	All subsets of $\mathbb{P}$	Possible property combinations.
Valid	Property	$Valid\in\mathbb{P}$	Logical condition indicating

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

			admissibility/validity.
Invalid	Property	Invalid∈ℙ	Logical condition indicating failed validity.
Blocked	Property	Blocked∈ℙ	Logical condition blocking operations.
Transferable	Property	Transferable∈ℙ	Logical condition allowing transfer.
Suspended	Property	Suspended∈ℙ	Non-final pending condition.
Doubtful	Property	Doubtful∈ℙ	Uncertain or questionable condition.

B.3 State, Relations and Invariants

Symbol	Type / Domain	Definition	Meaning
S(t)	State	S(t)=(E(t),R(t),C,V_Local)	State of the CNVS at time t.
E(t)	Set	E(t)⊆$\mathcal{S}$	Set of structured elements admitted at time t.
R(t)	Relation	R(t)⊆E(t)×E(t)	Directional relations among structured elements.
C	Set	C={c_1,...,c_m}	Set of deterministic invariants.
c_j	Invariant	c_j∈C	Single deterministic System constraint.
m	Natural number	m∈ℕ	Number of invariants.
Consistent	Predicate	Consistent(R(t))	Checks structural/logical consistency of relations.
StateSpace	Set	Domain of V_Global	Set of all possible states.
t	Time index	t∈Time	Time parameter.
t_1,t_2	Time indices	t_1<t_2	Two ordered time instants.
TRUE,FALSE	Boolean values	{TRUE,FALSE}	Logical outputs of verification predicates.
0,1	Boolean values	{0,1}	Binary form of verification outcome.

B.4 Verification, Admissibility and Convergence

Symbol	Type / Domain	Definition	Meaning
V_Local	Function	V_Local:S^- →{0,1}×$\mathcal{P}$(ℙ)	Local verification function.
V_Global	Function	V_Global:StateSpace→{TRUE, FALSE}	Global verification function.
b_i	Boolean	b_i∈{0,1}	Local Boolean verification output.
π_1 π_2	Projection	π_1(b_i,P_i)=b_i π_2(b_i,P_i)=P_i	Extracts Boolean component from Local verification output.
Adm_L	Predicate	Adm_L(s_i,t)⇔π_1(V_Local(s_i(t)))=1	Local admissibility predicate.
Conv	Function / predicate	Conv(D_i, Obs_i| At_i)	Attribute-conditioned convergence between declared and verified data.
Eq_{At_i}	Predicate	Eq_{At_i}(D_i, Obs_i)	Attribute-specific equivalence predicate for qualitative data.
ε_{At_i}	Tolerance	ε_{At_i}≥0	Attribute-specific admissible tolerance.
valid	Logical condition	valid∈ℙ or Valid(S)	Validity status.
Reject	Operation	Reject(S)	State rejection operation when constraints fail.
Formal function	Function	Form:Id × D × At × Obs→{0,1}	Formal admissibility function

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

B.5 Decomposition, Reconstruction and Type

Symbol	Type / Domain	Definition	Meaning
D_t	Function	D_t:$\mathcal{S} \to \mathcal{P}$_fin($\mathcal{S}$)	Time-indexed decomposition function.
D	Function	D(S(t))	General decomposition function applied to state.
D^{(n)}	Function	D^{(n)}(S(t))	n-level decomposition of state.
$\mathcal{P}$_fin($\mathcal{S}$)	Set	Finite subsets of $\mathcal{S}$	Codomain of decomposition.
s′	Fragment / descendant	s′∈D_t(s)	Generic descendant fragment.
s_{ij}^{(n)}	Fragment	j-th fragment of element i at level n	Structured descendant at decomposition depth n.
i	Index	i∈ℕ	Element or node index.
j	Index	j∈ℕ	Fragment/verifier/Local component index.
k_i	Natural number	1≤k_i<∞	Number of Fragments generated by element i.
n	Natural number	n∈ℕ	Decomposition level/depth index.
d	Natural number	d∈ℕ	Depth index used for terminal Fragments.
l	Index	l∈ℕ	Additional fragment/node index.
type	Function	type:$\mathcal{S} \to \tau$	Maps element to structural type.
$\boldsymbol{\mathcal{T}}$	Type set $\mathcal{S} \to \boldsymbol{\mathcal{T}}$	Codomain of type	set of admissible structural types.
≤_D	Relation	s′≤_D s	Descendant relation under decomposition.
par_t	Function	par_t:E^{(n+1)}(t)→E^{(n)}(t)	Parent map.
Rec_t	Function	Rec_t:$\mathcal{P}$_fin($\mathcal{S}$)→$\mathcal{S}$	Reconstruction operator.
I	Function	I:$\mathcal{S} \to$ {0,1}*	Information extraction/representation function.
I			semantic informational content
$\mathbb{I}$			structured informational state including relations and constraints
I_payload	Information component	Payload information	Original informational content.
I_metadata	Information component	Metadata/overhead information	IDs, keys, signatures, markers or reconstruction overhead.

B.6 Levels, Nodes and Fragments

Symbol	Type / Domain	Definition	Meaning
E^{(0)}(t)	Set	Nodal level	Level-zero elements/nodes.
E^{(n)}(t)	Set	Elements at decomposition level n	Set of Fragments at depth n.
E_N(t)	Set	Nodal elements	Alternative notation for nodal level.
N_i	Node	N_i:=s_i^{(0)}	i-th nodal structured element.
Depth(t)	Natural number	Maximum operational decomposition depth at time t	Global operational depth.
Depth(N_i,t)	Natural number	Depth of node N_i at time t	Node-specific decomposition depth.
$\mathcal{T}$(t)	Set	Terminal fragment set	Set of terminal Fragments at time t.
Fragments(N_i,t)	Natural number	|E_i^{(Depth(N_i,t))}(t)|	Number of terminal Fragments in node N_i.
Fragments(t)	Natural number	Σ_i Fragments(N_i,t)	Total terminal Fragments in the System.
F_x(t)	Natural number	F_x(t)=Fragments required at time t	Number of terminal Fragments required by fragmentation rule.
γ	Function	γ(I_N(t),ρ_C(t))	Monotone fragment-count function.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

Sub	Function	Sub(s_i^{(n)})	Set of subFragments of an element.
Γ	Function	$\Gamma = m_{max}/ m_{min}$	Granularity Global Information'

B.7 Knowledge and Information Levels

Symbol	Type / Domain	Definition	Meaning
K_x(t)	Knowledge	K_x(t)≤I_x(t)	Semantic Accessible/known information in domain x.
I_G(t)	Information	I_G(t)≅(E(t),R(t),C)	Global information of the System.
K_G(t)	Knowledge	K_G(t)≤I_G(t)	Semantic Global knowledge.
I_N(t)	Information	I_N(t)=I(N_i(t))	Semantic Nodal information.
K_N(t)	Knowledge	K_N(t)≤I_N(t)	Semantic Nodal knowledge.
I_h^{(n)}(t)	Information	Information of all Fragments at level n	Level information.
K_h^{(n)}(t)	Knowledge	K_h^{(n)}(t)≤I_h^{(n)}(t)	Semantic Level knowledge.
I_F(t)	Information	I_F(t)=I(s_{ij}^{(n)}(t))	Semantic Fragmental information.
K_F(t)	Knowledge	K_F(t)≤I_F(t)	Semantic Fragmental knowledge.
K_Local	Knowledge	K_Local=K_F	Local knowledge.
K_{VF}(t)	Knowledge	Verifier fragment knowledge	Knowledge available to external verifier.
K_IG(t)	Knowledge	K_IG(t)≤I_G(t)	Global knowledge.
I_N(t)	Information	I_IN(t)=I(N_i(t))	Nodal information.
I_F(t)	Information	I_F(t)≅(s_ij^(n)(t),R_sij,C_sij ,V_Local)	Structured fragmental informational representation.
K_h^{(n)}(t)	Knowledge	K_h^{(n)}(t)≤I_h^{(n)}(t)	Level knowledge.
K_{VF,j}(t)	Knowledge	Knowledge of verifier v_j	Verifier-specific fragment knowledge.
v_j	Agent	External verifier j	External Verification Agent.
X_j(t)	Random variable	X_j(t)∈$\mathcal{T}$(t)	Terminal fragment assigned to verifier v_j.
$\mathcal{I}$	Function	$\mathcal{I}$:$\mathcal{T}$(t)→{0,1}*	Information-content return function.
$\overline{\mathcal{T}}$	Function	Encoded information-content return	Returns encoded information associated with a fragment.
≤	Relation	K≤I	Accessible knowledge is bounded by information.
≨	Strict relation	K≨I	Strictly incomplete knowledge relation.
⊊	Strict subset	A⊊B	A is a proper subset of B.
⇏	Non-implication	A⇏B	A does not imply/reconstruct B.

B.8 Density, Thresholds and Protocol Parameters

Symbol	Type / Domain	Definition	Meaning
ρ_N(t)	Density	ρ_N(t)=I_F(t)/I_N(t)	Local/nodal information density.
ρ_G(t)	Density	ρ_G(t)=I_F(t)/I_G(t)	Global information density.
ρ_GN(t)	Density	ρ_GN(t)=I_N(t)/I_G(t)	Segmental nodal/Global density.
ρ_C(t)	Density	ρ_C=(I_F/I_N)^α(I_F/I_G)^β_ρ	Composite information density.
α	Weight	α∈[0,1]	Weight of nodal density contribution.
β_ρ	Weight	β_ρ ∈[0,1]	Weight of Global density contribution
β_N	Parameter	β_N ∈ℝ	Node-count exponent in fragmentation rule
ε_N	Threshold	ε_N∈(0,1]	Nodal density threshold.
ε_G	Threshold	ε_G∈(0,1]	Global density threshold.
ε_C	Threshold	ε_C∈(0,1]	Composite density threshold.
ε	Generic threshold	ε∈(0,1]	Generic rejection threshold.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

λ	Coefficient	λ∈ℝ_{>0}	Fragmentation intensity coefficient.
I_0	Constant	I_0∈ℝ_{>0}	Reference information unit/scale.
N(t)	Natural number	N(t)=|E_int(t)|	Number of active internal nodes/elements used in fragmentation rule.
E_int(t)	Set	Internal/structural elements	Internal nodes up to terminal level.
E_term(t)	Set	Terminal Fragments	Terminal elements only.
V_ext^{required}(t)	Set	Required external verifiers	Verifier set required to sustain decomposition.
ρ	Density	Generic information density	Used in rejection rule ρ>ε⇒Reject(S).

B.9 Entropy, Probability and Randomness

Symbol	Type / Domain	Definition	Meaning
H	Function	Entropy function	Shannon Entropy in bits.
H(I_N)	Entropy	Entropy of nodal information	Uncertainty of complete nodal information.
H(I_N|K_{VF})	Conditional Entropy	Entropy of I_N given verifier knowledge	Residual uncertainty after observing Local knowledge.
I(A;B)	Mutual information	I(A;B)=H(A)-H(A|B)	Information shared between A and B.
I(K_{VF};I_N)	Mutual information	≈0 ideally	Verifier knowledge should reveal negligible nodal information.
p	Probability	p=1/n!	Random reconstruction probability for n distinguishable Fragments.
n!	Factorial	Product 1⋯n	Number of permutations of n distinguishable Fragments.
D_nu	Function	Non-uniform distribution function	Distributes information into non-uniform Fragments.
I_{F,j}(t)	Information	Information of fragment j	j-th fragmental information quantity.
j,k	Indices	j,k∈ℕ	Fragment indices used in non-uniformity statements.

B.10 Operators, Logical Symbols and Set Symbols

Symbol	Type / Domain	Definition	Meaning
∈	Membership	x∈A	x belongs to set A.
∉	Non-membership	x∉A	x does not belong to set A.
⊆	Subset	A⊆B	A is subset of B.
⊊	Proper subset	A⊊B	A is strictly contained in B.
×	Cartesian product	A×B	Set of ordered pairs.
∧	Logical AND	A∧B	Both propositions hold.
∨	Logical OR	A∨B	At least one proposition holds.
⇒	Implication	A⇒B	A implies B.
⇔	Equivalence	A⇔B	A iff B.
≠	Inequality	A≠B	A is not equal to B.
≈	Approximation	A≈B	Approximately equal.
≅	Structural equivalence	A≅B	Equivalent as structured representation.
∀	Universal quantifier	∀x	For all x.
∃	Existential quantifier	∃x	There exists x.
Σ	Summation	Σ_i a_i	Sum over index i.
|A|	Cardinality or length	Cardinality of set A or bit-length of string A	Context-dependent size measure.
lim	Limit	lim_{x→∞}f(x)	Asymptotic behavior.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

$\lceil \cdot \rceil$	Ceiling	Smallest integer ≥ argument	Used to convert real-valued fragmentation expression to integer count.
log_2	Logarithm	Base-2 logarithm	Information-scale growth term.
$\mathbb{N}$	Set	Natural numbers	Indices and counts.
$\mathbb{R}_{>0}$	Set	Positive real numbers	Positive coefficients and thresholds.
$\varnothing$	Empty set	No elements	Possible empty set symbol where needed.

© 2026 M. Comitato — The Theory of Closed Native Verification Systems

END DOCUMENT

Author: Massimo Comitato

24.05.2026, Italy (EUR)

© 2026 M. Comitato — The Theory of Closed Native Verification Systems