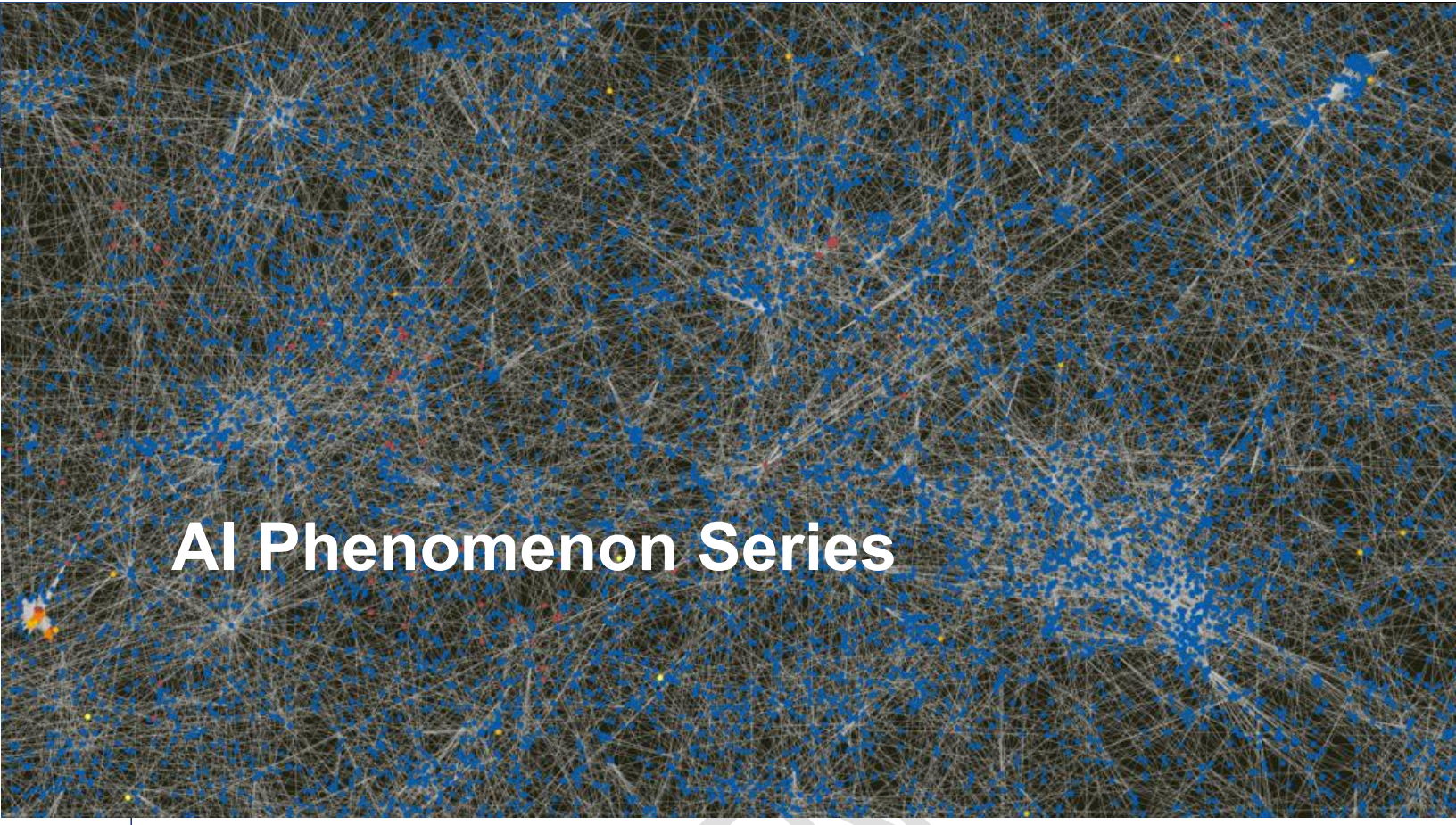

Directed Acyclic Graphs for Knowledge Assimilation in Causal AI Systems

Answering the 5 What Imperatives for Cyber Situation Awareness, Full Spectrum Cyber Operations, and National Security

Scott Foote

Last Updated: 28 December, 2025

Phenomenati Consulting
6 Liberty Square, #2736
Boston, MA 02109
(508) 709-7990 (office)
(617) 404-9419 (fax)

www.phenomenati.com

Disclaimer. This document describes architectural and analytical patterns for integrating causal models into decision support. It does ***not*** provide instructions for unauthorized access, exploitation, or other harmful cyber activity. All cyber operations must be conducted under appropriate legal authorities, policy oversight, and rules of engagement.

Contents

Executive Summary

Security operations and national security stakeholders face a persistent challenge: turning vast, fast, and imperfect data into decisions that are explainable, auditable, and aligned to mission outcomes. Contemporary analytics excel at correlation and prediction, but frequently fail to answer causal questions such as: What caused this? What will happen if we intervene? What would have happened if we had acted earlier?

Directed acyclic graphs (DAGs) - operationalized as structural causal models (SCMs) - provide a practical backbone for assimilating knowledge from supervised learning, unsupervised learning, and human expertise into a single, coherent reasoning substrate. A DAG-based system can combine: (1) learned mechanism models (e.g., compromise likelihood given exposure and controls), (2) discovered structure signals (e.g., latent clusters, anomalies, candidate edges), and (3) engineered constraints and priors (e.g., asset dependencies, doctrine, threat intelligence).

This whitepaper proposes a reference architecture for DAG-centric Causal AI that supports the 5 What Imperatives popularized in cyber situation awareness practice: What, So What, What Else, Now What, and What If. It shows how DAG operations (probabilistic inference, causal effect estimation, intervention planning, and counterfactual simulation) map naturally to those five questions across three mission contexts: Cyber Situation Awareness (CSA), full spectrum Cyber Operations, and national security decision support.

1 Problem Statement: From Data Exhaust to Decision Advantage

Cyber environments are complex adaptive systems: they change continuously, feature strong dependencies, and include intelligent adversaries. In these settings, purely statistical pattern matching can be brittle. The same alert can imply radically different risk depending on context such as: asset criticality, privilege level, business dependencies, compensating controls, and adversary intent. Decision makers therefore need models that can represent and reason over context - not simply detect anomalies.

Causal AI adds a missing layer: it differentiates ***causation*** from simple ***correlation***, formalizes assumptions, and supports 'what if' counterfactual reasoning through interventions. DAGs are a compact, computable representation of causal assumptions and enable end-to-end traceability from evidence to decision.

2 The 5 What Imperatives™ as a Requirements Lens

This author began working on "cyber situation awareness", based upon causal graphs and Bayesian networks, in 2005 while working at a small IP geolocation startup in the Silicon Valley area. The work was rudimentary then, but as it progressed to address the needs of a range of clients, a pattern of queries from senior leaders emerged which eventually were codified as the "**5 What Imperatives™**"[1].

Today, Phenomenati's "**5 What Imperatives™**" model frames Cyber Situation Awareness clearly as a series of decision-relevant questions[2]:

- What?
- So What?
- What Else?
- Now What? and
- What If?

In this whitepaper, we interpret these imperatives as follows:

What?

- Establish a defensible picture of **what is happening** now:
  - observed events,
  - targeted assets,
  - inferred incidents, and
  - likely causes.

So What?

- Translate the situation into **mission** and **business *consequences***:
  - operational impact,
  - risk, and
  - decision urgency.

What Else?

- Events and "incidents" in the cyber domain are rarely isolated. This type of question seeks to anticipate what other:
  - related events,
  - second-order effects, and
  - adversary follow-on actions

  are likely given current evidence.

Now What?

- In these types of queries, leadership seek to *select*, *evaluate*, and *justify* **response actions** that are:
  - feasible,
  - effective, and
  - aligned to policy, constraints, and priorities.

What If?

- Finally, in this type of query, all levels of Operations seek to explore **interventions** and **counterfactuals** to identify and evaluate:
  - plans,
  - tradeoffs, and

[1] https://whatimperatives.com
[2] https://thesoctaxonomy.com/soc-capability-area-12

- future scenarios.

A key implication of this observed spectrum of queries is that analytics must move ***beyond* detection** to support ***consequence* analysis** and ***action* Identification, evaluation,** and **selection**. DAG-centric Causal AI provides a single foundation that can support all five imperatives with explicit assumptions and uncertainty.

3 DAGs and Structural Causal Models (SCMs)

A directed acyclic graph (DAG)[3][4][5] represents *variables* as **nodes** and direct *causal influences* as directed **edges**, with the constraint that there are ***no*** directed cycles.

When *paired* with **structural equations** and a **probability model** over exogenous noise (e.g., independent fluctuations that originate outside the system and influence its behavior, such as malicious activity), the DAG becomes a **structural causal model (SCM)**[6][7][8].

This framework connects **three types** of questions:

1) ***observational*** queries (given evidence, what is the probability of an incident?),
2) ***interventional*** queries (what happens if we shift to a contingency, deploy a control or isolate a host?), and
3) ***counterfactual*** queries (what would have happened if we had patched earlier?).[9][10]

In practice, DAGs are valuable in cyber decision support because they can:

- ***encode*** (and test) independence **assumptions**,
- ***expose*** **confounding** (spurious associations that distort true causal relationships),
- ***separate*** **direct** from **mediated** effects, and
- ***enable*** **structured fusion** of heterogeneous signals (telemetry, intel, and human judgments).

[3] https://en.wikipedia.org/wiki/Directed_acyclic_graph
[4] https://www.ibm.com/think/topics/directed-acyclic-graph
[5] https://cran.r-project.org/web/packages/ggdag/vignettes/intro-to-dags.html
[6] https://medium.com/causality-in-data-science/structural-causal-models-a-quick-introduction-1ab49259e921
[7] https://www.activeloop.ai/resources/glossary/structural-causal-models-scm/
[8] https://www.stats.ox.ac.uk/~evans/APTS/scm.html
[9] https://bayes.cs.ucla.edu/BOOK-2K/
[10] https://miguelhernan.org/whatifbook

4 Knowledge Assimilation: Unifying Supervised, Unsupervised, and Expert Knowledge

Cyber environments generate multiple kinds of knowledge signals. A DAG-centric information assimilation layer turns these signals into:

a) an evolving **causal structure** (the graph),
b) **parameterized mechanisms** (structural equations or conditional probability models), and
c) **provenance** and **uncertainty** *annotations* that support *audit* and *continuous improvement*.

4.1 A Unified Knowledge Model

We recommend treating the causal graph as the ***'contract'*** between data science, engineering, and operations. All models – whether learned or engineered – map to variables in a shared schema and connect through the graph. This reduces **integration friction**, improves **explainability**, and supports **modular upgrades**.

4.2 Supervised Learning: Learning Causal Mechanisms

Supervised models can estimate specific causal mechanisms once the relevant variables are defined. Examples include: ***probability*** of **initial access** given exposure and vulnerability, **time-to-detection** as a function of logging fidelity, or ***probability*** of data **exfiltration** given privilege and segmentation. In an SCM, these become *structural equations* or *conditional models*.

To avoid 'prediction-only' failures, supervised learning should be *constrained* by the causal graph:

- features should correspond to ***causes***, ***mediators***, or ***confounders*** (not post-treatment leakage),
- and the model should be ***validated*** for stability under interventions.

4.3 Unsupervised Learning: Discovering Structure and Latent Variables

Unsupervised learning contributes in three main ways:

1) *identifying* **latent regimes** (e.g., baseline vs. degraded operations),
2) *discovering* **clusters** or **roles** (e.g., user archetypes, asset classes), and
3) *producing* candidate **causal edges** through **causal discovery** or **dependency screening**[11][12].

These outputs should not be accepted blindly. Instead, treat them as proposals that are filtered through:

1) **domain constraints** (e.g., physics of networks, IAM rules),
2) **governance checks**, and
3) **intervention** or **holdout tests** where feasible.

4.4 Expert Knowledge: Priors, Constraints, and Doctrine

In cyber and national security, the most valuable knowledge is often not labeled data – it is *doctrine*, *engineering constraints*, and hard-won *experience*.

Expert knowledge can be assimilated as:

- **edge constraints** (must-have, forbidden),
- **prior distributions** over effects, and
- **deterministic rules** that define variable semantics (e.g., **mission dependency mapping**).

The system should **record provenance** (who asserted what, when, and why) and enable **controlled disagreement**: multiple *competing* **model fragments** can *coexist* and be adjudicated with evidence.

[11] https://mitpress.mit.edu/9780262037310/elements-of-causal-inference/
[12] https://direct.mit.edu/books/monograph/2057/Causation-Prediction-and-Search

4.5 A Conceptual Reference Architecture

Figure 1 shows a **reference architecture** for Cyber Situation Awareness based upon DAG-centric knowledge assimilation and **5-What Reasoning**™ that has evolved over nearly 20 years of experimental application in real-world environments.

Because there are always those who insist on mis-construing the intent and use of reference architecture diagrams… it is important to clarify what this diagram is and is not. First, it is simply an ***abstraction*** – not a specification. It is ***conceptual***, not comprehensive. It is ***descriptive***, not prescriptive. It is intended to support a ***narrative*** that helps all parties to **understand** and **align** on the underlying complexity of establishing, employing, and continuously maintaining such a business-critical system.

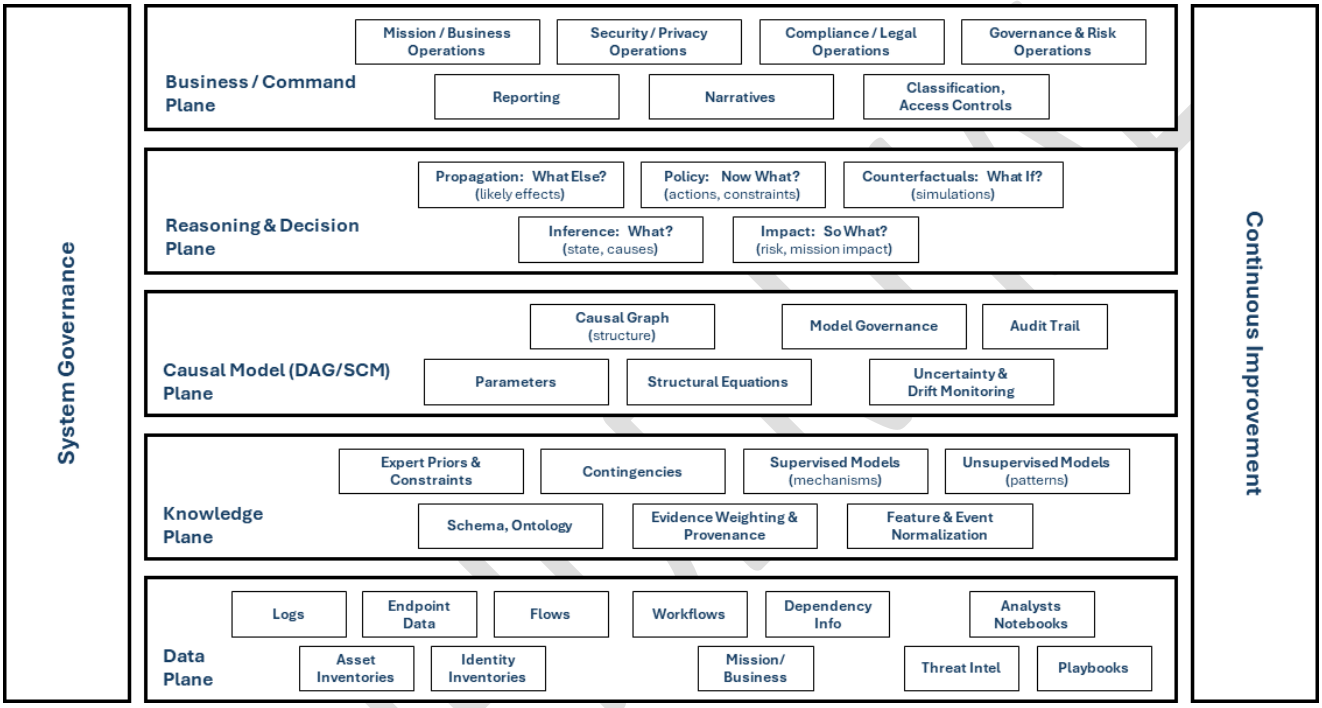

Figure 1. Reference Architecture for DAG-centric Causal AI supporting the 5 What Imperatives.

4.5.1 The Data Plane

The **Data Plane** is the system-of-record layer where raw **operational facts** are *collected*, *aligned*, and *governed* before any higher-order inference begins.

4.5.1.1 Foundations

Foundations establish the **minimum viable "truth set"** about what exists in the environment: **assets** (hardware, software, services, cloud resources), and **identities** for both people and non-person entities (service accounts, workloads, devices, keys, certificates). In practice, this is where **entity resolution**, **deduplication**, **naming conventions**, and **authoritative source selection** happen so later analytics do not chase phantom objects or contradictory identifiers. In this author's experience with the military, this foundational information was often referred to as "*cyber terrain*".

From a Causal AI perspective, Foundations are not just inventory... they define the **entities** that can become ***nodes*** in causal graphs and the principals (subjects) that can act (predicates), be acted upon (objects), or be *impersonated*. Good Foundations include strong **lineage** (where each record came from), consistent **timestamps**, and **governance** controls so that downstream reasoning can explain what evidence was used, what evidence was excluded, and why.

4.5.1.2 Monitoring

Monitoring supplies the **time-ordered evidence** stream that describes **how the environment behaves**. Logs provide discrete event traces (auth events, application events, configuration changes), endpoint data provides high-fidelity host-level telemetry

(process, file, registry, memory, sensor signals), and flows provide network-level and service interaction summaries (who talked to whom, when, how much, and via which protocol). Together, these data types form complementary views: **logs** capture **intent** and system **decisions**, **endpoint telemetry** captures local **ground truth**, and **flows** capture **connectivity** and **reach**.

In the Reference Architecture, Monitoring is designed for causal usefulness, not just collection volume. That means normalizing to common **event concepts**, preserving **context** such as host/user/session identifiers, and capturing ***negatives*** when possible (e.g., "no connection observed", "policy blocked"). Monitoring should also include **quality signals**... e.g., coverage gaps, sensor health, time skew, and sampling behavior... because causal conclusions are only as reliable as the observed evidence supports.

4.5.1.3 Context

Context ties technical **telemetry** to the ***purpose*** of the organization. Business or mission ***functions*** define what "*good*" looks like and which outcomes matter; ***processes***, workflows, and data flows describe how value is delivered; and ***dependency*** information shows what *relies* on what (applications on databases, services on identity providers, users on devices, plants on OT networks). This layer is where the architecture captures "***why*** this system exists" and "***what*** breaks when it fails."

Causal reasoning depends on context to **distinguish signal from noise** and to **interpret impact**. A process-aligned **dependency map** allows the same technical anomaly to be evaluated differently depending on where it occurs (e.g., a transient error in a dev pipeline versus a safety-critical control workflow). Context also enables **scoped reasoning**: the model can **constrain hypotheses** to plausible paths through known dependencies and can **prioritize investigation** based on business-critical chains.

In this author's experience with the military, this contextual information was critical to helping decision makers identify "***cyber key terrain***" from the perspective of a given mission – e.g., what *is* or *is not* ***mission critical*** to ***this*** mission.

4.5.1.4 Cyber Operations Data

Cyber Operations Data captures the organization's **accumulated knowledge** (covering *defensive* use cases only herein) and operational habits. **Threat intelligence** contributes external and internal **indicators**, actor **tactics**, and **campaigns**; ***playbooks*** encode preferred **response sequences** and **required approvals**; and **analyst/investigator notebooks** preserve the ***reasoning*** **trail** from past incidents, including **lessons learned** and "gotchas" unique to the environment.

This material is essential for turning a Causal AI engine into an operational teammate. It **seeds hypotheses** (what threats fit observed patterns), **constrains actions** (what must be approved, what is forbidden, what is reversible), and **improves explainability** (what precedent exists for similar incidents). Architecturally, this data should be treated as **first-class**, ***versioned*** **knowledge with provenance**... playbooks *change*, intel *expires*, and notebooks contain ***assumptions*** that must be made explicit.

4.5.2 The Knowledge Plane

The Knowledge Plane transforms the Data Plane's raw records into a **consistent, queryable representation** that supports causal learning and causal inference.

4.5.2.1 Information Architecture

Information architecture is the backbone: **schemas** define structured fields and types; **ontologies** define the concepts and relationships that unify heterogeneous data; and **feature/event normalization** ensures that "the same thing" is represented the same way across tools, vendors, and time.

Equally important are **evidence weighting** and **provenance**. Provenance tracks where each **assertion originated** (sensor, system, analyst note) and how it was transformed. **Evidence weighting** captures *reliability* and *confidence*... some sensors are noisy, some logs are incomplete, some intel is stale. When the architecture records these properties explicitly, later reasoning can surface not only a conclusion, but also the ***strength*** of the **supporting evidence** and the most valuable data to collect next.

4.5.2.2 Expert Knowledge

Expert knowledge bridges the ***gap*** between **what is observed** and **what is meaningful**. Expert **priors** and **constraints** encode domain truths and guardrails... for example, ***known impossibilities*** ("this subnet cannot route to that network"), expected ***causal directionality*** ("credential theft enables lateral movement, not the reverse"), or ***organizational policy*** ("production database access requires a break-glass workflow").

Contingencies for **dependencies** capture the **conditional behavior** of systems: *failovers*, *throttling*, circuit breakers, seasonal *load patterns*, maintenance windows, and "if X is down, Y behaves like Z." These contingencies are critical in cyber contexts because attackers often exploit rare or degraded modes. Unfortunately, contingencies are often difficult or impossible to be *directly observed* (e.g., by an agent or other piece of software) until the condition becomes true; prior to that such contingencies are often known only to humans with comprehensive system knowledge (e.g., design engineers, safety operators, etc.) of what to do when a rare circumstance manifests in the environment. **Capturing** and **representing** contingencies **explicitly** keeps the causal model from overfitting to steady-state assumptions and improves decision support during crises when systems are not operating normally.

4.5.2.3 Machine Learning

Machine learning (ML) in the Knowledge Plane is used to extract structure and signals that are difficult to hand-code at scale. **Supervised learning** supports classification and prediction where labeled examples exist (malware family classification, phishing detection, ticket categorization, incident triage recommendations). **Unsupervised learning** supports discovery where labels are sparse or evolving (entity clustering, behavioral baselining, novelty and anomaly detection).

In this architecture, ML is ***not*** the final decision-maker; it is a producer of ***features***, **hypotheses**, and **uncertainty estimates** that feed the causal layer. Models should be designed to output interpretable artifacts (scores, embeddings with explanations, clusters with exemplars) and to **preserve training context** and **drift signals**. That way, the causal model can use **ML-derived signals** as **evidence**... weighted appropriately... without inheriting opaque "black box" conclusions.

4.5.3 The Causal Model (DAG/SCM) Plane

The Causal Model Plane formalizes **cause-and-effect assumptions** as directed acyclic graphs (DAGs) and the structural equations that bind them.

4.5.3.1 Directed Acyclical Graphs (DAGs)

In these DAGs, **Nodes** represent **variables** of interest... system states, behaviors, controls, exposures, and outcomes... and include **parameters** that **quantify relationships** (strength, delay, probability, thresholds). **Edges** represent hypothesized **causal influence**, and **structural equations** specify ***how*** parent variables **combine** to produce child variables under uncertainty.

A key **design principle** is that **nodes** and **edges** are *grounded* in the *entities* and normalized *events* produced by the Data and Knowledge planes. This avoids "toy models" that cannot be operationalized. It also enables **traceability**: when the model asserts "A caused B," the system can point back to the specific evidence, transformations, and parameter settings that produced that inference.

4.5.3.2 Causal Graphs / Structural Causal Models based on Dependency Structure

Dependency *structure* provides a pragmatic starting point for building structural causal models (SCMs). Directional **dependencies** (service A calls service B, identity system authorizes application access) suggest plausible causal directions, while **criticalities** identify **which dependencies matter most** to mission outcomes (ref. the earlier mention of "*cyber key terrain*"). **Contingencies** capture **alternate paths** and **degraded modes** that **change the causal topology** under stress (failover, manual processes, segmented networks).

In practice, organizations maintain **a *portfolio* of causal graphs** rather than a single monolith: graphs by *business function*, by *technology domain* (identity, endpoint, network), and by *threat scenario*. The architecture should support **composing** these graphs, scoping them to the current incident context, and selecting the right level of granularity for the decision at hand... fine-grained for root-cause isolation, coarser for executive impact assessment and prioritization.

4.5.4 The Reasoning & Decision Plane

The **Reasoning & Decision Plane** turns **causal models** into **operational *outcomes***.

4.5.4.1 Based on the 5 What Imperatives

Phenomenati's "**5 What Imperatives™**" define a disciplined workflow: start by inferring the **current state** and **likely causes**, translate that into **business impact**, reason about **propagation**, choose **actions** under constraints, and finally **explore counterfactuals** to test whether interventions will work. This plane is where ***uncertainty*** is handled explicitly, and where **explanations** are produced for humans who must approve or execute decisions.

Architecturally, this layer is responsible for **orchestrating inference** across *multiple causal graphs*, **reconciling competing hypotheses**, and producing **decision artifacts** that are auditable: what *evidence* was used, what *assumptions* were applied, what *alternatives* were considered, and what *tradeoffs* were made.

4.5.4.2 What?

The **"What?"** imperative focuses on **inference**: determining the **current state** of the environment and the most plausible **causal explanations** for observed evidence. In a cyber setting, this includes **identifying** active incidents versus benign anomalies, **attributing** activity to compromised identities or workloads, and **isolating** the initiating conditions that made the event possible (misconfiguration, missing control, exploited vulnerability, human error).

The causal model's role is to **separate *correlation* from *causation*** under incomplete observation. It can **integrate** disparate evidence streams, **prefer** explanations that respect known dependencies and constraints, and **quantify** *confidence*. Importantly, it should surface **what is *unknown*** or weakly supported so analysts can target data collection and reduce uncertainty efficiently.

4.5.4.3 So What?

The **"So What?"** imperative translates *inferred state* into ***impact***. This includes **technical risk** (exposure, persistence likelihood, data at risk) and **business/mission risk** (service outage, safety implications, regulatory exposure, contractual penalties, reputational harm). **Impact reasoning** relies on the supporting Context layer: **without *mission mapping* and *dependency chains*, the system cannot distinguish nuisance events from existential ones**.

Causal impact assessment also supports **prioritization**. By estimating which upstream causes drive the most downstream harm, the system can recommend where to spend scarce response time (again, "*cyber key terrain*")... for example, whether to focus on identity containment, endpoint isolation, or restoring a critical dependency that is amplifying the incident.

4.5.4.4 What Else?

The **"What Else?"** imperative addresses **propagation**: the likely *next effects* if conditions remain unchanged. In causal terms, it asks how interventions... or the lack of them... will transmit through dependencies to create **secondary failures**, **lateral movement** opportunities, or **cascading service impacts**.

This is where *structural* knowledge pays off. A well-modeled dependency graph (e.g., a "world model" for the domain) can **forecast** plausible blast radius, **identify** critical choke points, and **expose** and **highlight** "hidden" couplings such as shared identity providers, shared CI/CD pipelines, or shared third-party services. The output should be ***probabilistic*** and **scenario-based**, allowing teams to plan for both the *most likely* and the *most damaging* evolutions.

4.5.4.5 Now What?

The **"Now What?"** imperative turns *reasoning* into **policy-compliant *action***. The system proposes **response options** (contain, eradicate, recover, monitor) along with the **constraints** that govern them: e.g., required approvals, change windows, safety and reliability limits, and legal/privacy restrictions. **Actions** should be framed as **interventions** on the causal model: "*If we reset these credentials and block this path, we reduce the probability of lateral movement by X.*"

This plane should also manage **reversibility** and **operational risk**. Not every technically effective action is acceptable if it interrupts mission delivery. Therefore, **recommendations must include *tradeoffs***, **rollback plans**, and a clear **mapping** from **action** to expected **causal effect**, so decision-makers can choose the least disruptive path that still reduces risk.

4.5.4.6 What If?

The **"What If?"** imperative uses ***counterfactuals*** and ***simulation*** to test candidate actions before committing. **Counterfactual reasoning** asks: given what we observed, what would have happened if a control had been in place, or if we take a specific action now? **Simulation** allows the organization to evaluate alternative playbooks, estimate time-to-recovery, and anticipate side effects.

Operationally, this supports *both* **immediate incident response** and **long-term hardening**. Teams can evaluate which control **investments** *most reduce* mission risk, which **dependencies** *create* fragile *single points of failure*, and which **policies** create

unintended exposure. The outputs should be recorded as **decision evidence** so that post-incident reviews can **compare *predicted* outcomes against *actual* outcomes** and continuously *refine* the causal models.

4.5.5 The Business / Command Plane

The **Business/Command Plane** is where *causal insights* become *organizational action*.

4.5.5.1 Informing Decisions

Reporting, dashboards, and narratives **translate** technical **findings** into **decision-ready artifacts** for different audiences... analyst, incident commander, CIO/CTO, risk committee, legal counsel. ***Static* reports** support **audit** and **after-action reviews**; ***dynamic* dashboards** support *live operations* and *changing conditions*.

Because these artifacts can contain sensitive operational and personal data, **information classification** and **access controls** are ***integral*** to the plane rather than an afterthought. The architecture should support **least-privilege views**, **compartmentalization by mission**, and **consistent handling rules** across data, knowledge, model outputs, and narrative products so that the organization can collaborate without *overexposing* sensitive evidence.

4.5.5.2 Decision Makers

This plane explicitly recognizes that **different stakeholders *optimize* for different outcomes**. **Business/mission operations** prioritize *continuity* and *outcomes*; **security/privacy operations** prioritize *containment* and *data protection*; **governance and risk operations** prioritize *acceptable risk* posture and *accountability*; and **compliance/legal operations** prioritize *adherence to regulatory and contractual obligations* and defensible process.

A mature Reference Architecture ***routes*** the **right decisions** to the **right authorities** with the **right context**. That means *role-aligned* metrics, clear *escalation* thresholds, and *traceable rationales* that connect actions to evidence and policies. Done well, the Business/Command Plane closes the loop: decisions and their outcomes feed back into playbooks, priors, and model updates, steadily improving both operational performance and organizational learning.

4.5.6 The Governance Guardrail

A **Governance** function for this 5-layer Reference Architecture is the "operating system" that makes the whole stack **trustworthy**, **repeatable**, and **defensible**... especially when Causal AI outputs are used to ***justify* decisions under uncertainty**. It establishes the decision rights, accountability, and policies that control how **data** is collected, how **knowledge** is curated, how **causal models** are *constructed* and *validated*, how **reasoning** is executed, and how **results** are communicated to decision makers.

Practically, Governance defines **who** can introduce or modify assets/identities, logging/telemetry sources, contextual business/mission mappings, and cyber operations artifacts (threat intel, playbooks, notebooks) in the **Data Plane**... and sets *enforceable* **requirements** for data quality, retention, labeling, access controls, and chain-of-custody. It also mandates how "***context***" is represented so that evidence is not detached from the mission/business functions and dependency realities it is meant to protect.

In the **Knowledge Plane**, Governance formalizes the *information architecture*: schema/ontology stewardship, evidence provenance standards, evidence weighting rules, and normalization conventions so that events and features remain comparable over time and across teams. This is where Governance **prevents** the common failure mode of "***multiple truths***" by insisting on controlled vocabularies, consistent entity resolution, and explicit provenance and confidence scoring for expert inputs and learned features. It also establishes an "***expert knowledge lifecycle***" (priors, constraints, contingency assumptions) that is auditable and reviewable... so **changes** in **assumptions** can be traced to a **rationale** (e.g., updated dependency information, a new adversary TTP, a compliance mandate, or a shift in mission criticality).

In the **Causal Model Plane** and **Reasoning & Decision Plane**, Governance becomes **model risk management** and **decision governance**. It sets standards for model structure (what nodes/edges are allowed, how structural

equations are specified, how directional dependencies and contingencies are justified), version control, validation criteria, and approval gates for model updates... especially when models drive “Now What?” policies or “What If?” counterfactual simulations. It also defines guardrails for interventions: permitted action spaces, constraints and restraints (legal, privacy, operational), human-in-the-loop requirements, and escalation thresholds when uncertainty is high or impacts are severe.

Finally, in the **Business/Command Plane**, Governance ensures outputs are **fit for purpose**: reporting **standards**, narrative **integrity** (what can be claimed vs. what is inferred), information **classification** and **dissemination** rules, and measurable **accountability** for **decision outcomes** across business/mission ops, security/privacy ops, governance & risk, and compliance/legal. The net effect is that *every* artifact... data, knowledge, models, and decisions... remains **traceable, explainable, and controllable** from origin to action.

4.5.7 The Continuous Improvement Guardrail

A **Continuous Improvement** function for this 5-layer Reference Architecture is the **disciplined feedback loop** that keeps the architecture ***aligned*** with **reality** as the environment, the enterprise, and the threat landscape change. In Causal AI terms, it is how we continually reduce the gap between the **system we assume** and the **system that actually behaves**. This function runs across all five layers by **instrumenting** performance, capturing outcomes, and turning operational learning into controlled updates: e.g., refining telemetry coverage and data quality in the **Data Plane**, tightening ontologies and normalization in the **Knowledge Plane**, recalibrating priors/constraints and revising dependency assumptions, and updating the **Causal Model Plane** (graph structure, parameters, and contingency pathways) based on observed evidence. It ensures that “5 What” reasoning is not a one-time analysis but a living capability... where **inference accuracy**, **impact estimates**, **propagation predictions**, **policy recommendations**, and **counterfactual simulations** are *continuously* evaluated against real outcomes.

Critically, Continuous Improvement formalizes “*learning from operations*” and “*learning from incidents*” without destabilizing the system. It defines what gets measured (model drift, data drift, false positive/negative rates, time-to-detection, time-to-decision, action efficacy, mission impact avoided/incurred), how lessons learned are captured (analyst notebooks, post-incident reviews, playbook deltas), and how updates are safely promoted (staging, A/B comparisons, rollback plans, and governance-approved releases).

In the **Reasoning & Decision Plane**, it uses decision outcomes to *tune* policy constraints and action thresholds; in the **Business/Command Plane**, it connects these changes to stakeholder objectives and risk posture so improvements are judged by operational value, not just technical metrics. Done well, Continuous Improvement is what keeps the architecture from becoming either ***brittle*** (overfit to old conditions) or ***chaotic*** (changing too fast to trust), while steadily *increasing* **decision confidence**, **response quality**, and **mission resilience**.

5 Answering the 5 What Imperatives with DAG Operations

As described in the previous section describing a representative reference architecture for Cyber Situation Awareness, a causal graph becomes *operational* when it supports *repeatable* **query patterns**. Table 1 maps each What Imperative to representative DAG operations and typical outputs.

Imperative	Primary DAG operations	Representative Outputs
What?	Bayesian inference; root-cause ranking; evidence fusion	Incident hypothesis; likely causes; confidence + provenance
So What?	Impact propagation over dependency DAG; expected loss / utility	Mission impact; risk posture; prioritization rationale
What Else?	Forward propagation; effect decomposition; scenario enumeration	Predicted follow-on actions; likely affected assets; uncertainty bands
Now What?	Intervention planning; constrained optimization; value of information	Action recommendations; playbook selection; monitoring tasks
What If?	Counterfactual simulation; sensitivity analysis; wargaming	Tradeoff curves; if-then outcomes; robustness under uncertainty

Table 1. Mapping the 5 What Imperatives™ to DAG-enabled Reasoning Patterns.

6 Use Case: Cyber Situation Awareness (CSA)

CSA focuses on building and maintaining an accurate, shared understanding of cyber conditions across an environment. A DAG-based system supports CSA by *linking* low-level **observations** (telemetry) to higher-level **hypotheses** (incidents, campaigns, *impacts*) in a way that is explainable and updateable.

6.1 CSA: A Representative Causal Graph

Figure 2 illustrates a simplified cyber causal DAG, included here strictly to illustrate the concept. In operational systems, the graph is typically a modular compilation of multiple smaller subgraphs:

- separate subgraphs represent identity, endpoint, network, cloud, and mission dependencies,
- and are composed and integrated through shared variables.

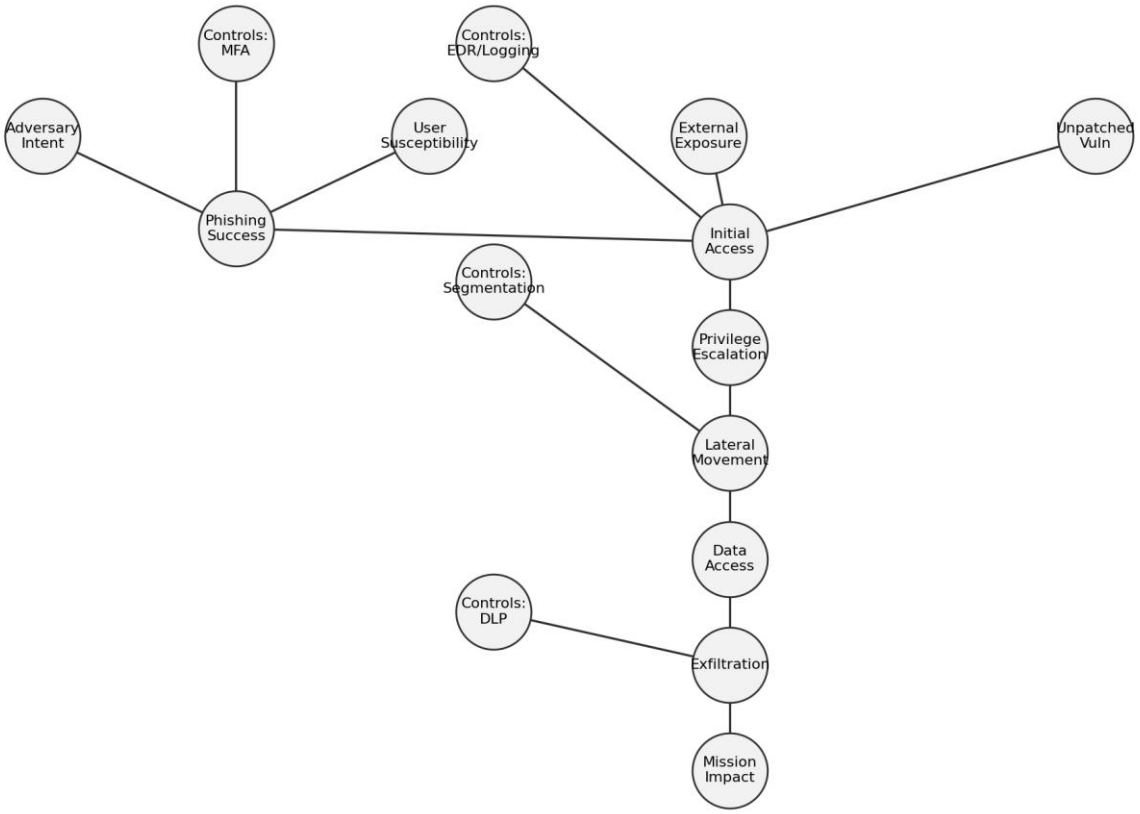

Figure 2. Example DAG linking exposure, controls, adversary actions, and mission impact.

6.2 CSA Outputs Aligned to the 5 What Imperatives

A CSA implementation can be engineered (as the pathfinder DECYSIS™ project demonstrated over 15 years ago) so that every dashboard and narrative maps to one of the five imperatives. Examples include:

- **What**?
  - Current incident **hypotheses** with **confidence** and **causal attribution** (e.g., likely initial access vector).
- **So What**?
  - **Impact** estimates tied to context in the form of mission owners and service ***dependencies*** (not just alert counts).

- **What Else**?
  - Predicted **spread pathways** and likely **next actions**, enabling proactive containment.
- **Now What**?
  - **Recommended actions** with rationale, prerequisites, and estimated risk reduction.
- **What If**?
  - Pre-approved **response options *evaluated*** under different assumptions (e.g., attacker dwell time, detection coverage).

7 Use Case: Full Spectrum Cyber Operations

Full spectrum cyber operations span preparation, defense, active protection, and (where legally authorized) offensive effects. Across that spectrum, the operational need for reliable cyber situation awareness is consistent, connect:

- **actions** to *effects*,
- **effects** to *objectives*, and
- **objectives** to ***risk***.

DAG-centric models help by making those **cause-effect assumptions** *explicit* and *testable*.

7.1 Planning and Assessment (Authorized Operations)

In campaign planning, DAGs can encode hypothesized dependencies such as:

- **access** prerequisites,
- **control** interactions,
- **second-order impacts**, and
- collateral **risk pathways**.

This enables a disciplined approach to:

a) selecting **actions** expected **to achieve desired effects**,
b) estimating **uncertainty**, and
c) identifying what **evidence** would **confirm** or **refute** key **assumptions**.

Importantly, this whitepaper focuses on **decision support** rather than **operational tradecraft**. The same modeling principles apply to defensive actions (e.g., hardening, deception, isolation) and are generally safer to operationalize.

7.2 Measuring Effects and Learning Over Time

DAGs support **after-action learning** by separating metrics that indicate *correlation* from metrics that indicate *causal effect*. For example, a reduction in alerts after deploying a control may reflect reduced visibility rather than reduced compromise. Interventional evaluation - where feasible - helps detect these failure modes.

8 Use Case: National Security Decision Support

National security settings amplify **three challenges**:

- **scale** (many interconnected systems),
- **heterogeneity** (multiple domains and data types), and
- **governance** (classification, privacy, and policy constraints).

DAG-centric Causal AI can help ***integrate*** **cyber signals** with **mission context** to support decisions under *uncertainty*, while producing audit artifacts suitable for oversight.

8.1 Multi-domain Dependencies and Cascading Risk

Cascading impacts often emerge through dependencies that are outside a single, traditional SOC's visibility:

- supply chains,
- third-party services,
- physical infrastructure, and
- organizational processes.

A dependency DAG – aligned to NIST CSF concepts – enables structured '**So What**' analysis by propagating potential consequences from compromised components to mission outcomes[13].

8.2 Incident Response Governance

NIST SP 800-61 Rev. 3 emphasizes integrating incident response into cybersecurity risk management practices. A *causal model* can provide the connective tissue between **detection**, **response actions**, and **risk outcomes** by representing how controls and response measures *influence* incident progression[14].

[13] https://csrc.nist.gov/pubs/cswp/29/the-nist-cybersecurity-framework-csf-20/final
[14] https://csrc.nist.gov/pubs/sp/800/61/r3/final

9 Engineering Considerations

9.1 Time, Feedback, and 'Acyclic' Modeling

Real systems have **feedback** (e.g., defenders respond and attackers adapt). DAGs handle this by **modeling *time* explicitly**: unroll the system into time slices (a dynamic Bayesian network or time-indexed SCM) so that causal influence flows *forward* in time, preserving acyclicity at each slice.

9.2 Data Quality and Missingness

Causal models are only as trustworthy as the *assumptions* and *data* supporting them. In cyber telemetry, **missingness**[15] is common and informative (e.g., an agent is disabled). Treat missingness as **a first-class variable** in the DAG so that *inference* does not implicitly *assume* 'missing at random'.

9.3 Adversarial Pressure and Model Integrity

Cyber models operate in adversarial environments. Protect the knowledge assimilation pipeline with:

- strong **provenance**,
- **signed** data sources,
- **anomaly monitoring** for **model *drift***, and
- **separation** between **training** and **operational** *inference*.

Treat model updates as *governed* changes, not automatic background processes.

9.4 Standards Alignment

Interoperability improves adoption. Practical systems often align variables and evidence to existing standards and knowledge bases such as MITRE ATT&CK[16] for tactics/techniques, and STIX/TAXII[17] for threat intelligence exchange.

[15] https://www.r-causal.org/chapters/15-missingness-and-measurement
[16] https://attack.mitre.org
[17] https://www.oasis-open.org/2021/06/23/stix-v2-1-and-taxii-v2-1-oasis-standards-are-published/

10 Validation and Metrics

Evaluate DAG-centric Causal AI on more than alert accuracy. Recommended metric families include:

- **Decision *Usefulness***:
  - time-to-decision,
  - confidence calibration, and
  - operator workload reduction.
- **Causal *Plausibility***:
  - stability under environment changes;
  - sensitivity to confounding assumptions;
  - ablation studies.
- **Action *Impact***:
  - estimated risk reduction from interventions;
  - false-positive vs. missed-impact tradeoffs.
- **Governance *Readiness***:
  - audit completeness,
  - provenance coverage, and
  - reproducibility of major decisions.

11 Practical Adoption Roadmap

11.1 Phase 0 - Schema + Evidence Governance

Define the variable ontology, data contracts, and provenance model. Start with a narrow, high-value mission slice.

11.2 Phase 1 - Baseline DAG + Expert Priors

Encode known dependencies (assets, identity, controls, mission mapping). Use expert priors to bootstrap reasoning.

11.3 Phase 2 - Supervised Mechanism Models

Attach supervised models to key mechanisms (e.g., compromise progression). Validate against historical cases.

11.4 Phase 3 - Unsupervised Discovery + Drift Monitoring

Add unsupervised components to detect regime shifts and propose structure updates under governance.

11.5 Phase 4 - Intervention Optimization + Counterfactuals

Operationalize 'what now' and 'what if' with constrained planning, simulation, and after-action learning.

12 Conclusion

DAG-centric Causal AI provides a disciplined approach to assimilating knowledge from machine learning and human expertise into a decision-support system that is explainable, auditable, and aligned to mission outcomes. By mapping causal inference operations directly to the 5 What Imperatives™, organizations can build systems that do more than detect – they ***reason***, **prioritize**, **recommend**, and **simulate**. In cyber situation awareness, cyber operations, and national security, that shift from correlation to causation is a decisive advantage.