

The Patches Are the Evidence

Convergence on DSRP-484 Across System Dynamics, Network Theory, and UML

Derek Cabrera ^{a,@}

a: Cabrera Research Lab

@: dac66@cornell.edu

ABSTRACT: This paper synthesizes three completed studies that applied the same preregistered structural-remainder test to three independently developed modeling traditions. System Dynamics returned sixty of sixty constructs without residual; Network Theory returned thirty-five of thirty-five; and UML returned a closed candidate set of one hundred forty-two metaclasses, with no exclusions and no residual among the two hundred seventy-one novel structural contributions those metaclasses make. Across 237 constructs, none required a fifth Pattern, ninth elemental role, fifth dynamic, or other irreducible organizational structure outside DSRP-484. The three run papers contain the construct-level evidence; this synthesis tests what can be inferred from their convergence. Two findings become visible only when the runs are read together. The first is forward-facing: three fields with no shared vocabulary—stocks and flows, nodes and edges, metaclasses and associations—resolve to the same four Patterns and eight elemental roles. The constructs that resisted decomposition most strongly did so for the same reason: each was a constraint placed on a Pattern rather than a new Pattern. The second finding is backward-facing and addresses the possibility that a flexible calculus can redescribe anything after the fact. When a formalism omits an elemental role from its native grammar, the missing organizational work reappears in auxiliary constructs, attributes, rules, or procedures. System Dynamics supplies one emblem: the cloud. Network Theory supplies six load-bearing cases. UML supplies roughly seven hundred countable representational expenditures, including eighty-three derived properties, eleven hand-built negations, four surrogate metaclasses for a Link the metamodel does not contain, and four hundred forty-nine constraint rules. These features were not necessarily designed to repair DSRP-defined omissions and may also serve computational, syntactic, or implementation purposes. The narrower claim is structural: they perform organizational work the native grammar does not perform directly. Together, the findings support more than retrospective containment: omissions in a formalism's native grammar predict the location and function of its later representational expenditures. The paper derives six prospective predictions and specifies their failure conditions through four connected tests—I/O, composition, ablation, and patch—applicable to any formalism.

KEYWORDS: DSRP | DSRP-484 | O-Theory | structural containment | convergence | formalism bias | representational debt | system dynamics | network theory | UML | Love Reality Loop

Cite this paper: Cabrera, D. (2026). The patches are the evidence: Convergence on DSRP-484 across system dynamics, network theory, and UML. *Journal of Systems Thinking*, 6(1), 1–18.
<https://doi.org/10.54120/jost.v26i1.dc6cest6>

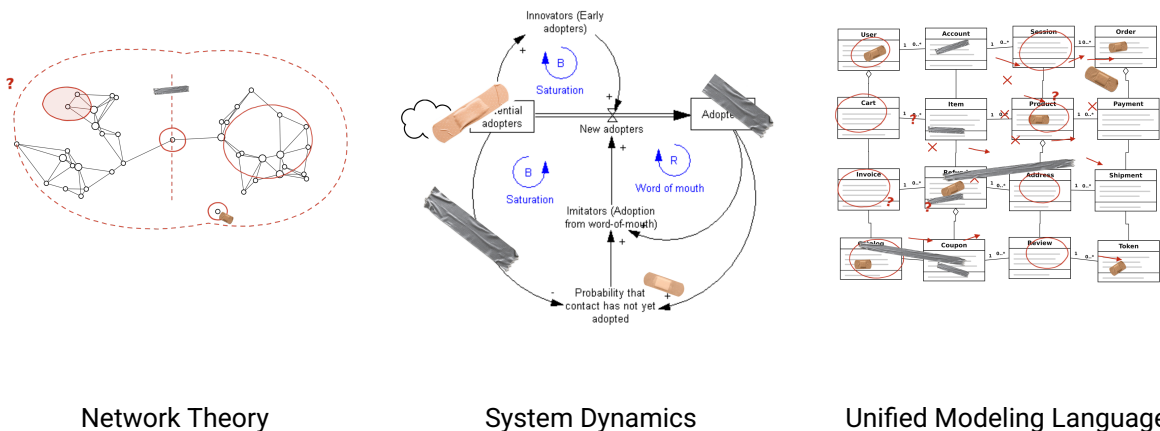
1. The question

Three fields were asked the same question, one after another, under a rule fixed before any of them was examined: Does anything in this field's own vocabulary require an organizational structure that Distinctions, Systems, Relationships, and Perspectives cannot build, or can every construct be composed from DSRP-484 without residual? System Dynamics answered first: sixty constructs, sixty compositions, no residual [1]. Network Theory answered second: thirty-five of thirty-five, no residual [2]. UML answered third, from a machine-readable specification rather than a prose glossary: one hundred forty-two metaclasses, no exclusions, no residual [3]. This paper asks what it means that all three gave the same answer. The fields did not converge on DSRP; DSRP is what they were already converging on.

That conclusion is easy to write and equally easy to dismiss. A calculus capable of *redescribing* anything after the fact is not noteworthy. Taken alone, the three successful compositions—System Dynamics, Network Theory, and UML—are consistent with the claim that DSRP-484 contains their Organization. But the same results are also consistent with a language flexible enough to accommodate whatever it is handed. Composition alone cannot distinguish structural containment from retrospective flexibility. The second body of evidence can: omissions in each formalism's native grammar predict the organizational work performed by its later patches. If the four Patterns (D,S,R,P), their 8 paired elements (*i-o*, *p-w*, *a-r*, *v-p*), and four dynamics (Equality, Coimplication, Simultaneity, and Enclosure) name universal, ontological (real) organization as O-Theory claims, then a formalism that leaves one of those out of its grammar should keep running into the structure it cannot say, and should keep building something else to cover for it.

That prediction is checkable against the historical record of all three fields, and the record is not ambiguous. System Dynamics has been drawing the not-System since Forrester and calling it a cloud. Network science has rebuilt the whole from relations four separate ways and named its missing boundary the boundary specification problem. UML hand-builds negation eleven times and writes four hundred forty-nine rules to police structure its own grammar cannot hold. We call these “patches”: bandage-and-duct-tape solutions to structural absences.

Figure 1. Convergence and patches. Three formalisms drawn as three Perspectives on one organizational structure. Each covers a region of it natively, and each bolts an extension outward toward a region it does not natively cover. The regions are schematic and do not imply measured proportions. Network Theory covers identity and a generic relation and extends toward wholes, boundaries, relational internals, and framing. System Dynamics covers quantified wholes and directed relations and extends toward everything it cannot accumulate. UML covers identity, part-whole, and relation across a very large vocabulary and extends toward negation, framing, and relations treated as objects.



This is a short paper by design. The three papers [1–3] carry the evidence and the construct-by-construct rulings; nothing is re-argued here, and no number appears that is not cited back to the run that produced it.

What is new is the pair of findings the runs support only together: the forward direction, in which three unrelated vocabularies land on the same organization, and the backward direction, in which the places they had to "patch" line up with the places they left something out.

2. The instrument

The reader who wants the full instrument is referred to the full description in the three papers [1–3]. This section gives the minimum needed to make the comparison legible and reproducible.

These three fields (NT, SD, and UML) were not randomly sampled, nor were they chosen to represent all formal knowledge. They were chosen because they meet two important conditions: they are *consequential*, and they are *sufficiently specified* to analyze. System Dynamics, Network Theory, and UML are widely taught, widely used, and highly influential ways of organizing reality. The more influence a theory or framework has—over research, institutions, technologies, and decisions—the more important it is to test what it can represent, what it leaves out, and what work it must perform to recover what it leaves out.

But influence alone is not enough. A framework must also contain enough explicit machinery to be examined. Broad philosophical schemes may be suggestive, but if their constructs, rules, and representational commitments are not sufficiently specified, they cannot be reliably extracted, classified, counted, or tested. Their ambiguity makes almost any retrospective interpretation possible and leaves claims about their structure with weak construct validity. These three traditions are unusually useful because their organizational commitments are materially instantiated in formal primitives, equations, diagrams, metamodel elements, rules, and constraints. They give the instrument something definite to test—and other researchers something definite to inspect, challenge, and replicate.

DSRP must meet the same standard. It is not four broad dictionary categories into which constructs are

sorted by resemblance. DSRP-484 specifies four formally defined patterns, eight elemental roles, four dynamics, six emergent mechanics, and rules governing their recursive composition [4,5]. Embedded in O-Theory [6], it has been studied by its founders and independent researchers, with evidence bearing on its ecological and convergent validity [7] and explicit conditions under which it can be falsified [8]. Its definitions generate inclusion and exclusion criteria, structural predictions, reproducible coding decisions, and identifiable residuals. Treating DSRP as four loose conceptual buckets therefore substitutes a simpler theory for the theory actually being tested. The question is not whether a construct can be made to *sound like* a distinction, system, relationship, or perspective. The question is whether its organization can be decomposed into the specified elements and dynamics without remainder. That is a formal and empirical claim: it can be operationalized, tested, replicated, and falsified.

All three traditions make substantial use of graph-like representations, but that does not explain the findings. A node-edge grammar makes identities and generic relationships relatively easy to represent; it does not thereby supply otherness, part-whole membership, action-reaction interiors, or perspectival framing. Indeed, the Network Theory run found those roles missing from the primitive node-edge grammar and returning through higher-order constructs. Nor are the three traditions reducible to nodes and edges. System Dynamics adds stocks, flows, accumulation, feedback, boundaries, and delays. UML extends typed elements and relationships through containment, multiplicity, properties, constraints, behaviors, interactions, deployment, and extensibility. Their structural richness makes them demanding tests of DSRP, not automatic confirmations of it. All three traditions are graph-like because graphs are one of the dominant ways formal models externalize organization [9,10]. That shared feature is therefore part of what is being investigated, not an accidental flaw in the sample. Graph-likeness might explain the easy recovery of identities and generic relationships. It cannot explain the recovery of all eight elemental roles and four dynamics.

These are therefore consequential and testable first cases, not a representative final sample. Three independently developed traditions cannot establish universality, and there are many more to test. Non-graph-native formalisms, including formal logic and thermodynamics, as well as formal “ontologies”¹ such as the Bunge-Wand-Weber ontology (BWW), the Descriptive Ontology for Linguistic and Cognitive Engineering (DOLCE), and the Basic Formal Ontology (BFO), provide especially valuable next cases and are identified prospectively in Section 7.

The candidate set for each run was drawn only from documents the field itself wrote as statements of its own vocabulary or foundations. This avoids straw-manning frameworks. The steel-man of the framework is provided by the most respected and authoritative sources in the respective fields. For the two prose-based runs, three kinds of nominator were used: a source giving the field’s terminology, a source in which the field reflects on its foundations, and a source by or representing its founder. For System Dynamics these were Ford’s official glossary [11], Richardson’s reflections [12], and Forrester’s account, as documented in the System Dynamics run [1]. For Network Theory they were Newman’s canonical survey [13], Barabási’s textbook [14], and Euler’s 1741 paper [15]. For UML the primary nominator was

¹*Ontologies* appears in quotation marks because an ontology must be tested against what exists, not merely against classifications produced by minds. An ontological theory must generate predictions about the structure of reality and survive tests of those predictions across physical, biological, and other mind-independent phenomena. A framework tested only against concepts, categories, language, models, or other products of cognition is being tested epistemologically, regardless of whether it is called an ontology. DSRP-484 is explicitly onto-epistemic: it predicts that the same elemental structures occur both in the organization of reality and in the organization produced by minds, and it is tested separately in both realms. And it is explicitly falsifiable.

the OMG UML 2.5.1 normative Abstract Syntax Metamodel [16], cross-checked against a historical nominator and an adversarial one [3]. The analyst added nothing to any candidate pool.

An entry could be excluded only under five categories declared in advance — numerical or mathematical technique, research practice or workflow, notation artifact, general-language term, and imported theory — and every exclusion was logged with its category. Nothing was dropped silently. One criterion did the sorting: a feature is organizational (O) if changing it changes what entities or relations exist, or how they are bounded, nested, ordered, or framed; it is informational (I) if changing it changes only a value carried on a structure that stays put. The rule is operational rather than intuitive: network edge weight can change while the edge remains, whereas reversing edge direction changes which relational roles exist; SD link polarity can change while the causal relation remains, whereas reversing causal orientation changes the relation; UML multiplicity values are information, whereas the allowable set they populate is part-whole organization. Borderline construct-level rulings remain open to contest and are recorded in the run papers. Candidate generation differs by source, but the comparison does not treat exclusion counts as outcomes. Prose nominators require logged exclusions; UML's normative abstract syntax had already been filtered for structural syntax and permits the candidate set and its 271 novel contributions to be regenerated from the public file. That asymmetry reduces discretion in the UML run rather than weakening the common test. One falsifier stood behind all three runs: a construct whose necessary organizational content cannot be composed from D, S, R, and P with their eight roles and four dynamics.

Formally, for a construct c , O-Theory separates information from Organization, and the test asks whether that Organization falls within what DSRP-484 can generate:

$$X = I \cdot O \quad \text{and} \quad O_c \in \text{Cl}(O_{\text{DSRP-484}})$$

Read plainly: any organized thing (X) is particular information (I) instantiated in a particular Organization (O). The test asks whether the Organization required by a construct (O_c) belongs to the closure of DSRP-484. Here, Cl is the set of structures constructible from its four Patterns, eight elemental roles, and four dynamics through finite recursive composition. The closure may contain structures of unbounded complexity, but every member must have a finite derivation from the specified basis. Any structure that cannot be derived from that basis falls outside the closure. A single verified irreducible residual—a fifth Pattern, ninth elemental role, fifth dynamic, or other necessary structure—falsifies the universal sufficiency claim.

The distinction becomes clear when information changes while Organization remains fixed. Consider a System Dynamics stock whose informational label is *Population*. The stock is the part-whole structure: an accumulation treated as a whole, composed at a given moment of 1,000 informationally identified units. Change the label from *Population* to *Inventory*, and the information identifying the whole changes while the stock structure remains. Change the quantity from 1,000 to 1,200, and the information identifying the number of parts changes: the stock now contains 1,200 units, but they remain parts organized within the same whole. Changing the Information does not change the fundamental structure of the model. Remove the stock structure altogether, and the whole-part Organization disappears. Next, consider a causal link whose informational endpoints are *Population* and *Births*. Change its polarity from *positive* to *negative*: information carried by the link changes, but the directed Relationship remains. Reverse the link so that *Births* acts on *Population*, and the action-reaction roles change.

The same subtraction works in Network Theory: an edge weight can change while the edge remains, but reversing a directed edge changes which node occupies which relational role. Information varies on Organization; changing Organization changes what entities or roles exist, how they are related, or where they belong. The residual question is whether any of these structures requires a fifth Pattern, ninth elemental role, fifth dynamic, or anything else outside DSRP-484. Because each counterfactual can be reversed and repeated, the separation is a rule a critic can inspect—not a label the analyst can apply without consequence.

The formal falsifier is:

$$\exists c \in C : O_c \notin \text{Cl}(O_{\text{DSRP-484}})$$

Read plainly: there exists ($\exists$) at least one construct (c) in the candidate set (C) whose required Organization (O_c) cannot be generated from DSRP-484. One verified irreducible case falsifies the universal containment claim.

The same rule was applied in all three runs, while candidate generation became progressively less dependent on prose interpretation. An editorial glossary is prose assembled by a committee. A canonical survey is prose written by one authoritative hand. A normative machine-readable specification is a public file that a script can parse and anyone can re-parse. UML therefore sharpens candidate generation: the file fixes the 142 candidates and sorts all 271 novel contributions by OMG’s own declared types—160 relationship ends, 96 primitive values, and 15 enumeration values. No candidate or contribution can be added or removed through retrospective interpretation of prose. Whether the Organization instantiated by those contributions decomposes without residual remains a separate analytical question, examined in the individually argued hard cases. The run that was hardest to curate toward a preferred result came last.

3. Three runs

Table 1. The three executed runs. Each run used field-native sources to generate its candidate set and applied the same structural-remainder test. Exclusions were logged but not counted as candidates. Every included construct decomposed into DSRP-484 without residual. The run papers provide the complete construct-level rulings and exclusion logs; the underlying replication files are available from the authors on request [1–3].

Field	Nominators	Candidates	Exclusions	Result
System Dynamics	Ford glossary (2019); Richardson (2011); Forrester (2007)	60	17 logged	60 / 60, no residual
Network Theory	Newman (2003); Barabási (2016); Euler (1741)	35	10 logged	35 / 35, no residual
UML	OMG UML 2.5.1 normative metamodel; Selic (1998); UFO/OntoUML (adversarial)	142	0	142 / 142, no residual

“No residual” does not mean that every construct turned out to be one of the four Patterns wearing a different hat. That would be the wrong theory, and none of the three runs claims it. It means something narrower and more checkable: after the informational content was subtracted, nothing remained that required a fifth Pattern, ninth elemental role, fifth dynamic, or other irreducible organizational structure.

Many constructs returned as composites—a feedback loop as a System of Relationships, a component as a System recovered from Relationships under a rule, and a UML Property as one organization performing three jobs selected by a flag. Composite is not a demotion. Molecules did not become less real when the elements were named, and none of the three fields loses a single empirical result through this analysis. What changes is the level at which their vocabularies can be explained, compared, and taught. UML's empty exclusion log reflects the structure of its nominating source rather than an omitted screening step. The prose sources used in the System Dynamics and Network Theory runs contain workflow terms, software terms, numerical procedures, and other entries that the structural test must set aside. UML's normative abstract syntax, by contrast, contains abstract-syntax metaclasses by design. Its source scope had therefore already removed the kinds of extraneous entries excluded from the prose-based runs [3].

The UML run also differs in how its candidates were ruled. The System Dynamics and Network Theory runs ruled every construct individually. The UML run individually ruled 28 constructs: the nine metaclasses from which the normative metamodel is assembled, plus 20 selected before ruling solely because each appeared capable of breaking the claim, with one construct belonging to both sets. The remaining candidates were ruled through the novel contributions each metaclass adds beyond those inherited from its ancestors. Across the 142 candidates, the file declares 271 such contributions: 160 relationship ends, 96 primitive values, and 15 enumeration values. Those three types exhaust the novel contributions declared in the normative file; they do not by themselves establish that no DSRP residual exists. Rather, they make the candidate set and its contributions enumerable and reproducible. A residual could occur within any contribution if its necessary Organization could not be generated from DSRP-484. Because every contribution is exposed to that test, the count—not selective argument—is what puts the UML claim at risk [3].

4. Different Vocabularies Share the Same Organization

System Dynamics speaks of stocks, flows, feedback, and delay. Network Theory speaks of nodes, edges, degree, and centrality. UML speaks of metaclasses, properties, associations, and generalizations. Forrester was modeling industrial supply chains in 1961, Euler was answering a question about a walk across seven bridges in 1735, and the Object Management Group was standardizing software notation in the 1990s. These vocabularies were developed for different purposes and do not translate directly into one another. Yet when their informational content is subtracted, their constructs repeatedly resolve into the same four Patterns and eight elemental roles.

Some correspondences are exact. A feedback loop in System Dynamics and a directed cycle in Network Theory are both chains of directed Relationships composed back onto their origin. An SD generic structure and a network motif both hold Organization fixed while allowing different information to occupy it. Other correspondences occur at the elemental rather than the complete composite level. Stocks, network components, UML Packages, and composite aggregations are not identical constructs, but each requires part-whole Organization. Reinforcing loops and preferential attachment are not the same formal mechanism, but each produces self-amplification through recursively related action and reaction. The claim is not that the fields secretly contain the same vocabulary. It is that their different vocabularies repeatedly build with the same universal elemental *Organization*.

That is the forward result. Across 237 constructs, three independently developed formalisms required no fifth Pattern, ninth elemental role, or fifth dynamic. The constructs remain different because they carry different information and combine the elementals differently. What converges is the organization from which they are built.

The result also creates a second question. If these elementals recur across all three fields, what happens when a field's primitive grammar does not represent one of them directly? The answer appears in the additional machinery each field builds around its primitives.

5. Patches

Suppose phenomena in the world repeatedly require four elemental forms of Organization, but a formalism includes only two in its primitive grammar. As the formalism is made more faithful to those phenomena, it has two choices: expand its primitive grammar or reconstruct the omitted forms indirectly through special constructs, attributes, rules, and procedures. If its primitive grammar remains fixed, the second path is unavoidable. The result is an innovation-and-discovery *patchwork*: the formalism accumulates local solutions that perform organizational work its primitives cannot perform directly.

DSRP-484 makes that prediction specific. If a formalism directly represents identity but not otherness, otherness should return through negations, exclusions, boundaries, or complementary labels. If it represents connected entities but not nested membership independent of connection, part-whole Organization should return through components, groups, boundary rules, or objectification. If it represents a relation as a line but not its action and reaction roles, those roles should return through direction, inverse properties, endpoints, derived lookups, or auxiliary objects. If it lacks a general point-view structure, framing should return through flags, profiles, visibility rules, enumerations, or other devices that select how the same structure is read. And if its grammar hard-separates nodes from edges—preventing a Relationship from simultaneously functioning as a Distinction, System, and Perspective—the missing dynamics should return through reification, association classes, auxiliary nodes, hyperedges, or other devices that allow one structure to perform several Pattern roles at once.

These additions may serve many practical purposes, and their designers need not have understood them as repairs. A Boolean flag can support efficient execution and carry a missing frame; a visibility modifier can serve encapsulation and specify a point-view restriction; a derived property can provide syntactic convenience and recover a severed relational direction. Multiple engineering functions do not erase the organizational function. The prediction concerns the structural work performed: Organization omitted from the primitive grammar should reappear in higher-order constructs, attributes, rules, or procedures. The test is whether those features perform work the primitive grammar does not perform directly, and whether the omitted roles and dynamics predict where that work recurs across independently developed formalisms.

The patch claim fails if the nominated features do not perform the predicted missing work, or if the same work is already performed directly by the primitive grammar.

The three fields expose that indirect work at different resolutions: one emblem in a compact notation, six load-bearing cases in a scientific formalism, and hundreds of machine-countable expenditures in a large software metamodel. The units are not numerically commensurate, but the structural recurrence is comparable.

System Dynamics: one emblem. The cloud. In a stock-and-flow diagram, a flow that originates beyond the model boundary or disappears past it is drawn with a cloud. The cloud marks an outside the model does not represent internally. Structurally, the cloud marks the other of the modeled System. Every System coimplies a Distinction: $S_{pw} \rightarrow (D_i \mid D_o)$. Read plainly, a whole composed of parts (S_{pw}) requires a boundary specifying what belongs to it (D_i) and what does not (D_o). The cloud draws the D_o . System Dynamics has been drawing both since Forrester, as documented in the run paper [1]. The construct is not wrong; it is under-generalized, because it applies at the scale of the model while DSRP predicts the same identity-other structure for every distinguished object inside it. One emblem, and it is a good one.

In six load-bearing places, Network Theory successfully reconstructs Organization its own primitives cannot represent—and does so without revising those primitives [2].

Betweenness centrality counts shortest paths through a node. It can therefore identify a low-degree bridge that *degree centrality* misses. The bridge either connects two already distinguishable part-whole Systems or is the relation whose removal partitions one network into two such Systems. The primitive node-edge grammar represents it only as another node; betweenness measures the numerical shadow of its higher-order relational and systemic Organization.

The *boundary specification problem* is the field's own name for the unresolved question of which nodes belong within the network and which remain outside it—the missing identity-other boundary, $D_i \mid D_o$, independently identified by sociologists in 1983 [17].

Small-world rewiring collapses distance by adding a shortcut across a ring. The shortcut helps constitute two regions of what the primitive grammar still represents as a single component; remove the shortcut, and that two-System Organization disappears. The calculation depends on a part-whole distinction the node-edge grammar does not explicitly contain.

Scale-free disproportion divides a network operationally into nodes with radically different effects on the whole. Hubs and non-hubs function as two sub-Systems that determine robustness, spreading, and search behavior, yet the primitive grammar provides no grouping structure in which to declare either one. The field calculates the consequences of wholes its notation does not explicitly contain.

The *isolated but contained member* exposes the same omission more directly. A vertex may belong to a represented network while having no path to any other vertex, but a component defined through reachability cannot express that membership independently of connection. The node may be contained in the modeled whole without belonging to any relationally generated component. The case separates part-whole membership from connectivity, while the primitive grammar can recover the former only from the latter.

Finally, the *hyperedge* calls a whole an edge because the grammar has no native primitive for a shared whole or for part-whole membership independent of pairwise connection. Its standard bipartite representation makes the patch visible: the hyperedge becomes a node, its participants become another set of nodes, and membership is reconstructed as ordinary edges between them. The missing whole is objectified, and the grammar remains unchanged. Six cases, six successful patches, and the same primitive grammar throughout.

UML makes the same pattern machine-countable. Its normative metamodel contains no general operator for negation, so negative conditions are expressed through eleven separate mechanisms, including *isCovering*, *isAbstract*, *neg*, and *break*. It contains no general point-view construct, so framing is distributed across 69 Boolean flags, 13 enumerations, visibility and multiplicity rules, and the machinery of *Profile*, *Stereotype*, *Extension*, and *ProfileApplication*. It contains no *Link* metaclass, so four other metaclasses stand in for links and eleven more operate on the runtime link UML does not represent as an object. The specification makes the omission explicit: a link cannot be passed as a runtime value and must instead be identified through its end objects.

The same pattern appears in UML's derived and constrained machinery. Eighty-three derived properties and 32 derived unions recompute information that the primitive grammar does not retain directly, often by recovering the reverse direction of a Relationship. Across the full normative metamodel, 156 of UML's 242 metaclasses carry 449 constraints stating conditions not embodied directly in its abstract syntax. *Property* alone carries eleven constraints while performing three different jobs selected by a flag. And because *ValueSpecification.isComputable* cannot be fully defined in OCL, a conforming implementation is expected to return *true*. Together, these features amount to roughly 700 representational expenditures. The count is a census of representational expenditure, not seven hundred individually demonstrated patches. The stronger patch claim applies where the full sequence is established: UML's native grammar omitted a necessary structure; later machinery appeared and performed the missing organizational work; and the native grammar remained unchanged. The construct-level mappings identify the expenditures for which that sequence can be shown.

The evidence takes a different form in each field and should not be compared as though one cloud, six network cases, and roughly 700 UML expenditures were equivalent units. What is comparable is the structural recurrence. In all three formalisms, Organization absent from the primitive grammar reappears through higher-order notation, constructs, attributes, rules, or procedures. The cloud, the boundary specification problem, the bipartite reconstruction of a hypergraph, UML's missing *Link*, and its constraint machinery existed before this analysis. DSRP did not produce them; it predicts the common work they perform and where similar machinery should appear next. That provides evidence against arbitrary retrospective accommodation. Subsequent fields determine how far the prediction travels.

6. Every formalism is a Perspective

A formalism is a Perspective in the technical sense: a point that frames, and a view that comes back framed. It over-sees the structures it implements and cannot see the structures it lacks. That is not a defect

peculiar to any of these three fields. It is what having a grammar costs, and it applies to DSRP too (Section 9).

What the runs add is that the shape of each field's blindness is now specifiable in advance, from the roles its grammar leaves out.

System Dynamics goes looking for feedback loops, stocks, flows, and nonlinearity, and finds them even where simpler structures would do. It imposes continuous aggregated accumulation on phenomena that are discrete and individuated — every accumulation must be a population, so cash becomes units of currency and morale becomes units of morale. And it holds endogeneity as doctrine, defining outside structure away rather than finding it absent [1].

Network Theory projects every system onto one flat, typeless level. It permits one kind of Relationship, with action and reaction suppressed, and — worse — it defines its Relationships entirely by their relata, so an edge exists as the pair (i, j) and has no identity, no parts, and no capacity to relate or be related to. It has no place at all to declare a Perspective, which is why every importance measure and every partition score smuggles one in as an index [2].

UML buries D_i inside Element and leaves the other pole, D_o , to be hand-built. It presents Relationships generically, builds one role-end, and files the other where neither participant can reach it: of 418 associations in the normative metamodel, 328 have one end owned by the Association itself rather than by either classifier it connects, 90 have both ends held by the things they connect, and none have both ends orphaned. UML consistently names one relational role somewhere; it is the reciprocal role that is disproportionately displaced. And it frames constantly while almost never naming the standpoint from which the framing is done [3].

The sharpest instance in the whole series is Euler's because it is role-level, historical, and checkable. The Königsberg bridge problem asked one question in 1735: can all seven bridges be crossed without crossing any twice? To answer it, Euler needed the land masses to be distinguishable and needed to know which bridges connected which. He needed nothing else, and he threw everything else away [15]. Stated in elemental roles, he kept identity (D_i) and a general relation stripped of its internal roles (R_{gen}), and left out seven: other (D_o), part and whole (S_p, S_w), action and reaction (R_a, R_r), and point and view (P_p, P_v). One of the eight elemental roles survived into the graph, together with a Relationship stripped of its internal structure. That is the chassis Network Theory inherited [2].

Every one of those omissions was appropriate given the question Euler was attempting to answer. The land masses were exhaustively enumerated, so what each was *not* was fixed by the others and D_o needed no separate notation. A walk does not care what a bridge is made of, so S_p and S_w cost nothing. A footbridge is symmetric, so R_a and R_r genuinely bundle. Perspective was not absent but constant—*fixed by the question itself*—and a constant can be dropped from a notation for free. Each omission was correct in Königsberg, and each breaks when the chassis travels. That is exactly what the next three centuries record: extensions that reconstruct combinations of the seven, over and over, without requiring a ninth elemental role.

7. Predictions

The three run papers state many predictions specific to their fields [1–3]. The predictions here apply the same four tests—I/O, composition, ablation, and patch—to formalisms beyond the three completed runs. They state what DSRP-484 predicts those tests will find.

1. **Next-formalism containment.** After its informational content is separated from its Organization, the organizational vocabulary of the next independently developed formalism will decompose into DSRP-484 without residual. Every required structure will fall within the closure of the four Patterns, eight elemental roles, and four dynamics; no fifth Pattern, ninth role, or fifth dynamic will be required.
2. **Omissions predict expenditures.** Before examining a formalism's extensions, its primitive grammar should reveal which elemental roles and dynamics it does not represent directly. Those omissions should predict where additional structural work later appears. Missing *otherness* should predict negations, exclusions, and boundary machinery. Missing nested part-whole membership should predict groups, components, nesting rules, or objectification. Missing action-reaction interiors should predict direction, inverse properties, role-ends, and derived lookups. Missing point-view structure should predict flags, profiles, visibility rules, indices, and other framing devices. A grammar that hard-separates its structural types should predict machinery that allows one structure to perform several Pattern roles at once.
3. **Patches recur as composites.** An omitted role should not return once under one tidy name. Because actual constructs combine multiple elemental structures (as DSRP-484 predicts ontologically and epistemologically), the same omission should be reconstructed repeatedly and in combination with other omissions. Independently invented extensions should therefore prove mathematically or computationally different while performing recurring combinations of the same lower-order organizational work.
4. **Extensions move toward omitted Organization.** As a formalism is extended to represent its target (reality) more accurately, its successful extensions should become more explicit about the structures its primitive grammar omitted. Across successive versions or historically ordered extensions, representation of boundaries, membership, relational interiors, framing, recursion, and multiple simultaneous Pattern roles should increase rather than decrease. The prediction concerns structural trajectory, not whether later formalisms become larger or more complicated in general.
5. **Pareto bias in representation.** The Cognitive Pareto Bias [18] is an epistemological trend, not an ontological law. Formalisms, as products of minds, are unlikely to represent the Patterns or their elemental poles with equal frequency or explicitness. Their vocabularies and notations will likely foreground *identities* over their *others*, *parts* over the *wholes* and memberships they imply, *generic Relationships* over their *action-reaction interiors*, and framed *views* over the *points* from which the framing occurs. Perspective is especially likely to remain implicit. Purpose-built formalisms may also show a compensatory reversal at the Pattern level: because they are designed to make ordinarily neglected Organization explicit, they may become Relationship- or Perspective-heavy. UML is such a reparative case. Yet the Pareto bias should persist within those emphasized Patterns, with one pole made explicit and its co-implied complement left implicit. This prediction concerns what minds selectively make visible in their models, not what exists in

reality. The predicted asymmetries should be measurable in independent candidate sets and machine-readable specifications rather than inferred from selected examples.

- 6. **Composition and ablation produce specific losses.** The Organization of each native or patched construct should be buildable from DSRP-484. Removing a role or dynamic identified as necessary to that composition should then remove a specific capacity of the construct. Removing otherness should eliminate the relevant negation or boundary; removing part or whole should eliminate the relevant membership or accumulation; removing action or reaction should eliminate the directed relational interior; and removing point or view should eliminate the relevant framing. If the construct retains the same Organization after a supposedly necessary structure is removed, that structure was not necessary to its composition.

These are not facile predictions that formalisms tested in the future will somehow look “DSRP-like.” They specify a finite generating basis, the exact roles and dynamics whose presence or absence matters, where omitted *Organization* (O=DSRP-484) should reappear, the structural work its replacements should perform, the direction extensions should travel, the asymmetries independent counts should recover, and the particular capacities that should disappear under ablation. Each prediction has a discrete failure condition. Prediction 1 fails with a single irreducible residual outside the closure of DSRP-484. Predictions 2–4 fail if omitted structures do not predict the location, function, recurrence, and historical trajectory of representational expenditures. Prediction 5 fails if independent counts do not recover the predicted epistemological asymmetries. Prediction 6 fails if the fields’ constructs cannot be composed from DSRP-484—or if an alleged elemental role or dynamic can be removed without a corresponding loss of organizational capacity. The completed runs applied I/O and composition across three fields and supplied the historical evidence examined by the patch test. Ablation adds a specific necessity check on the proposed compositions. Applying the full set of mathematical, historical, count-based, compositional, and ablative predictions to new formalisms will determine how far the result travels.

8. How the Argument Can Fail

The argument rests on four connected tests: *I/O*, *composition*, *ablation*, and *patch*. Together they ask whether *Information* can be separated from *Organization*, whether the remaining structure can be built from DSRP-484, whether each structure named in that composition is necessary, and whether omissions from a formalism’s native grammar predict the work performed by its later patches.

Table 2. The four-part structural test. The four tests operate together. I/O isolates the Organization being tested; composition determines whether DSRP-484 can build it; ablation tests whether the identified structures are necessary to it; and the patch test asks whether omissions from a native grammar predict the organizational work performed by later additions.

Test	Procedure and criterion
------	-------------------------

I/O	Separate Information variables from Organization variables. Information can change while structure remains fixed; changing Organization changes the structure itself.
Composition	Identify a native or patched structure, X . Test whether the Organization of X can be built from DSRP-484.
Ablation	Remove one DSRP-484 structure identified in the composition of X . If X loses a defining organizational capacity, that structure was necessary to X .
Patch	Establish that the native grammar lacked a necessary DSRP-484 structure; patch X appeared later; X performed the missing organizational work; and the native grammar remained unchanged.

Begin with I/O. Take any legitimate construct and separate what it is about—its informational content—from how it is organized. Then test composition: can the Organization that remains be built through finite recursive composition of the four Patterns, eight elemental roles, and four dynamics? If not, containment fails:

$$\exists c \in C : O_c \notin Cl(O_{\text{DSRP-484}})$$

A *residual* is a *counterexample* [8]. It does not need to look like a fifth Pattern in advance. It only needs to require some Organization that DSRP-484 cannot supply. One verified fifth Pattern, ninth elemental role, fifth dynamic, or other irreducible structure is enough. The claim is universal, so one real exception breaks it.

The *composition* must preserve the construct's defining organizational capacities. A composition of a feedback loop without action–reaction Relationships is not a feedback loop. A composition of a hyperedge without part–whole membership is not a hyperedge. If the proposed DSRP structures cannot build the Organization the construct requires, the composition fails.

Ablation checks the composition. Remove an element or dynamic that is compositionally required and ask whether the same construct can still be built with the same organizational capacities. The predicted loss must be specific. Remove action or reaction and feedback fails; remove part or whole and the hyperedge loses the membership structure that makes it a hyperedge. If the construct survives unchanged, that role was not necessary to the proposed decomposition. Across the candidate set, these specific losses test whether each part of DSRP-484 is doing work the others cannot simply replace.

The hyperedge in Network Theory provides a compact example of all four tests. Its Information may vary—the members may be people, authors, proteins, or other identities—while its Organization remains membership in a shared whole. Composition identifies that Organization as a part–whole System: multiple distinguished parts belong to one distinguished whole. Ablation is specific: remove the parts, the whole, or their contains–belongs structure, and the defining capacity of the hyperedge disappears. Network Theory's standard bipartite representation makes the same composition visible using its available grammar. It distinguishes original vertices from hyperedge-vertices, $D(V \mid H)$, and connects them with incidence Relationships that encode membership, $R_{\text{membership}}(V, H)$. Thus:

$$D(V \mid H) + R_{\text{membership}}(V, H) \cong S_{pw}(V, H)$$

For example, if Ann, Bob, and Carlos belong to the same family, represent the family as a node and connect Ann, Bob, and Carlos to it with edges indicating incidence—that is, membership in the family. The representation works precisely because it reconstructs the missing part-whole System indirectly: the whole becomes a node and belonging becomes an edge. The hyperedge therefore also passes the patch test. It appeared after the native node–edge grammar, performs the part–whole work that grammar omitted, and leaves the native grammar unchanged.

The *patch test* carries the same logic into the history of a formalism. Begin with its original grammar and identify the Organization it leaves out. Then examine what the field later built. If DSRP is right, later constructs, rules, and extensions should repeatedly perform the missing work. If those omissions do not predict where the added machinery appears or what it does, the patch argument fails.

The tests do not all apply to every construct in the same way. Consider a minimal two-variable feedback loop. I/O separates the variables' informational content from the loop's Organization. Composition builds that Organization from two distinguished relata, two directed action–reaction Relationships, $A \rightarrow B$ and $B \rightarrow A$, and the part-whole System they form. Ablation then checks the proposed composition: remove D_i and there are no identifiable relata, Relationships or Information variables; remove R_{ar} and there are no directed causal Relationships; remove S_{pw} and the Relationships no longer form one loop. The feedback loop is native SD structure, so it is not itself a patch. The patch test applies where a formalism's later machinery performs Organization omitted from its native grammar.

The four tests organize the combined argument. The three run studies executed I/O and composition; their historical findings supply the evidence evaluated by the patch test; and ablation provides an additional necessity check on the proposed compositions. Failure at any point breaks the relevant claim: failed I/O separation invalidates the test, an irreducible residual breaks containment, failed ablation breaks the proposed composition, and failed historical prediction breaks the patch argument.

9. Limits and Prior Work

Three successful runs do not establish universality. Considered alongside the broader multi-trait, multi-method evidence [7], they strengthen the case, but the universal claim remains open. What these runs establish is convergence across three independently developed formal traditions with different origins, purposes, and vocabularies. Within their candidate sets, all three separated Information from Organization and composed the resulting structures from DSRP-484 without residual. Their findings also identified later machinery performing work omitted from the native grammars. The present synthesis makes that historical pattern explicit through the patch test and introduces ablation as a further check on whether the structures identified in each composition are necessary. The selection limits are stated in Section 2; the predictions and failure conditions are stated in Sections 7 and 8.

DSRP is also a formalism and is therefore subject to the same risk of seeing only what its own grammar makes visible. The answer is not to exempt it. DSRP-484 fixes four Patterns, eight elemental roles, four dynamics, and their rules of composition [4–6,19]. Those commitments determine what counts as Information, what counts as Organization, how a construct must be composed, what should be lost under

ablation, and where omitted Organization should return as a patch. A filing system can accommodate whatever it is handed. A theory says what should be there and gives us a way to show that it is not.

This is not the first attempt to test a modeling language against an ontology. The Bunge–Wand–Weber model has done so since the early 1990s, using the terms construct deficit, construct overload, construct redundancy, and construct excess for failures of alignment between a grammar and an ontology [20]. Opdahl and Henderson-Sellers applied that approach to UML, found construct deficits, and proposed additional metaclasses to cover them [21].

DSRP-484 asks a different and more elemental question. BWW compares the categories supplied by an ontology with the constructs supplied by a modeling language. The present method separates Information from Organization, tests whether each resulting structure can be composed from DSRP-484, removes nominated structures to see what capacities disappear, and asks whether omissions from the native grammar predict the work performed by later patches. Its candidate sets also come from documents in which each field defines its own vocabulary rather than from a vocabulary assembled by the analyst.

The BWW result does not confirm the particular DSRP decompositions, and the deficits identified by the two approaches should not be treated as identical. It is independent evidence consistent with the broader patch claim: another research program, using another ontology and another method, examined UML, found structural deficits, and proposed additional machinery to cover them. That agreement matters precisely because it came from outside DSRP.

10. Conclusion

Learn the four DSRP-484 Patterns first.

Each of these three formalisms is a tool with a shape, and after three runs that shape is known rather than guessed. Learn System Dynamics and you inherit loop-seeking and accumulation-seeking; you may find stocks in things that are not stocks because, once morale has been declared a stock, the method knows exactly what to do with it. Learn Network Theory and you inherit flattening; every system arrives as one typeless level, and the wholes you need must be rebuilt from relations. Learn UML and you inherit orphaning; it names one relational end while often filing the other where neither participant can reach it. None of this makes any of the three a bad tool. It makes each a tool, with a grip, a bevel, and a set of jobs it was ground for.

Learn the four first, and the three become tools again. Reach for System Dynamics when a phenomenon is usefully represented as populations changing through flows under feedback, and pause when the method begins manufacturing stocks merely because stocks are what it can manipulate. Reach for a graph when the question is traversal or reachability, as Euler did, and notice when the question begins asking about wholes, boundaries, or perspectives. Reach for UML when software structure must be specified for others to build, while tracking the negations, viewpoints, and reverse relations carried indirectly. A general structural vocabulary does not replace specialist methods. It makes what each method reveals, suppresses, and reconstructs visible enough to choose among them intelligently.

DSRP-484 is not a fourth domain formalism competing for the same job. It proposes the elemental Organization from which such formalisms are built. The combined evidence bears on that proposal in four connected ways. The three runs separated Information from Organization and composed every native and patched construct examined from DSRP-484. Their historical findings support the comparison between omissions in the native grammars and the work performed by later patches. The ablation test introduced here adds a further check: removing a structure identified as necessary to a composition must eliminate the specific organizational capacity attributed to it. Within the three candidate sets, no construct required a fifth Pattern, ninth elemental role, fifth dynamic, or other residual structure.

The patch evidence matters because it runs in the other direction. System Dynamics drew an outside it could not otherwise represent. Network Theory repeatedly rebuilt wholes, memberships, boundaries, relational identities, and perspectives from a grammar that lacked them. UML distributed the same missing work across flags, derived properties, surrogate metaclasses, profiles, enumerations, and constraints while leaving its native architecture intact. DSRP did not manufacture these features after seeing them. Its specified elementals identify what the original grammars omitted and predict the organizational work the added machinery performs.

Three runs do not establish universality. They establish that three consequential, independently developed, and sufficiently explicit formalisms passed the same test without residual. The claims that follow are unusually exact. A single residual defeats containment. A composition that fails to preserve the construct defeats the analysis. Removing a supposedly necessary structure without loss defeats that ablation claim. And omissions that fail to predict the location and function of later patches defeat representational debt. These are discrete, inspectable failures, not broad predictions that can be rescued by interpretation.

The practical conclusion is therefore modest but consequential. Learn the elemental Organization first. Then learn the specialist grammar with a clearer view of what it makes easy, what it makes difficult, and what it has already had to rebuild. The next formalism may extend the convergence or break it. Either result will tell us something real.

References

1. Cabrera D, Cabrera L. Is System Dynamics Made of DSRP? Testing Whether the Organizational Structure of System Dynamics is Contained within DSRP-484. *Journal of Systems Thinking*. 2026;6: 1–59.
2. Cabrera D, Cabrera L. Is Network Theory Made of DSRP? A Structural Decomposition of Nodes, Edges, Centrality, Communities, and Network Growth. *Journal of Systems Thinking*. 2026;1: 1–33.
3. Cabrera D, Cabrera L. Is UML made of DSRP? A structural decomposition of elements, relationships, classes, behavior, and views. *Journal of Systems Thinking*. 2026;6: 1–30.
4. Cabrera D. The Unified Calculus of Organization, Structure, Emergence, and Meaning: The Patterns that Connect Mind and Nature. *Journal of Systems Thinking*. 2025;5: 1–46.
5. Cabrera D, Cabrera L. DSRP-484: Building an advanced understanding of O-Theory, DSRP Theory, and the DSRP Calculus. *Journal of Systems Thinking*. 2026;6: 1–12.

6. Cabrera D, Cabrera L. O-Theory: The Science of Organization. *Journal of Systems Thinking*. 2025;5: 1–12.
7. Cabrera D, Cabrera L, Cabrera E. Living DSRP Evidence Compendium for the Science of Organization. Cabrera Research Lab; 2026. Available: <https://dsrpevidence.org/evidence>
8. Cabrera D, Cabrera L, Cabrera E. Counterexamples and Resolutions Data. The DSRP counterexample record: candidate falsifications of O-Theory and the resolutions that settle them. 2026. Available: dsrpevidence.org/
9. Canales, A, Maymon, P, Cabrera, D. The Universal Cognitive Grammar of Systems Mapping: A Rubric to Evaluate the Various Tools and Techniques of Systems Mapping. *Journal of Applied Systems Thinking*. 2020; 9.
10. Cabrera Research. Object-Oriented Thinking (OOT). 2026. Available: <https://help.cabreraresearch.org/object-oriented-thinking-oot>
11. Ford DN. A System Dynamics Glossary. *System Dynamics Review*. 2019;35: 369–379.
12. Richardson GP. Reflections on the Foundations of System Dynamics. *System Dynamics Review*. 2011;27: 219–243.
13. Newman MEJ. The structure and function of complex networks. *SIAM Review*. 2003;45: 167–256.
14. Barabási A-L. *Network Science*. Cambridge: Cambridge University Press; 2016.
15. Euler L. *Solutio Problematis ad Geometriam Situs Pertinentis*. *Commentarii Academiae Scientiarum Petropolitanae*. 1741;8: 128–140.
16. Object Management Group. *OMG Unified Modeling Language (OMG UML), Version 2.5.1*. Object Management Group; 2017 Dec. Available: <https://www.omg.org/spec/UML/2.5.1/>
17. Laumann EO, Marsden PV, Prensky D. The Boundary Specification Problem in Network Analysis. In: Burt RS, Minor MJ, editors. *Applied Network Analysis: A Methodological Introduction*. Beverly Hills, CA: Sage; 1983. pp. 18–34.
18. Cabrera D, Cabrera L. The Pareto structure of thought: empirical discovery of the six foundational mental moves. *J Syst Think*. 2025;2025: 1–41.
19. Cabrera D, Cabrera L. The Structure of Thought and Reality: Empirical Evidence for the Three Universal Laws Linking Mind, Matter, and Meaning. *Journal of Systems Thinking*. 2025;5: 1–20.
20. Wand Y, Weber R. On the Ontological Expressiveness of Information Systems Analysis and Design Grammars. *Information Systems Journal*. 1993;3: 217–237.
21. Opdahl AL, Henderson-Sellers B. Ontological Evaluation of the UML Using the Bunge–Wand–Weber Model. *Software and Systems Modeling*. 2002;1: 43–67.