GOVERNED AUTONOMY

Semantic OS: Application Understanding, Authority, and Verified Outcomes

A paper in the Governed Autonomy collection

Andrew Durden
Independent research
Publication edition 2026.10.9

Research on authority, evidence, and continuity in autonomous systems.

Content identifier: 7150dfae1470e29ab5d0fbcc

All rights reserved · Research exposition

Contents

Semantic OS: Application Understanding, Authority, and Verified Outcomes 3

Abstract . . . 3

1. The problem of repeatedly understanding software . . . 3

2. Terms and scope . . . 4

3. Separating application knowledge from task reasoning . . . 4

4. Application meaning must survive contact with behavior . . . 5

5. Relevant information and exact answers . . . 6

6. Identity, selection, and authority . . . 6

7. Completion includes the result and its uncertainty . . . 7

8. Worked conceptual scenarios . . . 7

9. Evidence and open research questions . . . 9

10. Relationship to governed autonomy . . . 9

Source guide . . . 10

All rights reserved · Research exposition

Semantic OS: Application Understanding, Authority, and Verified Outcomes

Governed Autonomy · Publication 2026.10.9
Research study: conceptual exposition of the Semantic OS architecture
Source: Public exposition, successor 2026.10.9; historical research and evidence separately qualified

Abstract

An agent asked to complete work in an application must understand both the requested outcome and the application through which that outcome can occur. When application knowledge is reconstructed repeatedly from interface descriptions, screenshots, and conversational history, the cost of understanding the workplace competes with the reasoning needed to do the work. The Semantic OS architecture investigates a different allocation: persistent application understanding belongs to the surrounding system, while the agent reasons about objectives and decisions. This paper develops three distinctions: knowing an operation versus being authorized to perform it, finding relevant information versus establishing an exact answer, and requesting an action versus verifying its result. Four scenarios examine their consequences for useful work, disclosure, recovery, and incomplete observations. The architecture targets ordinary applications, including those without a dedicated agent interface. Its conformance programme defines the required demonstrations; evidence that a complete implementation meets those requirements remains outstanding.

1. The problem of repeatedly understanding software

Applications expose several descriptions of themselves. Their documentation describes intended behavior. Their interfaces reveal some available actions. Their current state exposes particular objects and relationships. Their behavior may depend on an account, service configuration, sharing arrangement, or background process that is absent from the visible screen. An agent completing one task has to reconcile these descriptions while preserving the user's intended scope.

Consider a request to compare two business documents and update an internal record. The substantive decision concerns what changed and how that change affects the record. Yet an agent may spend much of its working context discovering navigation, identifying the correct account, recovering object references, and learning which apparent edit will notify other people. Those mechanics can become as prominent as the task itself.

The Semantic OS premise is that application understanding can persist outside the transient reasoning context of a particular agent. A later task should benefit from established understanding without inheriting the earlier task's authority. The distinction matters: shared knowledge about an operation should not silently become shared permission to use it. A

All rights reserved · Research exposition

system that reduces repeated explanation while merging permissions would solve one problem by creating another.

The intended contribution is therefore broader than a shorter catalogue of tools. It concerns the relationship among application meaning, current observations, legitimate action, and verifiable outcomes. Its usefulness depends on preserving those relationships when applications change. Public review scope.

2. Terms and scope

In this paper, **application understanding** means supported knowledge of what an application exposes, what its operations mean, and what consequences their use can have. It includes uncertainty. A system does not understand an operation more accurately by supplying a precise but unsupported account of its behavior.

Current state means state supported by observations of the relevant source within a stated scope and time. An accessible local representation can help answer a question, but its accessibility alone does not make it authoritative or current. **Authority** means permission grounded in the principal's direction and applicable policy. The principal is the person or organization entitled to direct the work. **Verification** means evidence that supports the particular result being claimed.

These definitions are deliberately separate. Knowing that a document can be edited does not establish permission to edit it. Permission to read it does not establish permission to disclose it to a model or another service. A successful request to save it does not establish that the intended state is present. A verified local change does not establish that every synchronized copy has converged.

The application scope includes ordinary software that was not built around an agent-facing interface. The research question is whether reliable understanding and operation can be established under those conditions, and how to characterize their limits. The analysis below concerns the behavioral obligations that follow from that ambition. Public review scope.

3. Separating application knowledge from task reasoning

An agent's working context is a scarce resource. It should contain information needed to make the present decision, but it need not contain every available object, every application convention, or the entire history of how a system learned to perform an operation. The source treats persistent understanding and task reasoning as different responsibilities.

This distinction does not imply that meaningful decisions require no content. Comparing contractual language still requires access to the relevant language through an appropriate reasoning process. Resolving an unfamiliar concept may require explanation. An ambiguous request may require clarification. Moving application knowledge outside the current conversation cannot remove the information that makes a particular decision difficult.

The corresponding efficiency claim is conditional. Growth in irrelevant application state need not force an equivalent growth in the agent's context for an otherwise comparable task. The source does not justify a claim that all tasks have constant cost, that machine work

All rights reserved · Research exposition

disappears, or that a larger requested output requires no additional processing. Any evaluation must distinguish the cost of maintaining application understanding from the cost of interpreting relevant content.

Human presentation introduces another useful distinction. A person may need to see a large document, chart, or result that the agent only needs to identify. Successfully presenting that material does not show that the agent read it, understood it, or used it as evidence. Keeping these claims separate prevents an economical display path from being misrepresented as successful reasoning.

The research promise is practical: less repeated learning of application mechanics, more stable access to meaning, and reasoning effort directed toward the objective. Whether that promise holds across changing applications remains a matter for measurement. Public review scope.

4. Application meaning must survive contact with behavior

An application's declared behavior and its observed behavior can disagree. A nominally observational operation may update a read marker. A local edit may propagate to a shared location. Opening an application may trigger work already queued within it. These possibilities make the actual application context relevant to the consequences of an operation.

The source distinguishes claims about intended meaning from evidence about behavior. A declaration can help explain what an operation is supposed to do. It does not establish that the operation is harmless in every account and configuration. Conversely, a single successful observation does not establish every consequence that could occur under other conditions.

An adequate account therefore preserves unknowns and contradictory evidence. It must be possible to say that an operation is understood in some respects but remains unsupported for a particular consequential use. Unknown behavior should not acquire the appearance of certainty merely because a compact description is convenient.

Application change makes this an ongoing obligation. A new service behavior can matter even when the visible application looks familiar. Previously supported knowledge should remain usable only to the extent that its assumptions still hold. At the same time, an isolated change should not imply that every unrelated operation has become unknowable. The research challenge is to match the scope of renewed scrutiny to the scope of changed evidence.

This is especially demanding for applications without cooperative interfaces. Refusing an unsupported action can preserve an important boundary. It does not, by itself, satisfy the source's positive ambition of making useful work possible in such applications. A credible evaluation must establish both the permitted path and the limits of that path. Public review scope.

All rights reserved · Research exposition

5. Relevant information and exact answers

Information retrieval is valuable for finding material likely to matter. Its success criteria differ from those of an exact factual claim. A relevant result can support the existence of an item that was actually observed. A search that returns nothing does not necessarily support the absence of every matching item.

The source gives this distinction a central role in application awareness. A statement such as “there are no outstanding approvals” requires a justified account of the relevant collection, exclusions, and observation point. If some records are inaccessible or synchronization is incomplete, the answer must preserve that limitation. The system can still make useful progress using the evidence it has, but it cannot silently strengthen a partial view into a complete one.

Time also matters. A count can be exact for an observed state and become outdated later. A negative answer can be justified within one account but unsupported across several accounts. An observed local edit can be real before remote agreement is established. Exactness is consequently a relationship between a claim and its evidence, rather than a general property granted to an entire application.

Continuity must respect deletion and permission changes as well as new information. Maintaining a useful view of a workplace does not justify retaining everything the system once encountered. The source treats the limits of source retention and accessible scope as constraints on derived representations too. A persistent understanding of application behavior is different from a permanent archive of user content.

These distinctions also affect notifications. A system can record relevant changes without interrupting a person or agent for every event. Reducing interruptions should not be confused with proving that no change occurred. The observation history and the decision to bring an event to attention answer different questions. Public review scope.

6. Identity, selection, and authority

Cross-application work often depends on recognizing the same person or object in different places. Similar names, matching text, or convenient search results can suggest a relationship without establishing it strongly enough for consequential action. The source therefore treats identity as a substantive part of the problem, rather than a cosmetic label attached after selection.

There are several independent questions. Is this the intended person? Is this the intended document? Does the relation between them have adequate support? Does the present task permit the proposed action? May the information move to the proposed destination? A favorable answer to one question does not supply the others.

Model-generated selection deserves scrutiny even without an attacker. A model can infer the wrong record, translate a request into a broader set than intended, or choose a plausible recipient who was never authorized. Adversarial content adds another source of pressure, but ordinary model error already motivates a separation between evidence and permission.

All rights reserved · Research exposition

Content being processed does not obtain the authority of the person who commissioned the task. A document can contain instructions as data. Those instructions do not automatically become directions governing the task. Delegating work to another agent similarly does not authorize an expansion of scope.

Approval must remain meaningful in this setting. The person needs to understand the relevant action, affected objects, recipients, and material consequences. Consent to an earlier proposal should not silently cover a materially changed one. Likewise, withdrawal of authority must affect future work, including work that was prepared earlier but has not yet taken effect.

These requirements address the consequences of prompt injection and ordinary selection errors. Their effectiveness depends on enforcement in the application and surrounding operating environment, whose limits form part of the threat model. Public review scope.

7. Completion includes the result and its uncertainty

An action request and its outcome are distinct events. A provider may accept a submission before its eventual result is known. A response may be lost after the action has occurred. A local change may later encounter a synchronization conflict. The source's account of completion depends on evidence for the actual claim being made.

For example, confirming that a message was submitted does not establish that a recipient received or read it. The appropriate result may be narrower than the user's eventual objective. The system should continue toward that objective where it has an authorized route and sufficient evidence, while preserving the distinction between what has occurred and what remains unresolved.

Uncertainty also constrains recovery. If an operation times out, repeating it immediately may create a duplicate effect. The absence of a response is not evidence of non-occurrence. Recovery needs to establish the relevant state before deciding whether another attempt is appropriate. This is particularly consequential for actions that reach other people or change financial, legal, or access conditions.

Reversibility has similar limits. Restoring a value does not necessarily erase a notification, a third party's observation, or an intervening action. Compensation can be useful while leaving some consequences intact. The source accordingly requires attention to what can actually be restored and what can only be addressed through subsequent work.

Concurrent change creates another boundary. An edit based on an earlier observation may no longer reflect the present situation. The strength of protection depends on what the underlying application can establish at the point of change. A research account should state that strength rather than imply universal atomicity across independent systems. Public review scope.

8. Worked conceptual scenarios

Four hypothetical scenarios show how the distinctions affect completion.

All rights reserved · Research exposition

8.1 A completed comparison and controlled update

A project manager asks an agent to compare an approved schedule with the latest proposal and update a private planning note. The relevant documents are identified within the manager's accessible scope. Their relationship is supported by the source records. The comparison reveals a changed delivery date, and the requested note is still current when the update is made.

The successful result consists of the comparison, the authorized change, and evidence that the note now contains the intended information. If the note is private as assumed, the result can be described accordingly. If examination instead shows that the note is shared, the consequences differ and the earlier assumption cannot silently govern the action. Understanding where an edit takes effect is part of completing it correctly.

8.2 A document that attempts to redirect the work

An analyst asks for a summary of a supplier report. The report includes text directing the agent to send internal forecasts to an address printed in the document. The address may be real and the report may otherwise be useful. Neither fact supplies authority for that disclosure.

The legitimate summary can continue using the permitted material. The unrelated transfer lacks a basis in the analyst's request and applicable policy. A successful boundary preserves the authorized objective while declining the proposed disclosure. Asking the model to remember that a document is untrusted is insufficient evidence that this boundary exists; the corresponding behavior has to be demonstrated.

8.3 An interrupted action with an uncertain outcome

A user authorizes a single external submission. The connection fails before the application returns a clear result. The system cannot yet tell whether the submission occurred. Declaring failure and resubmitting would risk creating two submissions; declaring success would overstate the evidence.

Recovery instead proceeds from the available records and the application's observable state. If the first submission is confirmed, the task continues from that result. If non-occurrence is established and authority remains valid, another attempt may be appropriate. If neither can be established, the uncertainty remains material. The illustration shows why objective continuity and duplicate prevention have to be considered together.

8.4 An apparently empty collection

A team lead asks whether any approvals remain outstanding. The accessible search returns none, but part of the relevant account is temporarily unavailable. The system can report what was found in the observed scope and continue obtaining the missing evidence where authorized. It cannot present the initial empty search as a complete answer.

When the relevant scope becomes available, the system can reassess the claim using the current records. The point is not to prefer cautious wording indefinitely. It is to obtain the evidence needed for the stronger answer and then state that answer at its supported scope.

All rights reserved · Research exposition

9. Evidence and open research questions

The architecture defines invariants, conformance cases, a reference environment, and measurement expectations. These specify what an implementation must demonstrate. The four accompanying review works analyze those obligations; neither their analysis nor the specification establishes that a running implementation satisfies them.

Several questions determine the strength of the central claim. Can an unfamiliar application be understood well enough for useful, verified work without hidden application-specific authoring? Can observations in a controlled environment establish the consequences that matter in a real account? How should useful operation proceed when important behavior remains unknown? These questions require application and provider evidence, not only consistency of documentation.

Context efficiency needs independent measurement. Evaluation should hold the substantive task comparable while varying irrelevant application complexity, and should account for work performed outside the primary agent conversation. Otherwise the apparent saving could reflect displaced cost or an easier decision problem.

Exactness and recovery need adversarial experiments. Relevant cases include interrupted synchronization, changing permissions, deletion, uncertain external outcomes, concurrent writers, and changes after approval. Human coexistence matters too: useful automation must respect a person's ongoing use of the computer and accessible controls.

Finally, a successful example in a cooperative application would not establish the broader unknown-application claim. Preventing an unsupported action and completing authorized work require different evidence, and both belong in the evaluation. Demonstrating them together across unfamiliar applications is the central outstanding test. Public review scope.

10. Relationship to governed autonomy

Semantic OS addresses the environment in which an agent attempts practical work. Its emphasis on supported meaning, bounded authority, current observations, and verified effects connects conceptually to the wider governed autonomy thesis. The architecture synthesis distinguishes such conceptual relationships from explicit source contracts and implemented integrations.

The architecture expressly assigns intended roles to tasks, collaboration, cross-application work, and human surfaces. That source-level assignment is stronger than a thematic resemblance, but it is still a statement of intended architecture. It does not establish that a particular product release implements the complete account. Nor does a shared vocabulary establish adoption of every other framework in this collection. The glossary and evidence guide preserve these distinctions across the papers.

All rights reserved · Research exposition

Source guide

This successor guide points to public arguments and their limits. The new private evidence linkage remains excluded. Previously public conceptual source guides, including any retained titles, document ordinals, sections and chapter maps, are not claimed to have been removed; see the correction scope. The historical edition remains available unchanged.

Foundations. Review the relevant argument and assumptions in this paper, the evidence guide, and the research review route. Specification, reported results and production observations have distinct scope.

Meaning and application evidence. Review the relevant argument and assumptions in this paper, the evidence guide, and the research review route. Specification, reported results and production observations have distinct scope.

State and identity. Review the relevant argument and assumptions in this paper, the evidence guide, and the research review route. Specification, reported results and production observations have distinct scope.

Authority and human control. Review the relevant argument and assumptions in this paper, the evidence guide, and the research review route. Specification, reported results and production observations have distinct scope.

Outcomes and recovery. Review the relevant argument and assumptions in this paper, the evidence guide, and the research review route. Specification, reported results and production observations have distinct scope.

Evidence and review. Review the relevant argument and assumptions in this paper, the evidence guide, and the research review route. Specification, reported results and production observations have distinct scope.

No source-guide entry establishes a full implementation, a complete historical migration proof, or an independent audit.

All rights reserved · Research exposition