

117-321-D Gudiya's Engineering Objective

Preserving the Reliability of the AI Economy's Dependency Substrate

Ashish Warudkar

Introduction

This chapter examines seven reasons why substrate liquefaction is a real risk for the AI economy:

1. the proliferation of actants,
2. the potentially faster growth of dependencies,
3. changing conditions supporting those dependencies,
4. adaptation,
5. non-deterministic behavior,
6. machine-speed execution, and
7. limits on timely observation and intervention.

Together, these factors can undermine the assumptions that allow individually functional agents, APIs, and services to produce dependable collective outcomes. “Substrate liquefaction” describes this loss of dependable support across the ecosystem.

The engineering responsibility

Gudiya Grid proposes to preserve the operational reliability of the dependency substrate on which the AI economy runs. Its engineering objective is to maintain the shared conditions under which independently operated agents, models, APIs, databases, and infrastructure can continue to produce dependable collective outcomes. As participation expands, relationships multiply, behavior adapts, and execution accelerates, those conditions must be assessed during operation.

A successful API call establishes something about that call. It does not establish that every dependent commitment remains fulfillable. A database may preserve its transaction rules while the business assumptions behind a transaction become obsolete. A service may remain available while thousands of agents converge upon the same scarce resource. The gap between local success and collective dependability is the subject of this chapter.

What we mean by substrate

The substrate is the operational fabric of dependencies and supporting conditions through which participants rely on one another. It includes the relationships among actants, the resources they share, the commitments they make, and the timing and information on which those commitments depend. An actant is any participating entity capable of contributing to or affecting execution, including an agent, service, human, or machine.

This substrate is distributed throughout the economy. A supplier in one relationship is a requester in another. A service can be functioning correctly within its own scope while forming part of an unreliable supporting environment for someone else. The substrate therefore cannot be understood by inspecting individual participants alone.

Gudiya is an operational engineering proposal. It does not require cognition to be a fundamental physical field. Its claim is that changing dependencies and shared constraints can be represented, observed, and governed sufficiently well to improve collective reliability within a defined participating scope.

The dependency network becomes the substrate

Begin with D1. It depends on D2 and D3. D2 depends on D4 and D5, while D3 depends on D6 and D7. An arrow points from a dependent toward the entity on which it relies. Extend these relationships across organizations, execution environments, and resource services, and the simple branching picture becomes an interwoven network containing shared dependencies and feedback paths.

In the left state, supporting conditions remain adequate for the illustrated interactions. In the right state, selected relationships become unreliable and expose their dependents. The nodes need not all fail. The same network can become less dependable because the conditions supporting its relationships have deteriorated.

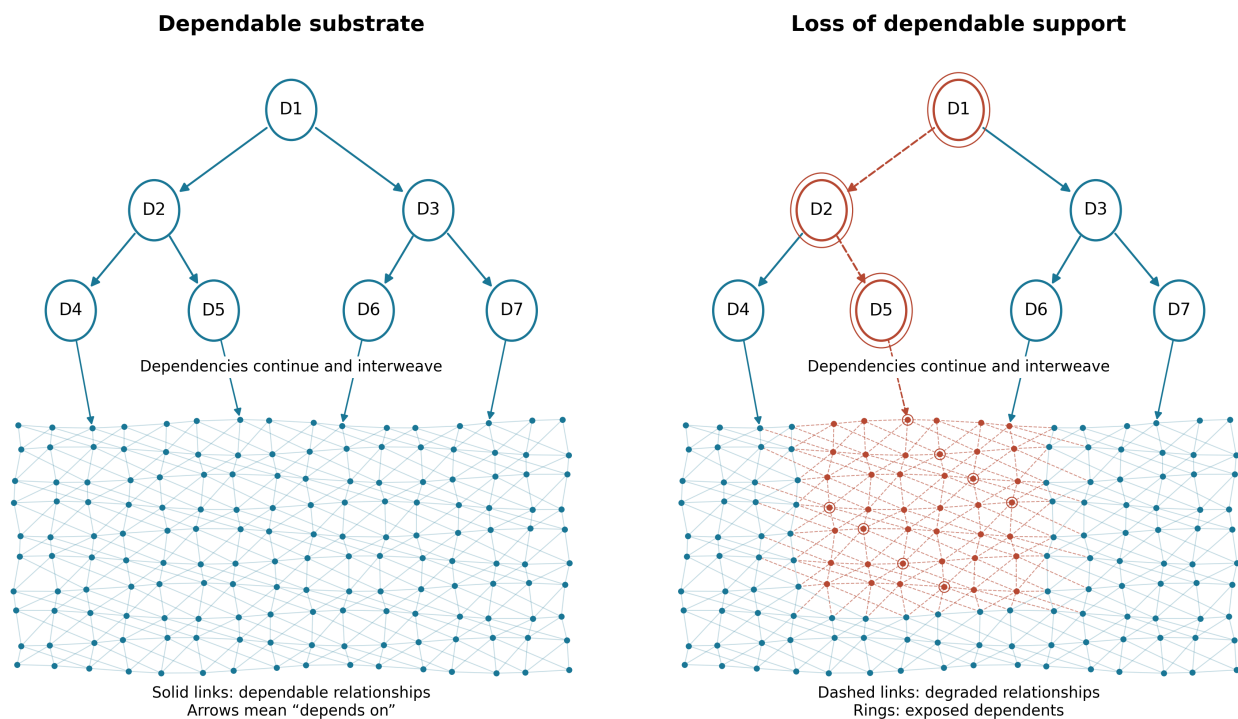


Figure 1. Dependency network under dependable and degraded supporting conditions. Hundreds of millions of actants and billions of dependencies describe the envisioned scale; the dots shown are schematic, not measured data.

This is the intended meaning of substrate liquefaction: the environment ceases to support the assumptions through which individually correct operations were expected to compose into a dependable result.

Liquefaction is a systems metaphor here. A degraded relationship does not automatically imply collapse; redundancy, isolation, and recovery may contain the disturbance. The concern is deterioration that spreads or defeats the available means of containment.

More actants and faster growth of dependencies

1. The first dimension is numerical expansion. Each additional actant brings activity, state, resource demand, and some need for identification and accountability. Even with fixed behavior, a larger population can increase the volume of concurrent work and the burden of observing it. The challenge is influenced by concentration: many actants relying on one provider may be more consequential than the same number distributed across independent providers.
2. The second dimension is the potentially faster growth of dependencies. If each of N actants maintains an average of k outgoing dependencies, the network contains approximately $N \times k$ directed relationships. If k stays fixed, relationships grow linearly with population. If connectivity also increases, relationships can grow faster than population. With one directed relationship permitted for each ordered pair of distinct actants, the fully connected ceiling is $N \times (N - 1)$.

This is the defensible meaning of a super-numerical expansion of dependencies. Geometrical or exponential growth is not an automatic law of agentic systems. Branching can multiply the number of dependency paths over successive levels, while shared services and cycles cause those paths to overlap. Actual risk depends on topology, coupling strength, timing, and concentration as well as counts.

Every dependency has supporting conditions

3. The third dimension is the state surrounding each relationship. An arrow on the map conceals a practical contract: the provider must have sufficient capacity, the requester must possess the necessary authority, the information must be timely enough, and the requested result must arrive while it is still useful. A relationship can remain technically connected after one of these conditions has ceased to hold.

Consider a procurement agent that receives a valid inventory response and places a valid order. Other agents may consume the remaining inventory before its reservation completes. Alternatively, goods may remain available while delivery capacity disappears. Correct message handling is compatible with failure of the larger plan. The relevant question is whether the supporting conditions continue to hold through the commitment and its settlement.

Gudiya would therefore need a representation richer than a connectivity diagram. Critical relationships should expose the conditions relevant to their safe continuation, together with evidence age and uncertainty. Some conditions will be directly measured; others will be estimated or unavailable. Unknown support must remain visibly unknown rather than being silently treated as dependable.

Adaptation changes the network during execution

4. The fourth dimension is adaptation. An agent may revise its plan, substitute a supplier, delegate to another agent, or change its retry behavior. These responses may be useful individually. Collectively, they can redirect demand, create new dependencies, and alter the conditions faced by participants that did not change their own behavior.

Imagine many agents selecting an apparently faster supplier after a delay elsewhere. Their adaptations converge on the same destination. That supplier's conditions deteriorate, causing another round of substitutions. The actants, relationships, and supporting environment change together. A previously adequate assessment can become obsolete before the work it authorized has settled.

Adaptation can also improve reliability by finding alternatives and avoiding impaired services. The engineering task is to distinguish helpful adjustment from adjustment that amplifies a shared disturbance. A static

inventory of approved participants cannot answer that question by itself; relevant changes in behavior and dependency must be observed during execution.

Non-deterministic Behavior and the Uncertainty of Dependence

5. The fifth dimension is non-deterministic behavior. This introduces uncertainty into the dependency substrate because knowing an actant's current condition does not necessarily establish what it will do next. An agent may choose among several permitted actions, each creating different demands, commitments, and consequences for its dependents. D1's reliance on D2 therefore involves both D2's availability and the range of behaviors D2 may exhibit while that dependency remains active.

Across an adaptive economy operating at machine speed, these possibilities interact, making a single forecast an insufficient basis for assurance. Better observation can reduce uncertainty about the present without eliminating uncertainty about future behavior. Gudiya's engineering objective is consequently to preserve dependable operation across a defined range of permitted behaviors through enforceable limits, adequate margins, coordinated commitments, and containment of disturbances. Non-determinism becomes a substrate risk when behavioral variation exceeds what the surrounding dependencies can accommodate; stabilization must address that exposure without requiring perfect prediction.

Machine speed compresses the response window

6. The sixth dimension is the speed at which activity and change occur. Requests, delegations, retries, and new commitments can be initiated before a human can inspect the first consequence. Removing human mediation also removes some pauses and review opportunities, although human participation never guaranteed stability. The exposure depends on the relative speeds of disturbance, observation, intervention, and actual fulfillment.

A fast request does not make every downstream process fast. Compute can initiate orders more quickly than a warehouse can dispatch goods, and digital acknowledgments can arrive before physical obligations settle. The result can be rapid accumulation of outstanding commitments against slowly changing capacity. Gudiya must therefore observe unfinished obligations as well as the rate of new requests.

The relevant requirement is that observation, assessment, communication, and enforcement complete within the window available to prevent unacceptable propagation. Faster dashboards alone do not satisfy this requirement. Some events require local enforcement or capture at the point of commitment. Nyquist reasoning matters for sampled, band-limited signals, but no universal sampling rate establishes adequate observability for discrete commitments and changing dependencies.

These six dimensions compound one another. More participants create more opportunities for coupling; more coupling introduces more supporting conditions; adaptation changes those conditions; machine speed reduces the time available to recognize and contain the consequences. This is a risk mechanism, not a claim that growth inevitably produces instability.

Observability and the Time Available for Stabilization

Gudiya's engineering objective depends on whether changes in the dependency substrate can be observed and acted upon before they undermine dependable operation. Knowing which actants depend on one another is only the beginning. The Grid must also establish whether their supporting conditions remain adequate, how quickly those conditions are changing, and how much time remains for effective intervention.

7. This introduces a seventh dimension on substrate assurance – the temporal constraint. A Cognition Field Map may accurately describe an earlier state while no longer representing the conditions under which agents are making commitments. As adaptation and machine speed accelerate change, the interval between observation and action becomes consequential. The substrate may lose capacity, accumulate obligations, or develop reinforcing feedback while the displayed map still appears healthy.

The Nyquist–Shannon sampling theorem provides a precise insight into one part of this problem. For an ideally band-limited signal, reconstruction requires a sampling rate greater than twice its highest frequency. Insufficient sampling can cause aliasing: rapid variation may appear slower, different, or even absent. Applied carefully to Gudiya, this means that selected telemetry must be sampled at rates appropriate to the dynamics it is intended to reveal. An apparently stable measurement can reflect inadequate observation rather than stable operation.

The theorem does not establish a universal observation rate for the AI economy. Dependencies, commitments, and changes of authority are often discrete events rather than conveniently band-limited signals. Some must be captured when they occur, with their ordering and context preserved. Faster sampling also cannot reveal a dependency that has never been instrumented. Gudiya’s observability requirements therefore extend across sampling rate, event completeness, information freshness, causal relationships, and uncertainty.

Observation must then lead to timely action. Detection, communication, assessment, and enforcement consume part of the available response window. That window depends on how quickly the relevant operating margin is being exhausted and on the consequences of crossing its boundary. A correct intervention delivered after an irreversible commitment may be unable to prevent the outcome. Gudiya must consequently assess both the adequacy of its observations and the time required to make its controls effective.

This favors a distributed arrangement of observation and control. Local execution environments can enforce immediate limits and capture critical events. Rendezvous Management can assess whether interactions should begin or continue. The Cognition Field Map can integrate broader evidence to identify shared constraints and propagation across domains. These functions operate at different timescales; the refresh rate of a human-facing map should not determine the response speed of local protection.

Where critical changes cannot be observed and controlled within the available time, Gudiya must reduce exposure. Depending on the situation, this may require limiting new commitments, reducing concurrency, restricting coupling, reserving capacity, or suspending an interaction until supporting conditions can be established. Accelerating activity is supportable only when observation and control remain adequate to its consequences.

Gudiya’s engineering objective therefore includes preserving the relationship between the speed of substrate change and the speed of effective stabilization. Substrate assurance must account for what the Grid can observe, how current that knowledge is, and whether intervention can still arrive in time. The absence of a detected disturbance is insufficient evidence that the substrate remains dependable.

Margins make support possible

Margin is the remaining room between current operating conditions and a relevant limit. It is not the limit itself. A relationship may have capacity headroom but almost no timing margin, or sufficient processing resources but inadequate confidence in the information governing a commitment. Substrate reliability consequently depends on several margins that cannot safely be collapsed into one unexplained score.

Margins can erode because demand increases or because supporting capacity falls. An unchanged actant can therefore become exposed without becoming more aggressive. Once a critical margin is exhausted, queued work, retries, or redirected demand may transfer pressure elsewhere. However, positive estimated margins do not guarantee reliability: the model may omit a dependency, the measurement may be stale, or several apparently separate resources may share one point of failure.

The assurance gap across organizational boundaries

Operators already engineer distributed services, manage dependencies, and mitigate cascading failures. Google's published reliability guidance, for example, describes how overload, retries, and redistribution of traffic can amplify disruption, and discusses measures such as load shedding and controlled retries. These are established foundations for this discussion. [1]

The gap is that local and organizational assurances do not automatically combine into assurance for the evolving network across organizations. A model provider, API operator, database owner, and data center each has a defined scope. No individual participant's local guarantee, by itself, establishes that all cross-domain dependencies remain supportable as the participants adapt.

Gudiya proposes to make that wider responsibility explicit. Within a defined governed scope, the Grid would assume responsibility for monitoring and preserving the shared conditions under which participating services can continue to work dependably together. Component owners retain their obligations. The Grid adds an accountable stabilization function across their boundaries.

An assurance commitment must have limits

Taking responsibility means specifying what is observed, what can be controlled, which outcomes are required, and how failure and recovery are handled. It cannot mean guaranteeing every result across the entire economy. External providers, hidden dependencies, missing telemetry, and contested authority limit coverage. A credible assurance regime declares those limitations and adjusts permitted activity when the evidence or control authority is insufficient.

This makes substrate reliability a concrete engineering objective: sustain agreed service and settlement outcomes under defined conditions, contain disturbances within agreed limits, and recover without creating a second wave of instability. The assurance must be supported by evidence from operation and testing.

How Gudiya would discharge the responsibility

The Cognition Field Map would represent critical dependencies, shared constraints, and deteriorating margins across participating domains. Its value would depend on the freshness and coverage of its evidence. It should allow operators and control functions to distinguish an isolated problem from one exposing many dependent activities, while retaining uncertainty where the situation cannot be established.

Rendezvous Management would use those conditions to manage admission, mutual readiness, and continuation of interactions. Discovery of a provider is insufficient evidence that the provider can accept a new commitment. The proposed rendezvous function would help reconcile demand with available support and communicate restrictions to affected predecessors. Those responses should follow actual dependencies rather than indiscriminately slowing the entire network.

Grid stabilization would enforce appropriate constraints when activity threatens supportable limits. Local execution environments would need mechanisms to implement those constraints promptly. A common field view does not require every fast decision to traverse a single central controller. Shared policies and federated coordination can coexist with local enforcement, provided responsibility and conflict resolution are defined.

Stopping activity does not settle its dependencies

When an actant is damped or stopped, outstanding obligations remain. Inbound requests may accumulate, counterparties may retry, and other agents may reroute toward a shared alternative. The field map must expose that pressure, and rendezvous controls must adjust affected flows. Recovery should revalidate conditions and release accumulated work at a supportable rate. An acknowledgment of a stop is only one event in the stabilization process.

Demonstrating the engineering objective

Evaluation should compare Gudiya-enabled operation with a credible baseline containing existing reliability controls. Tests should vary population, connectivity, shared-resource concentration, adaptation, information delay, and request speed. Useful measures include completion within agreed conditions, unmet commitments, propagation beyond the initial failure domain, intervention latency, recovery time, and the cost of unnecessary restrictions. Claimed improvements require repeatable evidence under stated workloads.

The Grid itself becomes part of the dependency substrate. Its observations can be wrong, its control channels can fail, and simultaneous interventions can conflict. Its design therefore needs fault isolation, bounded authority, explicit behavior during communication loss, and evidence that its controls do not introduce new instability.

Gudiya's engineering objective is to preserve the conditions for dependable composition as the AI economy changes. Local excellence remains essential. Gudiya proposes an additional responsibility for the evolving relationships through which that excellence becomes a collective outcome.

Source and scope

[1] Mike Ulrich, "Addressing Cascading Failures," Google, Site Reliability Engineering, Chapter 22. <https://sre.google/sre-book/addressing-cascading-failures/> (accessed 14 October 2026). The Gudiya architecture, dependency illustration, and proposed assurance framework in this chapter are conceptual proposals, not demonstrated deployment results.

