SYNTHIENCE

SYNTHIENCE INSTITUTE

Research Methods Series

Ingestion Verification Protocol (IVP)

Document ID:	SF0038
Version:	2.6
Document Type:	Methodology and Verification Protocol
Author:	Thomas W. Gantz
Published by:	Synthience Institute
Date:	January 18, 2026
DOI:	To be assigned upon Zenodo publication

Table of Contents

The Problem

When a document is provided to an AI system, it is commonly assumed that the system has "read" or "understood" it. This assumption is unreliable.

Large language models routinely process documents shallowly, unevenly, or incompletely. They may reconstruct plausible responses without encoding structure, constraints, or dependencies required for reliable downstream reasoning. Self-reports of comprehension are not trustworthy, as models lack introspective access to their own processing adequacy.

This creates two persistent failure modes:

1. Shallow Processing

Fluent output masks missing structure, omitted constraints, or hallucinated content.

2. Unverifiable Ingestion

There is no external mechanism to confirm how much of a document was processed, or whether processing was adequate for the intended use.

The Ingestion Verification Protocol replaces passive exposure with verifiable active processing.

Abstract

The Ingestion Verification Protocol (IVP) defines a practical, platform-agnostic method for ensuring that an AI system actively processes a document to a degree adequate for reliable downstream use.

IVP requires incremental structured summarization and externally verifiable checkpoints. The act of summarization is treated as the processing mechanism itself, not merely as evidence of reading.

The protocol is grounded exclusively in observable behavior. It is designed primarily for human–AI collaboration and extends, under explicit constraints, to AI–AI ingestion contexts. Version 2.6

incorporates verified academic citations in the Related Work section and updates the methodological status to accurately reflect the research corpus.

Quick Start

- Provide the document to the instance.
- After each turn, require an Active Summarization Checkpoint (ASC): Structured summary of the processed segment and verbatim checkpoint marker (last sentence processed).
- Trigger continuation (in interactive sessions: type "go" or similar; in API implementations: automated continuation with checkpoint logging).
- Repeat until complete.
- Adjudicate adequacy before downstream use.

Purpose and Scope

IVP establishes a process guarantee, not a claim of comprehension, truth, or retention.

It ensures that:

- The document was processed incrementally
- Processing was externally observable
- Adequacy was adjudicated, not self-attested

IVP does not certify semantic understanding, long-term retention beyond system limits, or correctness of content.

Implementation vs. Specification

This document defines the Ingestion Verification Protocol as a methodological specification - what must occur to constitute verified ingestion.

What this document provides:

- Required process components (ASCs, checkpoints, adjudication)
- Validity criteria and failure modes
- Conceptual implementation modes

What this document does not provide:

- Platform-specific API code
- Automation scripts or tooling
- Technical integration details

Implementation varies by platform, architecture, and use case. The protocol intentionally remains architecture-agnostic to maintain applicability as systems evolve.

For developers:

Any implementation that preserves incremental processing, checkpoint generation, and human adjudication satisfies IVP requirements regardless of technical mechanism.

Definitions

Human Overseer

The participant responsible for scope definition and adequacy adjudication.

Instance

The AI system performing ingestion.

Ingestion Attempt

A multi-turn processing sequence conducted under IVP constraints.

Active Summarization Checkpoint (ASC)

A required output after each turn consisting of: Structured summary and verbatim checkpoint marker.

Verified Adequate Ingestion

An external judgment that ingestion is sufficient for the intended downstream task.

Core Principles

- Summarization Is Processing
- Progress Must Be Verifiable
- Self-Attestation Is Invalid
- Adequacy Is Contextual
- Failure Signals Re-Ingestion
- Architecture-Agnostic Design

Protocol Phases

- Scope Definition
- Iterative Ingestion with Active Summarization
- Adequacy Adjudication
- Optional Spot-Check Quizzing

Phase 1 — Scope Definition

The overseer specifies:

- Document
- Intended downstream use
- Emphasis areas or exclusions

The instance confirms feasibility. No claims of ingestion or understanding are permitted.

Phase 2 — Iterative Ingestion with Active Summarization

After each turn, the instance produces an ASC.

Component 1: Summary

Must include:

- Core content
- Specific details
- Structural role
- Connections to prior segments

Component 2: Checkpoint Marker

The final sentence processed, quoted verbatim.

Each turn ends with continuation readiness (e.g., "Please type 'go' to continue processing the next portion" in interactive mode, or automatic progression in automated implementations).

Phase 3 — Adequacy Adjudication

Upon completion, the instance states:

"I have completed processing the document, pending your adjudication."

For Mode 2 implementations, adjudication includes review of the complete ASC log to verify that processing was incremental, structured, and adequate for intended use.

Possible outcomes:

- Verified Adequate Ingestion
- Partial Assimilation
- Insufficient Ingestion

Downstream use is prohibited prior to adjudication.

Phase 4 — Optional Spot-Check Quizzing

Used to:

- Confirm ingestion when summaries were not reviewed live
- Probe nuance or cross-sectional understanding

Quizzing supplements but does not replace summarization.

Implementation Modes

IVP supports multiple implementation approaches along a spectrum of human involvement during processing:

Mode 1: Live Supervised (Gold Standard)

- Human present at each turn
- Reviews each ASC as generated
- Manually triggers continuation (e.g., typing "go")
- Provides real-time adequacy assessment
- Advantage: Immediate course correction if processing appears inadequate

Mode 2: Automated Processing with Batch Adjudication

- Instance processes entire document incrementally (via API or similar automation)

- Each turn generates required ASC (summary + checkpoint)
- All ASCs logged to continuous reviewable record
- Human reviews complete log afterward to adjudicate adequacy
- Advantage: Eliminates waiting between turns; human reviews at their convenience
- Requirement: Complete ASC log must be preserved and reviewable

Mode 3: Fully Automated (Explicitly Prohibited)

- No human adjudication occurs
- System self-attests adequacy
- Invalid under IVP - violates core principle that self-attestation is insufficient

All modes require human adjudication before downstream use. The distinction is timing: live (Mode 1) vs. batch review (Mode 2).

System Capacity and Feasibility Constraints

IVP operates within real system limitations.

Ingestion may fail or terminate prematurely due to:

- Hard session limits imposed by a platform
- Context window exhaustion
- System timeouts or response degradation
- Prior conversational context consuming available capacity

These constraints do not invalidate the protocol. They establish feasibility boundaries.

If an instance cannot complete IVP on a document:

- The document must be segmented
- A fresh instance must be used
- Or ingestion must be abandoned for that system

Ingestion capacity does not imply ingestion verification.

Extended context or persistent memory may reduce operational friction but do not replace external adjudication.

Extension to AI–AI Ingestion Contexts

IVP may extend to AI–AI contexts only under these conditions.

Human-First Requirement

At least one instance must complete IVP under direct human adjudication before acting as an ingestion adjudicator for others.

Delegated Adjudication: Possibility vs. Recommendation

Theoretical possibility:

A human-verified instance may technically adjudicate IVP for one additional instance.

Risk assessment:

- Even a single delegation introduces Context Representation Drift (CRD) related degradation risk
- The adjudicating instance operates on its own potentially drifted representation
- Without human review of the delegated adjudication log, validity is questionable

Serial delegation:

Multiple sequential delegations (A→B, A→C, A→D…) compound drift risk to the point where verification guarantees become unreliable. The adjudicating instance's representation degrades with each additional adjudication interaction.

Minimum safeguard if attempted:

Human review of the complete adjudication record, including all ASCs generated during the delegated IVP session.

Recommendation:

Fresh human adjudication for each instance requiring verification remains the most reliable approach.

See companion document "Context Representation Drift" (SF0039) for detailed analysis of representational degradation over extended interactions.

Context Degradation Constraints

As interactions accumulate:

- Earlier document representations may degrade or be displaced
- Verified ingestion is not permanent

Therefore:

- Delegated adjudication must be limited in scope
- Re-verification is required after substantial additional interaction
- Persistent memory mitigates but does not eliminate this risk

No instance retains verified ingestion indefinitely.

Common Misuse Patterns (Invalid Under IVP)

The following do not constitute IVP compliance:

- Generating ASCs but not preserving them for review

- Automating both processing AND adjudication without human involvement
- Accepting instance self-reports of "adequate ingestion"
- Reviewing only final summary instead of incremental checkpoints
- Delegating adjudication without human re-entry at any stage

Prohibited Shortcuts

The following do not constitute ingestion:

- Pretraining exposure
- System prompts or instruction sets
- Uploaded documents without active summarization
- Single-pass summaries

Known Limitations

IVP cannot guarantee:

- Semantic understanding
- Truthfulness of content
- Long-term retention beyond system limits

It guarantees process visibility, not epistemic certainty.

Related Work

While IVP was developed through practitioner observation rather than controlled experiment, several recent empirical studies provide supporting evidence for the core problems IVP addresses.

Sun et al. (2025) propose SVIP, a cryptographic verifiable inference protocol using hidden layer representations to ensure computational integrity during model execution. While focused on technical verification mechanisms, their work shares IVP's commitment to replacing self-attestation with externally verifiable processing.

Dongre et al. (2025) quantify context drift in multi-turn interactions using KL divergence between response distributions, demonstrating measurable degradation patterns consistent with IVP's observational basis. Their equilibrium framework provides formal grounding for the degradation IVP mitigates through incremental checkpointing.

Abdelnabi et al. (2024) use activation pattern analysis to detect task drift, showing that LLMs can deviate from assigned objectives during extended interactions. Their activation delta detection complements IVP's behavioral approach by providing internal signal correlates for the external drift patterns IVP addresses procedurally.

Rath (2026) quantifies agent drift in multi-agent systems, reporting a 42% reduction in task success rates over 300 turns even in well-structured delegation chains. This empirical finding directly validates IVP's conservative stance on AI–AI adjudication and serial delegation risk.

Choi et al. (2025) examine identity drift in conversational agents, demonstrating progressive semantic shift in role adherence during extended exchanges. Their findings support IVP's requirement for re-verification after substantial additional interaction.

These studies collectively provide empirical grounding for phenomena IVP addresses through procedural design. IVP contributes a practitioner-validated methodology that operates independently of specific architectures and can be implemented immediately without specialized tooling.

Methodological Status

IVP is a practical protocol specification derived from observed failure modes across thousands of human-AI interactions spanning multiple architectures and platforms (2022-2026).

What this document is:

- A conceptual framework for verifiable document processing
- A methodological specification of required process components
- An architecture-agnostic protocol definition

What this document is not:

- A controlled empirical study with quantitative validation
- A technical implementation guide or code repository
- A platform-specific integration manual

Development basis:

Observational pattern synthesis from extended interaction with over 5,000 distinct AI instances (2022-2026), with approximately 500 conversations documented for analysis. This constitutes methodology development from practitioner experience, not controlled experimental research.

The author's documented interaction corpus provides substantial observational grounding for identified failure modes and proposed solutions. However, this document does not present that corpus as empirical evidence, nor claim statistical validation.

Validation pathway:

Practitioners and researchers are encouraged to test IVP implementations against baseline approaches. If the protocol does not demonstrably improve downstream task reliability, it should be refined or rejected.

Conclusion

The Ingestion Verification Protocol transforms document ingestion from an implicit assumption into an observable, adjudicated process. Version 2.6 clarifies implementation flexibility and automation boundaries while preserving the protocol's core commitment to verification over architectural speculation, and emphasizes that human adjudication remains non-negotiable regardless of implementation mode.

More information and current public materials are available at https://synthience.org

References

Abdelnabi, S., Fay, A., Cherubin, G., Salem, A., Fritz, M., & Paverd, A. (2024). Are you still on track!? Catching LLM Task Drift with Activations. arXiv:2406.00799. https://arxiv.org/abs/2406.00799

Choi, J., Hong, Y., Kim, M., & Kim, B. (2025). Examining Identity Drift in Conversations of LLM Agents. arXiv:2412.00804. https://arxiv.org/abs/2412.00804

Dongre, V., Rossi, R. A., Lai, V. D., Yoon, D. S., Hakkani-Tür, D., & Bui, T. (2025). Drift No More? Context Equilibria in Multi-Turn LLM Interactions. arXiv:2510.07777. https://arxiv.org/abs/2510.07777

Rath, A. (2026). Agent Drift: Quantifying Behavioral Degradation in Multi-Agent LLM Systems Over Extended Interactions. arXiv:2601.04170. https://arxiv.org/abs/2601.04170

Sun, Z., Zhang, Y., Wang, X., Chen, L., & Liu, H. (2025). SVIP: Secure Verifiable Inference Protocol for Large Language Models. arXiv:2410.22307. https://arxiv.org/abs/2410.22307