DIAGNOSTIC AND REPLACEMENT SERIES · PAPER 5

THE INSTITUTIONAL REPLACEMENT PIPELINE

From Recurring Structural Failure to Buildable Substitute Form

Jamie Forrester

Independent Systems Architect · Edinburgh, UK

hello@jamieforrester.com · jamieforrester.com

April 2026 · v3.0 · Working Paper

The institution persists while hidden carriers absorb what it no longer reliably holds. Replacement begins when that burden becomes avoidable.

Abstract

This paper proposes an Institutional Replacement Pipeline for domains where recurring failure is driven by function fusion, continuity fragmentation, and burden transfer — where the work required to keep an institution usable is absorbed by hidden carriers outside its formal boundary, making critique and reform insufficient to reduce dependence. The pipeline has five sequential stages: Diagnosis, Kernel Extraction, Redesign, Governed Manufacture, and Substitute. Together they form a structured path from recurring structural failure to a buildable substitute form that reduces hidden burden in lived use.

The paper synthesises prior work on function fusion (Paper 2), redesign failure (Paper 3), and institutional migration (Paper 4) into a single operational method. It introduces the Truth Kernel (the minimal durable continuity structure any replacement must protect), the Build Grammar (Primitive → Law → Sequence → Gates → Output), and the distinction between a product and a substitute. Spectrum Registry provides preliminary evidence for the manufacturing layer. Cross-domain transfer is held explicitly as a hypothesis, not a proven law. A Pipeline Readiness Test and practitioner handoff complete the paper's instrument layer. The governing tagline: replacement begins where burden becomes avoidable.

For System Builders and Institutional Designers — Read This First

You are in the failure class this paper addresses if the same burdens keep returning despite repeated correction — and the reason the institution persists is that hidden carriers absorb what it no longer reliably holds. Diagnosis alone does not produce a substitute. The Institutional Replacement Pipeline is the construction path from structural failure to a buildable substitute that reduces hidden burden in lived use.

The pipeline has five stages in dependency order: Diagnosis, Kernel Extraction, Redesign, Governed Manufacture, and Substitute. Each stage requires the output of the one before it. A substitute missing any stage is a product, a principle, or an architecture — not a substitute.

Three questions reveal where the stall is:

- Has the truth kernel been named — the minimal continuity structure that must persist in the substitute's custody, institution-agnostic?
- Is the redesign lawful — does it specify custody, authority, invariants, and refusal conditions before the build begins?
- Does the governed build system refuse invalid states structurally, or correct them after the fact?

If any stage is incomplete, the pipeline has not been completed. The Pipeline Readiness Test in Section XI identifies where the stall is. The practitioner handoff in Section XII specifies what to do at each stage. Paper 6's Admissibility Stack specifies what the finished substitute must then satisfy to become institutionally real.

Series Position

This is Paper 5 in the Diagnostic and Replacement Series. Papers 1–3 extracted the diagnostic laws: interface-legitimacy mismatch, function collapse under scale, and the structural reason why correction loops cannot generate redesign from within the same frame. Paper 4 named the Migration Law: institutions lose centrality not when exposed, but when their transferred burdens become avoidable. Paper 5 answers the construction question: once migration is theoretically possible, how is the substitute built?

Paper 6 names the admissibility threshold the finished substitute must then cross — being correctly built is necessary but not sufficient for institutional inheritance. Paper 7 names the Governed Correction Sequence, the meta-layer showing what Papers 1–6 are all instances of. Paper 8 applies the complete toolkit to a new failure class: externally imposed primitive compression.

Paper 4's Migration Law explains when replacement begins. Paper 5 explains how to build the substitute that triggers it. Paper 6 explains what the substitute must become to be trusted. These three papers form the replacement arc.

Reader Map

Reading Path	Sections
Full read — founders, institutional designers, system architects	All sections and appendices
Executive read — CEOs, CTOs, reform programme directors	For System Builders box, then Sections I, IV, V, XI, XII, XVI
Series reader	Sections I and XVI only

Section I The Replacement Name Problem

"Institutional replacement" is not a category. It is a label applied to four structurally different acts that must be separated if the argument is to remain rigorous. Conflating them produces the most common failure in reform: visible institutional motion mistaken for displacement.

Type	What It Is	What It Produces	What It Cannot Do
Critique	Makes failure visible. Describes what is wrong and names where the burden falls.	Description of structure	Change the structure it describes. A critique, however thorough, does not alter function allocation.
Reform	Modifies process inside an inherited frame. Adds procedure, redistributes load, or improves interface.	New procedure, redistributed load, improved interface	Change the primitive, boundary, or function allocation. Reform is internal to the institution’s own correction loop.
Redesign	Changes the frame itself — the primitive, boundary, continuity logic, function allocation, and invariants.	A new architecture	Guarantee that architecture is ever instantiated. Redesign does not produce replacement. It produces a specification.
Replacement	Another coordination route becomes the place people actually go to meet the same need.	Migration of reliance	Be produced by design alone. Replacement is a migration event, not a design event. It occurs when burden becomes avoidable.

These are not interchangeable. A system can survive critique. It can survive repeated reform. It can survive coherent redesign proposals. Its centrality ends only when reliance actually shifts elsewhere. The structural question is not whether the institution is moving. It is whether the burden of remaining inside it has become avoidable.

That is the paper’s entry point.

Section II The Institutional Promise

Institutions are encountered first not as structures, but as promises. They promise that relevant signals will be noticed. They promise that those signals will be competently interpreted. They promise that legitimate action will follow. They promise that continuity will survive across time and encounters.

These can be stated as four irreducible functions (Paper 2's Four-Function Law): Sensing — reality enters the system; Interpretation — reality becomes meaning; Authority — action is selected and enacted; Memory — continuity is preserved across time.

The institution's public promise is coherence across all four. The structural question is where that coherence actually lives — whether inside the formal institution or dispersed into the people surrounding it who compensate for its absence. Where coherence lives outside the formal boundary, burden transfer is the mechanism of institutional persistence.

Section III The Structural Failure Pattern

Not every institutional problem requires a replacement pipeline. Some failures are narrow, technical, and already understood at the implementation layer. This paper is concerned with a different class: recurring failures in which the same burdens reappear despite repeated corrective effort. Three mechanisms matter most.

Function Fusion

Where sensing, interpretation, authority, and memory converge into the same human node under ambiguity and consequence, recurring strain becomes predictable (Paper 2). Consider a GP appointment. The clinician must simultaneously notice what the patient is describing (sensing), form a diagnostic interpretation (interpretation), decide on and enact a response (authority), and hold or reconstruct the patient's continuity record (memory). Where these functions converge in a single node without adequate infrastructure, continuity reconstruction absorbs the attention that should have gone to clinical judgment.

The structural problem is not that clinicians lack skill. It is that the infrastructure forces skill to be spent on the wrong function. This is not workload. It is structural concentration producing predictable failure under scale and time pressure.

Burden Transfer

Institutions often remain usable because unresolved work is absorbed by hidden carriers: families who become the memory layer the health record system is not, carers who bridge coordination gaps between services that do not communicate, patients who translate their own experience into the categories a broken intake form imposes, frontline workers who carry the informal knowledge that makes the formal system navigable.

Burden transfer is therefore not a side effect of institutional failure. It is frequently the mechanism of its persistence. The institution continues to function, and that functioning conceals the structural cost borne elsewhere. Paper 4's Migration Law follows directly: institutions persist while that burden can be transferred outward; they begin to be replaced when a superior architecture makes it avoidable.

Missing Redesign Capacity

Institutions may become highly skilled at detecting and documenting failure while remaining unable to redesign the frame that produces it (Paper 3's Redesign Law). They can patch, review, audit, escalate, and defend. They cannot necessarily redefine the primitive, reallocate

the functions, or redesign the boundary from within the same correction loop. Correction inside the loop is not equivalent to redesign of the loop. The difference is structural, not motivational.

Section IV The Institutional Replacement Pipeline

The Replacement Pipeline Law

Institutions do not become replaceable when their failures are exposed. They become replaceable when the burden they transfer can be converted into a governed substitute that reduces that burden in lived use.

Replacement becomes possible only when five stages are satisfied in sequence. Each stage requires the output of the one before it. No stage can be skipped without degrading the stage that follows.

Stage	Required Question	Key Output	Failure Mode if Skipped
1. Diagnosis	What topology is failing — where does function fusion concentrate, and where does burden transfer?	Located failure pattern: named hidden carriers and traced burden paths	Critique without a construction path — the source is described but not addressed.
2. Kernel Extraction	What minimal continuity structure must persist for the replacement to preserve coherence without reproducing the legacy failure topology?	Named truth kernel: what is held, by whom, under what invariants, and who may act on it	Redesign built on the wrong object — burden transfer remains intact at the root.
3. Redesign	What lawful architecture preserves the kernel across encounters, transitions, and time — without requiring the incumbent?	Specified custody model, authority model, continuity logic, permitted transitions, and refusal conditions	Architecture theatre — a correct specification that cannot build or enforce its own invariants.
4. Governed Manufacture	Can the architecture be instantiated under invariant gates — where invalid states are refused structurally before execution continues?	Working system governed by the Build Grammar: Primitive → Law → Sequence → Gates → Output	Implementation improvisation — specification dissolves during build; drift re-enters the architecture.
5. Substitute	Does the new route actually reduce hidden burden in lived use — does the hidden labour fall?	Measurable burden reduction: hidden carrier work falls, habitation friction drops, relief is directly experienceable	A product — technically correct and lighter by design, but the hidden labour has not been measured or confirmed to fall.

Paper 4’s Migration Law explains when institutions begin to lose centrality: when the burden of remaining inside them becomes avoidable elsewhere. The Replacement Pipeline is the construction method that makes that avoidability real. Without a governed path from diagnosis to substitute, migration remains theoretical.

Section V Kernel Extraction

Diagnosis alone does not produce a replacement. The next stage is to identify what the replacement must protect. Without kernel extraction, redesign remains cosmetic. An institution may improve interface, increase reporting density, or digitise a workflow while leaving burden transfer intact and continuity fragmented.

The Truth Kernel

A truth kernel is the minimal durable continuity structure that must remain intact if the replacement system is to preserve coherence without reproducing the failure topology of the legacy institution.

The truth kernel is not the whole replacement system. It is the irreducible continuity object the replacement must protect. Everything else is built around it — but the kernel must be defined before anything else can be specified.

The design shorthand: Give the record back. Where an institution fails by fragmenting memory, dispersing continuity, or forcing individuals to repeatedly reconstruct truth under external control, the replacement must restore durable record custody to the correct node.

Kernel Integrity

A truth kernel is institution-agnostic if and only if it can be held, read, and acted upon without requiring the original institution’s interpretation or permission.

If the record still requires the old institution to explain it, validate it, or authorise access to it, the kernel has not been extracted. It is still fused. The replacement has reproduced the dependency it was built to eliminate.

Kernel extraction forces the redesign question into a stricter form: What exactly must persist? Who properly holds custody? What may be written, verified, or amended, and by whom? What continuity must survive transition across time, institutions, and encounters? What authority may act on the record, and under what conditions? These are replacement questions, not optimisation questions. They cannot be answered by improving what exists. They require specifying what must exist instead.

Section VI Redesign

Diagnosis and redesign are not the same activity. Conflating them produces two distinct failure modes: diagnosis without redesign produces critique with no construction path; redesign without diagnosis produces confident construction of a system whose failure modes were never understood.

Diagnosis works backward from symptoms to structure. Redesign works forward from primitive to admissible form.

A lawful replacement architecture must specify, at minimum:

- the kernel to be preserved
- the invariants that must not break
- the custody model — who holds what, and under what conditions
- the authority model — who may act on the record, and when
- the continuity logic — what must survive across transitions
- the permitted state transitions
- the enforcement boundaries
- the failure modes to be refused by design rather than handled by exception

At this stage the replacement is not yet software. It is a lawful coordination architecture. A better structure that cannot be instantiated remains theory. The next stage is governed manufacture.

Section VII Governed Manufacture

The central operational claim of this paper is that replacement requires a manufacturing layer: a governed build system capable of converting a redesign architecture into working software where invalid states are refused structurally rather than corrected after the fact.

The Build Grammar

Primitive → Law → Sequence → Gates → Output

Primitive: the irreducible unit the system governs (an ownership claim, a patient record, a continuity node).

Law: the invariant the system cannot bypass without ceasing to be what it claims. This is the refusal rule, not a preference.

Sequence: the ordered build pipeline, each stage dependent on the one before it.

Gates: automated checks that refuse promotion when invariants fail. Negative promotion is the mechanism: if the law is violated, the build does not advance.

Output: working software in which invalid states are refused by design, not corrected after execution.

This grammar places ambiguity resolution upstream of execution. Generation is one constrained stage inside a larger governed architecture, not the primary intelligence of the build process. The approach shares goals with formal specification methods — including model-driven engineering and specification-first methodologies — but operates through natural-language constitutional specifications and gate-based generation rather than mathematical proof.

In the current implementation (PromptFactory), this grammar operates as follows. The primitive is a symbolic ownership claim. The law is uniqueness: no two valid claims may occupy the same canonical coordinates simultaneously. The sequence is a multi-stage build pipeline ordered by dependency. The gates are automated checks that block promotion whenever any invariant fails. The output is a working system in which invalid ownership transitions are refused structurally.

Substitute vs Product

	Product	Substitute
What it serves	A user task	A burden previously carried by a failing institution
Success test	Does the user complete the task?	Does the hidden labour fall in lived use?
Requires burden reduction	No — a product may be lighter and correct without it	Yes — without measurable burden reduction, it is not a substitute
Relationship to incumbent	Independent — does not inherit institutional reality	Inherits the burden the incumbent displaced outward to hidden carriers

Conventional implementation often follows this sequence: specification is partial when execution begins; interpretation continues during coding; design remains unstable throughout the build; quality is checked after substantial drift has already entered the system. Governed manufacture attempts a different sequence: identify the primitive, define the law, specify the sequence, apply gates, refuse promotion when invariants fail. This is a move from implementation improvisation toward governed manufacture.

Section VIII The Substitute Criterion

A replacement system should not be called a substitute merely because it exists and is technically correct. Three criteria must be satisfied. They are ordered by dependency: burden reduction is the terminal test, but continuity preservation and lower habitation friction must hold for burden reduction to be real rather than relocated.

Criterion	Required Question	Failure Mode if Absent
Burden reduction	Has the hidden carrier work actually fallen — not in principle, but in	The substitute is a product: technically correct and lighter in design, but the hidden

Criterion	Required Question	Failure Mode if Absent
	measured lived use?	labour remains unchanged. The burden has not been absorbed. It has been ignored.
Continuity preservation	Does the truth kernel remain operationally intact across time, encounters, and system transitions?	The substitute has relocated the fragmentation rather than resolving it. The kernel is held in theory; its custody breaks at transition.
Lower habitation friction	Is the lived burden of using the substitute lower than remaining inside the legacy route?	Relief has not occurred. The old route remains the default under pressure. Hidden carriers continue to compensate because the substitute is not yet lighter to inhabit.

Relief precedes theory. Replacement begins when users move because the new route is lighter to inhabit — not because they have adopted an argument.

Section IX Preliminary Instance: Spectrum Registry

Spectrum Registry is a law-first symbolic registry under active construction. Its evidentiary role in this paper is narrow: it demonstrates the manufacturing layer and part of the redesign discipline in a bounded, non-trivial domain. It does not demonstrate institutional substitution in healthcare, education, or finance.

Specifically, it demonstrates: primitive identification before build begins; law-first system design where invariants precede features; constrained ownership logic with explicit write rules; refusal conditions that block invalid transitions by design; and governed build methodology across a multi-stage pipeline on a genuinely hard software problem — an event-sourced, ledger-first platform with a canonical ownership model.

The domain — symbolic colour ownership — is not the point. The method is: invariant-led refusal rather than permissive execution followed by post hoc correction. Its contribution here is methodological rather than domain-general. It demonstrates that the Build Grammar can survive translation into functioning software under real build constraints.

Spectrum Registry is the first step in the replication hypothesis (Section XIV). It establishes that the manufacturing layer operates in at least one bounded domain. The second domain transfer has not yet been attempted.

Section X A Compact Worked Example: Patient Continuity

The following applies the pipeline to a high-stakes domain as a design exercise — not a completed substitute, but an illustration of the pipeline’s logic in a domain where burden transfer is the structural norm.

Pipeline Stage	Application to Patient Continuity
Failure condition (pre-Diagnosis)	A patient with a complex chronic condition attends a new specialist appointment. The specialist has no access to medication history, prior diagnoses, or documented contraindications. The patient reconstructs what they can from memory. The hidden labour — the patient reconstructing their own continuity, the family member attending specifically to supply what the system cannot — is invisible to the institution's formal accounting.
Diagnosis	Function fusion at the clinician node: sensing, interpretation, authority, and memory are all required of the clinician at the point of consequence, with insufficient continuity infrastructure. Burden transfer: patient and family absorb the memory and coordination function the institution cannot hold.
Kernel Extraction	Truth kernel: a durable, append-only personal health record held in the patient's custody, not distributed across institutions, and accessible to any clinician the patient authorises. Kernel integrity: holdable and actionable without the hospital's permission or re-entry.
Redesign	Lawful architecture: custody remains with the patient across all encounters; append-only, no silent modification; read access is patient-controlled; write authority is bounded by encounter; refusal condition blocks contradictory states without justification.
Governed Manufacture	Build Grammar applied: primitive = patient-held continuity record; law = append-only, patient custody, no silent modification; sequence = staged pipeline from record creation through read/write authority to refusal gate; gates block invalid transitions before they enter the system.
Substitute criterion	If a working substitute preserved this kernel, the following hidden labour would become unnecessary: the patient reconstructing their history at each encounter; the clinician proceeding on incomplete context; the family member attending to supply continuity the system cannot hold. The substitute does not make the clinical judgment. It preserves the continuity on which that judgment depends.
Paper 6 handoff	A technically complete substitute in this domain must then satisfy Paper 6's Admissibility Stack: recognition by adjacent clinical systems, refusal integrity, contestability, governance legibility, fallback integrity, legal interface, and recapture resistance. Being correctly built is the pipeline's output. Being admissible is Paper 6's domain.

Section XI The Pipeline Readiness Test

Before committing to the pipeline, determine whether the failure class matches. Answer each question honestly. Score each yes as 1, each no as 0.

Q	Question	What a Yes Reveals
1	Does the failure recur despite repeated correction attempts?	Structural failure, not implementation error. The correction loop is working inside the frame that generates the failure (Paper 3).
2	Is burden transfer the main	The pipeline's entry condition is met. The institution

Q	Question	What a Yes Reveals
	persistence mechanism — are hidden carriers absorbing what the institution cannot hold?	persists because its cost has been externalised, not because it is coherent.
3	Can you name a truth kernel — the minimal continuity structure that must persist in the substitute's custody?	Kernel extraction is possible. The replacement has a defined primitive.
4	Is redesign capacity absent from the existing correction loop (Paper 3's missing redesign layer)?	The correction loop will not generate a substitute internally. The pipeline is the only construction path.
5	Is the domain continuity-heavy — driven by record custody, function fusion, and fragmentation across encounters?	The pipeline applies most directly. Continuity-light domains may require different approaches.
6	Is there a governed build path — can invariants be enforced structurally, with gates that refuse invalid states before execution?	Governed manufacture is feasible. The Build Grammar can be applied.
7	Can burden reduction be directly measured in lived use — is the hidden carrier work identifiable and trackable?	The substitute criterion (stage 5) is testable. Substitution can be confirmed rather than assumed.
8	Is there a substitute form that does not require the incumbent's permission to operate?	The admissibility threshold (Paper 6) may be reachable. The pipeline can produce an institution-agnostic output.

Score	Status	Action
6–8 yes	Pipeline applies — proceed	Begin at Diagnosis. Name the failure topology and the hidden carriers before anything else.
4–5 yes	Partial fit — identify the missing condition	Locate which condition is absent. The pipeline applies to the stages before that condition. Do not advance past the absent condition.
Fewer than 4 yes	Pipeline does not apply to this failure class	The failure may require a different approach: targeted reform, technology upgrade, or a domain with different structural characteristics.

Section XII Practitioner Handoff

For each stage, the following specifies the key action, the primary failure mode to avoid, and the exit criterion — how you know the stage is complete and the next stage can begin.

Stage	Key Action	Primary Failure Mode	Exit Criterion
Diagnosis	Map the failure topology: locate function fusion nodes, name hidden carriers, trace where burden lands, identify who performs the hidden labour.	Confusing symptoms with structure — stopping at critique without locating the structural source.	The structural source is named. Hidden carriers and their labour are identified and documented.
Kernel Extraction	Define what must persist: what is held, who holds it, who may write, read, or amend, under what invariants, and what must survive institutional transition.	Extracting a kernel that still requires the incumbent — fused kernel reproduces the dependency.	A kernel statement that is institution-agnostic: holdable and actionable without the incumbent's presence or permission.
Redesign	Specify the lawful architecture: custody model, authority model, continuity logic, permitted transitions, refusal conditions, enforcement boundaries, failure modes refused by design.	Architecture theatre — a correct specification with no build path and no enforceable invariants.	A written redesign that a governed build system can implement directly, with structurally enforceable refusal conditions.
Governed Manufacture	Build under the Build Grammar: Primitive → Law → Sequence → Gates → Output. Refuse promotion when invariants fail. Treat the constitution as law, not suggestion.	Implementation improvisation — gates become advisory; drift re-enters; specification dissolves.	A working system where invalid states are refused structurally. Gates have blocked at least one real promotion.
Substitute	Test burden reduction, continuity preservation, and habitation friction in lived use. Run the Hidden Carrier Audit: list every task a hidden carrier performs. Confirm which items have fallen.	Declaring substitution before burden has measurably fallen — confusing adoption with replacement.	Hidden carrier work has demonstrably fallen. Relief is directly experienceable. The old route is no longer the default under pressure.

Each stage's exit criterion is the gate to the next stage. Do not advance until the criterion is met. An incomplete prior stage does not merely slow the pipeline — it invalidates the stage that follows it.

Section XIII	Pipeline Failure Modes

Each incomplete pipeline produces a predictable output. Naming these failure modes prevents confusing an incomplete pipeline with a completed one.

Failure Mode	What It Produces	What It Cannot Produce
Diagnosis without kernel extraction	Critique with precise structural naming. The failure is located, named, and described.	A construction path. The structural source is known but no replacement primitive exists.
Kernel extraction without redesign	A principle — a named continuity object without architecture. "Give the record back" without specifying who holds it, under what invariants, and how refusal works.	A buildable specification. The kernel is defined but cannot be instantiated.
Redesign without governed manufacture	Architecture theatre — a correct design that cannot enforce its own invariants. The specification exists. The build does not.	Working software that refuses invalid states. The architecture is real; the enforcement is not.
Governed manufacture without burden reduction testing	A product — technically correct, potentially lighter, and governed. Users may prefer it.	A substitute. The hidden labour has not been measured. Displacement has not been confirmed.
Substitute without admissibility (Paper 6)	A correct, lighter route that reduces burden for users who can access it. The pipeline is complete.	Institutional inheritance. The substitute cannot carry consequence without the incumbent’s recognition. Paper 6 begins where this pipeline ends.

Section XIV The Replication Hypothesis

Cross-domain transfer remains a hypothesis. The pipeline proposes a test sequence. Only after a significant portion of this sequence is satisfied should the pipeline be treated as a general method rather than a coherent and partially evidenced architecture.

Test	Condition	Current Status
1. Bounded substitute instance	The pipeline produces a working route in at least one domain, with governed manufacture and measurable burden reduction	Partial — Spectrum Registry demonstrates the manufacturing layer and kernel discipline. Full burden reduction testing is in progress.
2. Second-domain transfer	The method survives contact with a structurally different domain with different hidden carriers, different continuity structures, and different institutional incumbents	Not yet demonstrated
3. Repeated	The architecture is not a one-off exception;	Partially evidenced in Spectrum Registry

Test	Condition	Current Status
invariant stability	invariants hold under real build pressure and at scale	build pipeline across a forty-stage sequence
4. Real-world burden reduction	Substitution has occurred in lived use, not just in description; hidden carrier work has measurably fallen in at least one domain	Not yet demonstrated at domain scale
5. Independent reproduction	The method functions when applied by a builder other than the originator, using the Build Grammar without the originator's involvement	Not yet tested

Section XV Limits of the Argument

This paper has clear limits that must be stated without softening. The evidentiary standard for this pipeline is displacement, not description. Until a domain shows measurable burden reduction in lived use, the pipeline remains a high-fidelity hypothesis.

Limit	What It Means
Full institutional replacement not yet proven	The paper proposes a pipeline and partially grounds it in live build work. Spectrum Registry demonstrates the manufacturing layer. It is not a health substitute, an education substitute, or a justice substitute.
Strongest in continuity-heavy domains	The argument applies most directly where recurring failure is driven by continuity loss, burden transfer, function fusion, and ownership ambiguity. It may apply less directly where primary constraints are physical infrastructure, coercive enforcement, or coordination problems that are not primarily record-based.
Governed manufacture not broadly validated	The manufacturing layer is conceptually developed and partially evidenced through live build work. It has not been demonstrated across multiple substitute domains.
Replication remains a hypothesis	Cross-domain transfer is a plausible next step, not a demonstrated general law. The paper proposes the test conditions (Section XIV) rather than reporting their satisfaction.
Paper 6 not reached by the pipeline alone	A substitute that satisfies all five stages is technically complete and burden-reducing. It is not yet admissible. Paper 6's Admissibility Stack is the separate threshold the substitute must then cross.

This pipeline, like the substitutions it describes, currently holds truth without a full recognition channel. Its refusal invariants are not yet structurally enforced across multiple domains. Whether it becomes a general method depends on independent replication and on whether the substitutes built from it are lighter to inhabit than the institutions they are meant to replace. By its own criteria, it is a buildable candidate awaiting the empirical test.

Section XVI Conclusion

What this paper adds to the series

Instrument	Definition
The Replacement Pipeline Law	Institutions become replaceable not when failures are exposed but when the burden they transfer can be converted into a governed substitute that reduces that burden in lived use
Institutional Replacement Pipeline	Five sequential stages: Diagnosis → Kernel Extraction → Redesign → Governed Manufacture → Substitute. Each depends on the output of the one before it
Truth Kernel	The minimal durable continuity structure that must remain intact for the replacement to preserve coherence without reproducing the legacy failure topology
Kernel Integrity	A truth kernel is institution-agnostic only if it can be held, read, and acted upon without the original institution's permission or re-entry
The Build Grammar	Primitive → Law → Sequence → Gates → Output. The governing grammar of governed manufacture
Governed Manufacture	A build methodology in which invariants precede features, gates refuse promotion when laws fail, and ambiguity resolution is upstream of execution
Substitute vs Product	A product serves a user task. A substitute inherits a burden previously carried by a failing institution. The test: does the hidden labour fall?
Pipeline Failure Modes	What each incomplete pipeline stage produces, and what it cannot produce without the next stage
Pipeline Readiness Test	Eight-question instrument for determining whether the failure class matches the pipeline, with scoring and action guidance
Practitioner Handoff	Per-stage specification: key action, primary failure mode, exit criterion
Replication Hypothesis	Five-step test sequence specifying what would constitute stronger evidence for the pipeline as a general method

Falsifiable Prediction

Falsifiable Prediction

If the full pipeline is applied to a continuity-heavy domain with clearly identified burden transfer and named hidden carriers — where all five stages are completed, governed manufacture gates have blocked at least one invalid promotion, and burden reduction is measured in lived use — the pipeline should produce a buildable substitute candidate whose burden-reduction claim can be directly tested in the domain.

If all five stages are followed and the burden does not measurably fall in lived use, the pipeline's claim to generality is falsified. If the burden falls but admissibility (Paper 6) is not subsequently achieved, the pipeline is confirmed and Paper 6's Admissibility Stack is the binding constraint.

Position in the series

Most institutional writing stops at critique. Most software methodology starts at implementation. This paper proposes the missing bridge: a structured sequence for moving from recurring structural failure through kernel extraction and lawful redesign to a manufactured substitute — testable against the only criterion that matters: whether the hidden labour actually falls.

The loop closes conceptually with diagnosis. It closes architecturally with redesign. It closes operationally with governed manufacture. It closes in reality only when relief becomes directly experienceable in use. And it closes institutionally — when the substitute becomes the place consequence lives — only through the admissibility threshold Paper 6 specifies.

The hidden carriers are the test. If the family member still attends every appointment to supply what the system cannot hold, the pipeline has not completed. If the patient still reconstructs their own history at each encounter, relief has not occurred. The institution persists while those carriers absorb what it no longer reliably holds. The pipeline ends when that burden becomes avoidable through something they can reach without asking the old institution for permission.

Diagnosis names the failure.
Manufacture builds the exit.
Admissibility makes it institutional.

The institution persists while hidden carriers absorb what it no longer reliably holds. Replacement begins when that burden becomes avoidable.

Jamie Forrester · hello@jamieforrester.com · April 2026

If this maps to a system you are building or a failure pattern you are trying to replace, you can reach me at hello@jamieforrester.com

Appendix A — Evidence Reference and Theoretical Grounding

Tier	Source Type	Contents and Weight
A	Prior series papers (direct)	Forrester 2026a–d: interface-legitimacy mismatch (Paper 1), Four-Function Law (Paper 2), Redesign Law (Paper 3), Migration Law (Paper 4). These provide the diagnostic foundation the pipeline synthesises.
B	Live build evidence (manufacturing	Spectrum Registry build pipeline: forty-stage sequence, event-sourced ledger, canonical ownership model, gate-based invariant enforcement.

Tier	Source Type	Contents and Weight
	layer)	Demonstrates Build Grammar in a hard, non-trivial domain.
C	Theoretical grounding (institutional theory)	North (1990): why institutions persist via path dependency. Scott (2001): institutional legitimacy and cognition. Star & Strauss (1999): invisible work and hidden carriers. Hevner et al. (2004): design science as legitimate scholarly method.
D	Worked example (design exercise)	Patient continuity case in Section X. Illustrative, not empirical. Demonstrates pipeline logic in a high-stakes domain without claiming completed substitution.
E	Structural inference (framework-derived)	Pipeline failure modes, substitute criteria, and the replication hypothesis are derived from prior series laws and the pipeline's own logic. Clearly marked as inference throughout.

Appendix B — Glossary of Named Concepts

Term	Definition and Source
Institutional Replacement Pipeline	The governed five-stage sequence by which recurring structural failure is converted into a buildable substitute form: Diagnosis → Kernel Extraction → Redesign → Governed Manufacture → Substitute. [Paper 5]
The Replacement Pipeline Law	Institutions do not become replaceable when failures are exposed. They become replaceable when the burden they transfer can be converted into a governed substitute that reduces that burden in lived use. [Paper 5]
Truth Kernel	The minimal durable continuity structure that must remain intact for the replacement system to preserve coherence without reproducing the legacy failure topology. [Paper 5]
Kernel Integrity	A truth kernel is institution-agnostic if and only if it can be held, read, and acted upon without the original institution's interpretation, validation, or permission. [Paper 5]
The Build Grammar	Primitive → Law → Sequence → Gates → Output. The governing grammar of the governed manufacture stage. [Paper 5]
Governed Manufacture	A build methodology in which invariants precede features, gates refuse promotion when laws fail, ambiguity is resolved upstream of execution, and the constitutional specification governs all downstream generation. [Paper 5]
Substitute	A coordination route that inherits a burden previously carried by a failing institution, demonstrated by measurable reduction in hidden carrier work in lived use. Distinct from a product, which serves a user task. [Paper 5]
Hidden Carrier	An individual or group (family member, frontline worker, carer, patient) who absorbs the functions an institution no longer reliably holds, performing work that does not appear in the institution's formal

Term	Definition and Source
	accounting. [Paper 2, Paper 5]
Burden Transfer	The mechanism by which institutional failure is absorbed by hidden carriers rather than resolved. Institutions persist while burden can be transferred; they begin to be replaced when a superior architecture makes that burden avoidable. [Paper 4, Paper 5]
Pipeline Failure Mode	The predictable output of an incomplete pipeline: diagnosis without kernel produces critique; kernel without redesign produces principle; redesign without manufacture produces architecture theatre; manufacture without burden testing produces a product; substitute without admissibility produces a peripheral route. [Paper 5]
Function Fusion	The concentration of sensing, interpretation, authority, and memory in the same human node at the point of consequence. The structural source of recurring strain under scale. [Paper 2]
Migration Law	Institutions persist while the burden of their incoherence can be transferred outward; they begin to be replaced when a superior coordination architecture makes that burden avoidable. [Paper 4]
Admissibility	The threshold at which a substitute becomes sufficiently trustworthy, inspectable, and recognisable to carry consequence without the incumbent's permission. The Pipeline produces a buildable substitute. Admissibility determines whether it becomes institutionally real. [Paper 6]

Appendix C — Series Reference

Forrester, J. (2026a). The Expertise Illusion in AI Task Marketplaces. SSRN Working Paper.

Forrester, J. (2026b). The Four-Function Law of Scalable Institutions. SSRN Working Paper.

Forrester, J. (2026c). Why Systems Can't Fix Themselves: The Missing Redesign Layer. SSRN Working Paper.

Forrester, J. (2026d). Institution Migration: How Better Coordination Makes Legacy Institutions Unnecessary. SSRN Working Paper.

Forrester, J. (2026e). The Institutional Replacement Pipeline. SSRN Working Paper. [This paper]

Forrester, J. (2026f). The Admissibility Problem: Why Better Substitutes Still Fail to Replace Worse Institutions. SSRN Working Paper.

Forrester, J. (2026g). The Governed Correction Sequence. SSRN Working Paper.

Forrester, J. (2026h). Host-Constrained Primitive Mismatch: The Structural Decay of Relationship-Coded Systems Inside Transaction-Coded Hosts. SSRN Working Paper.

Academic grounding: North, D. (1990). Institutions, Institutional Change, and Economic Performance. Cambridge University Press. · Scott, W.R. (2001). Institutions and Organizations. Sage. · Star, S.L. and Strauss, A. (1999). Layers of Silence, Arenas of Voice. CSCW 8(1–2). · Hevner, A. et al. (2004). Design Science in Information Systems Research. MIS Quarterly 28(1). · Simon, H. (1969). The Sciences of the Artificial. MIT Press.

hello@jamieforrester.com · jamieforrester.com · Edinburgh, UK · Paper 5 v3.0 · April 2026