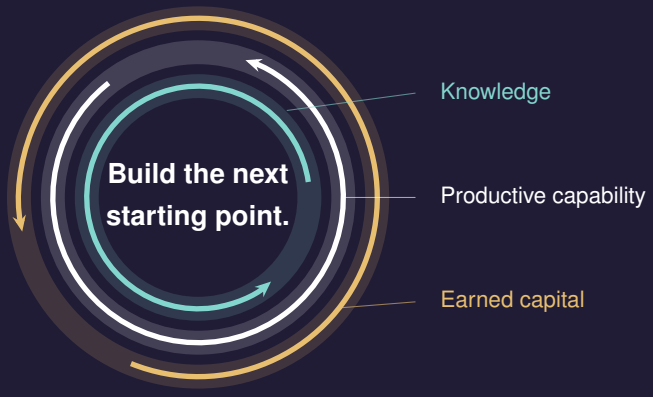
AGI ALPHA

FLYWHEEL

Inventing What Becomes Possible

TRANSFORMATIONAL INTELLIGENCE ∞



The output of one invention can become the means of the next.

Vincent Boucher

MONTREAL.AI • Montréal, Québec, Canada

2 November 2026 • Research edition 1.1

THE THESIS

Make useful invention a repeatable productive activity.

Abstract

AGI ALPHA FLYWHEEL studies an enterprise that searches for valuable outcomes, constructs missing means, and carries useful gains into further invention. Its object is the complete productive arrangement: what becomes possible, how it is produced and accepted, who participates, and how continuation is funded. Building on *Transformational Intelligence*, we ask when a constructed instrument becomes a starting point for a different instrument, and when the procedure for discovering such transitions improves.

This revised edition makes construction executable. A finite study synthesizes typed measurement programs from observation models and target quantities, verifies their certificates independently, and executes them on separate latent inputs. Twelve declared episodes of three generations compare fresh construction, a frozen first-generation library, accumulating inheritance, and a conventional rank-aware compiler. Exact identifiability tests distinguish an unavailable answer from exhausted search. Executed arithmetic, archive traffic, and runtime are reported in their declared units. Each arm accepts 30 programs and certifies six unidentifiable requests. Accumulation reduces expression-search work, while conventional compilation wins the complete arithmetic comparison; execution work is unchanged.

We distinguish reuse, constructive succession, funded recurrence, and improvement of discovery. A composition result corrects a common frontier claim: constructing a tool can improve a continuation state or a bounded search procedure, while a complete initial policy class may already contain the plan that constructs and uses it. An economic model then connects productive gains to acceptance, competition, liquidity, and complete costs. Prior public results supply both useful constructions and failed successor or scale claims. The contribution is an operational synthesis, a reproducible construction study, and a testable route from invention to new productive activity.

The question

Which useful activity is missing because something necessary cannot yet be measured, constructed, coordinated, verified, or afforded?

Read in three passes

For the idea and a concrete invention, read Sections 1–3. For the architecture and its limits, read Sections 4–7. For economic succession, improved discovery, and the next decisive test, read Sections 8–13.

What this edition adds

The first edition's central numerical example stipulated a useful instrument and a lower successor cost. It established a funding relation, not instrument construction. This edition adds generated executable programs, independent algebraic and runtime checks, multigeneration controls, and a corrected reachability argument. The finite constructor is authored software. The broader autonomous enterprise and the US$3 billion trailing-year complete-cost profit objective remain research goals.

Keywords transformational intelligence; automated invention; productive arrangements; measurement programs; constructive succession; autonomous enterprise.

1 Build the next starting point

A useful invention changes more than the answer to today's question. It can change the means available for tomorrow's question. A measurement process can reveal a quantity that earlier observations could not identify. A compiler can turn a new specification into a qualified instrument. An interface can make previously incompatible resources productive together. An arrangement can make a small, unusual activity worth undertaking when its former fixed cost was prohibitive.

AGI ALPHA aims to make the invention of useful capabilities a repeatable autonomous activity. It searches for valuable outcomes, identifies what blocks them, constructs or acquires the missing means, and follows the consequences into another activity. The product, customer grouping, production method, organization, and financing can all be objects of this search. Automating an existing job is one possible route; inherited job categories do not define the opportunity set.

Transformational Intelligence supplies the central move: separate the authorized purpose from the inherited formulation, construct a different setup, and establish that the change matters [1]. Here that move is followed through time. A successful project may leave an instrument, a constructor of related instruments, evidence about their applicability, or available earnings. The research question is whether those assets help a distinct successor after their complete burden is counted.

The output of one invention can become the means of the next.

This proposition has several strengths. Reusing one tool amortizes its construction. Constructing a different tool with inherited means is a stronger technical relation. Funding later work from settled earnings is an economic relation. Becoming better at discovering such transitions is a further learning claim. None follows merely from the one before it.

1.1 A stable charter, an expanding opportunity set

The **authorized charter** states intended benefits, scope, and constraints. A **particular opportunity** states a beneficiary outcome and an acceptance contract. The enterprise may propose another opportunity within the charter; an experiment holds its particular outcome fixed during comparison. Thus a failed request for absolute local stress cannot be relabeled a successful request for a relative average. The average can support a separately specified offer if it has value.

This distinction permits ambition without moving a goalpost. A profitable current service can support a research programme without defining its permanent industry. A new instrument can serve a purpose its predecessor did not serve. Yet each comparison still has a definite output, complete alternatives, current authority, and an observable condition for acceptance.

1.2 The running opportunity: measurements on demand

Consider a small or unusual production run that needs a local condition measured to a specified accuracy. A custom measurement process may cost more than the activity can support. A proposed instrument foundry would construct and qualify that process, retain a useful constructor, and adapt it to another measurement problem. A related retrospective service would answer questions that were not anticipated when historical traces were collected.

These are consequential prospective activities, not validated markets. Their key obstruction is specific: the requested quantity must be identifiable from legitimately available information, and producing a defensible estimate must fit the activity's full resource envelope. The next section makes that obstruction and one construction precise.

2 Construct the missing measurement

2.1 Why more inference can be insufficient

Let the state of a two-cell process be $z = (z_1, z_2)$. Suppose its only observation is the average

$$m = (z_1 + z_2)/2. \tag{1}$$

The required output is z_1. The states $(1, -1)$ and $(0, 0)$ both produce $m = 0$ but require different answers. No procedure using only m can always return the correct local value. The obstruction belongs to the observation channel, not the sophistication of an explanation.

If a legitimate differential observation

$$d = (z_1 - z_2)/2 \tag{2}$$

can be added, the reconstruction is

$$z_1 = m + d, \qquad z_2 = m - d. \tag{3}$$

With each observation error bounded by ε, each reconstructed error is at most 2ε. This bound follows by the triangle inequality; a correlated calibration bias still has to satisfy the assumed individual bounds. The useful instrument is the observation arrangement, its calibration, reconstruction, and acceptance together.

A formula cannot install a sensor. In a physical application, the differential measurement must actually be available and qualified. In a retrospective application, the additional information must survive in accessible traces; a sensor installed today cannot recover an erased historical distinction. The finite study below constructs programs over supplied observations. It does not manufacture information absent from them.

2.2 From a formula to a constructor

Write the observation contract as $y = Az$ and the target as qz, with rational matrix A and row q. A linear reconstruction exists exactly when

$$q \in \mathrm{rowspan}(A) \quad \Longleftrightarrow \quad \ker(A) \subseteq \ker(q). \tag{4}$$

When it exists, a constructor can produce coefficients w with $wA = q$ and an executable expression for wy. An independent verifier can check the equality without trusting the constructor's search. When it does not exist, a witness h with $Ah = 0$ and $qh \neq 0$ establishes ambiguity: z and $z + h$ look identical but require different answers. These are elementary linear-algebra facts, used here as constructive interfaces.

Now change the mission. For a three-cell process, observations

$$m = (z_1 + z_2 + z_3)/3, \qquad d_{12} = z_1 - z_2, \qquad d_{23} = z_2 - z_3$$

permit the different instrument

$$\begin{aligned} z_1 &= m + \tfrac{2}{3}d_{12} + \tfrac{1}{3}d_{23}, \\ z_2 &= m - \tfrac{1}{3}d_{12} + \tfrac{1}{3}d_{23}, \\ z_3 &= m - \tfrac{1}{3}d_{12} - \tfrac{2}{3}d_{23}. \end{aligned} \tag{5}$$

If each observation error is bounded by ε, the respective output bounds are 2ε, $\frac{5}{3}\varepsilon$, and 2ε. The previous formula does not solve this changed mission. A constructor that accepts a new (A, q) can generate its reconstruction and its checks.

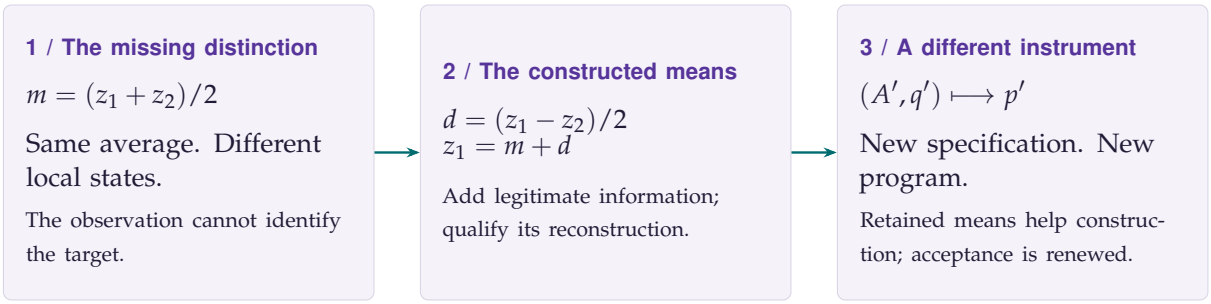


Figure 1. Constructive succession has an executable middle. A missing distinction directs construction toward an observation or representation. A qualified program implements the reconstruction. A retained constructor or compatible subprogram can help build a different program. The algebraic examples assume the stated observations; the finite study measures program construction over supplied observation contracts. Physical observation acquisition and commercial delivery remain separate tests.

2.3 What survives the first mission

Four assets have different uses. The **instrument** computes a specified output. Its **applicability contract** identifies valid inputs, quantities, units, versions, and errors. A **constructor** generates instruments for a declared family of specifications. A **discovery procedure** decides which specifications or constructions are worth investigating and how to search for them.

The first three can be implemented while the fourth stays fixed. A compiler producing a new valid program is real construction, even when it uses established mathematics. It is not by itself evidence that the compiler has learned to be a better inventor. The distinction is central to the study and to the broader enterprise.

3 A finite instrument-foundry study

The companion study replaces the first edition's stipulated capability advantage with actual construction and execution. It is a deliberately small, exact domain: typed linear-expression programs for measurement targets. Its purpose is to expose what inheritance does, what it costs, and where it fails. The public protocol fixes the finite comparison before substantive execution; the package binds that protocol, source, and results.

3.1 Inputs, outputs, and independent acceptance

Each task supplies an observation matrix, target, versioned operator identity, allowed expression primitives, and compatible retained programs. The constructor sees the specification, not the independent latent inputs used for runtime acceptance. Its expression language contains observation reads, rational scaling, addition, subtraction, and calls to retained programs.

A successful output is an actual expression with a coefficient certificate. The verifier independently expands the expression, checks its types and operator identity, and confirms that the resulting coefficients satisfy $wA = q$. Separate execution compares its output with qz on fresh latent inputs. The symbolic identity establishes exact correctness within the declared rational linear model; the runtime inputs exercise implementation. Agreement on finite inputs alone would be weaker evidence.

An impossibility certificate is also checked: $Ah = 0$ and $qh \neq 0$. Exhausting a finite search budget is a third, separate outcome. It does not establish nonidentifiability. A retained program from a changed observation operator or incompatible unit contract must be rejected or rebuilt; a familiar name cannot make it applicable.

3.2 Whole histories, distinct controls

The protocol declares 12 episodes, each with three successive target specifications. Four families cover shared substructure, disjoint targets, changed operator or type, and rank-deficient observations. Deterministic task transformations vary representations. These authored families stress the claimed mechanism; they are not random samples from industrial demand.

The fresh arm constructs without a retained executable library. The frozen arm retains only qualified first-generation programs. The accumulating arm can retain qualified new programs across generations. The conventional arm uses a rank-aware compiler with reusable algebraic work and common-subexpression opportunities. The same acceptance conditions apply to all arms. The conventional method is a serious alternative with useful problem structure, not a deliberately restricted foil.

A retained call has no invented execution discount: its body executes and its operations are counted. Costs of unsuccessful search, construction, checking, execution, library validation, retrieval, and storage remain visible. Rational coefficient size is tracked because one arithmetic operation on a large rational is not equivalent to one on a small integer. Wall and CPU measurements are reported separately from deterministic counters and have no claim to hardware-independent precision. Software development and workload authoring are supplied inputs outside these counters, as are the integer-level internals of rational arithmetic. Observation generation is common evaluation setup; program execution and independent target checks are charged.

Let $\mathbf{C}_a$ denote the reported cost vector for arm a over its complete three-generation history, with accepted results shown alongside it. The declared comparisons are

$$\Delta_{\text{reuse}} = \mathbf{C}_{\text{fresh}} - \mathbf{C}_{\text{accumulate}}, \qquad \Delta_{\text{succession}} = \mathbf{C}_{\text{frozen}} - \mathbf{C}_{\text{accumulate}}. \tag{6}$$

The first can capture ordinary reuse. The second asks whether retaining later constructions contributes beyond the first library. Neither is a universal scalar score: one component can improve while another worsens. Construction and retention costs are included from the start. Dropping the first generation would conceal the investment whose value is at issue.

3.3 Results and their interpretation

Every arm produced 30 accepted programs and six independently verified nonidentifiability certificates over its 36 requests. No primary request was rejected or exhausted. All seven separate adverse and execution-accounting controls passed. The constructors had the same accepted measurement capability on these tasks; inheritance changed work, not the observation span.

Table 1. The conventional compiler wins the full comparison. Exact scalar rational arithmetic over all 36 requests per arm, including unsuccessful expression search and independent acceptance. All arms have 30 accepted programs and six certified information-boundary results. Columns are operations, not time, money, or a complete economic cost.

Arm	Construct	Verify	Execute	Total
Fresh	47,466	4,026	141	51,633
Frozen first generation	44,442	4,458	141	49,041
Accumulating	43,326	4,674	141	48,141
Conventional rank-aware	8,751	4,674	141	13,566

Accumulation saves 3,492 counted operations against fresh construction and 900 against the frozen library. Construction/search saves 4,140 against fresh, partly offset by 648 additional verification operations. Expanded execution is exactly 141 in every arm. The observed benefit is therefore reduced constructor search, not faster execution of the delivered instruments. The conventional compiler removes much more unnecessary search: it uses 34,575 fewer total operations than accumulating expression synthesis. The complementary *OUT-ACCELERATE* model follows construction through verification and acquisition delays under stipulated durations [2]; neither those assumptions nor our arithmetic counters establish measured end-to-end acceleration.

Table 2. Inheritance helps selectively. Complete three-generation histories, summed over three episodes per family. Every cell has nine requests. The rank-deficient family has three accepted programs and six certified nonidentifiability results; each other family has nine accepted programs.

Family	Fresh	Frozen	Accumulate	Conventional
Shared substructure	8,082	5,202	4,158	3,819
Disjoint targets	3,321	3,609	3,753	2,661
Changed identity/type	8,082	8,082	8,082	5,094
Rank-deficient	32,148	32,148	32,148	1,992

In shared-substructure episodes, later retention saves 1,044 operations beyond freezing the first generation. In disjoint episodes, accumulation instead adds 432 operations relative to fresh construction. Changed operator or type invalidates earlier qualifications, eliminating the reuse advantage. Rank-aware nonidentifiability checks avoid the expression search's failed search burden. These authored families establish no estimate of the frequency of gains in naturally occurring work.

An actual constructed lineage

In shared episode 1, the accumulating arm emits the following programs. Here y_i is primitive observation i under the same versioned six-dimensional operator; the names p_1, p_2, p_3 abbreviate serialized artifact identities.

$$\underbrace{p_1 = y_0 + y_1}_{\texttt{26ce34730345}} \quad \longrightarrow \quad \underbrace{p_2 = \mathrm{call}(p_1) + y_2}_{\texttt{96c34d77e688}} \quad \longrightarrow \quad \underbrace{p_3 = \mathrm{call}(p_2) + y_3}_{\texttt{422d62f096b0}}. \tag{7}$$

The retained certificates are respectively $(1,1,0,0,0,0)$, $(1,1,1,0,0,0)$, and $(1,1,1,1,0,0)$. The third program calls the second, which calls the first: dependency depth reaches two. Independent checks accept all three. Fully expanded execution takes one, two, and three additions per input, with no free macro discount. The conventional arm constructs identical artifacts in this episode. Thus the lineage demonstrates constructive succession while the comparator prevents it from being misreported as a unique advantage.

Accumulating retention writes 13,155 archive bytes and reads 13,870 across the campaign; these bytes are traffic measurements, not a sum of disjoint business expenses. Archive writes are included within produced-artifact writes. Nine invalidation events occur in accumulating and conventional arms. Maximum rational numerator or denominator bit length is six in the expression arms and seven in the conventional arm. The complete record retains phase counts, artifact identities, archive traffic, execution visits, and descriptive timing.

Retain useful constructions. Replace avoidable search with the strongest suitable method.

Within this exact linear domain, the conventional rank-aware compiler is the preferred route on the reported arithmetic measure. The frozen and accumulating comparisons still reveal a real conditional inheritance mechanism. They do not justify deploying a more expensive constructor merely because it has a more elaborate lineage.

3.4 What the finite result can establish

The study can establish that a declared constructor generated distinct accepted programs, that particular inherited calls participated in subsequent construction, and how this affected the specified cost measures. It can expose invalid retention, negative returns, or a superior conventional route. A tie with conventional compilation is useful evidence about the mechanism and a reason to use the conventional method.

All arms operate within supplied observation contracts. Retaining expressions cannot enlarge $\mathrm{rowspan}(A)$ or identify an erased distinction. The study therefore tests construction and inheritance, not discovery of a new physical sensing principle, industrial demand, or improvement of a learned inventor. Those stronger claims need the changed observation, complete activity, or fresh discovery campaigns specified later.

4 Invent the complete productive arrangement

Definition 1 (Productive arrangement)**.** *A productive arrangement specifies an intended beneficiary outcome, the means of producing and accepting it, the participants and their outside options, the terms of access and exchange, the financing schedule, and the allocation of remaining work and risk.*

This definition makes commercial design part of construction. A technically superior output can fail because verification is unaffordable, the beneficiary cannot integrate it, the producer must fund an excessive delay, or another provider can supply the same accepted result at better terms. An agreement can fail because the promised means do not exist. Joint search can change these interdependencies [3].

The unit of comparison is therefore a complete route to the same accepted outcome. The incumbent may use humans, conventional software, frontier agents, partners, different financing, or a process that avoids the proposed invention altogether. It must receive comparable material permissions. A restricted baseline can diagnose the effect of a particular restriction; it cannot establish superiority to competent unrestricted practice.

4.1 Purpose, formulation, and evaluation

Three objects govern a candidate. The **purpose** states the authorized outcome and material requirements. The **formulation** states the current representations, information procedures, dependencies, and means. The **evaluation instrument** supplies evidence about whether the outcome was achieved. A transformation may revise the formulation. If an evaluation instrument proves inadequate, it may be improved under an explicit bridge between old and new measures; the revision must not retroactively erase a failure or silently replace the purpose [1, 4].

Improving a method of checking a claim does not license weaker acceptance. These distinctions protect the meaning of an outcome while leaving its means open to invention.

4.2 The missing means may be a connection

Represent a candidate instrument by

$$m : (\text{inputs}, \text{state}, \text{resources}) \longmapsto (\text{output}, \text{new state}, \text{evidence}). \tag{8}$$

Its contract includes prerequisites, units, input and output semantics, authority, timing, costs, failure conditions, and invalidation rules. Two qualified instruments do not automatically form a qualified composition. Their clocks may disagree, their quantities may refer to different objects, or both may consume a resource that was counted only once. The connector that resolves such an incompatibility can be the enabling invention.

This calls for a graph with *joint prerequisites*: several parents can be necessary together. A measurement and a calibration rule may jointly enable an estimate; an estimate and an acceptance agreement may jointly enable delivery. Individual components can have little value before the bundle works. The system should therefore compare bounded compositions, including ones whose individual edits are locally unhelpful, while charging the search and integration costs.

4.3 A compact arrangement record

Each serious candidate needs a short, reviewable record: the beneficiary outcome; the strongest complete alternative; the suspected binding constraint; the executable construction; independent acceptance; participants, rights and refusal options; lifecycle costs and payment timing; and a named successor that could use the resulting asset. The record's function is to expose a consequential next experiment. It is not a requirement to grow an administrative framework before doing useful work.

5 The flywheel architecture

Figure 2 makes the causal proposal visible. Selection starts with consequential outcomes, finds a binding condition, and constructs or acquires means that could change it. Independent evidence determines what actually works. Delivery on viable terms can produce both reusable capability and settled surplus. Retention changes the starting state for subsequent work. Discovery itself is evaluated separately.

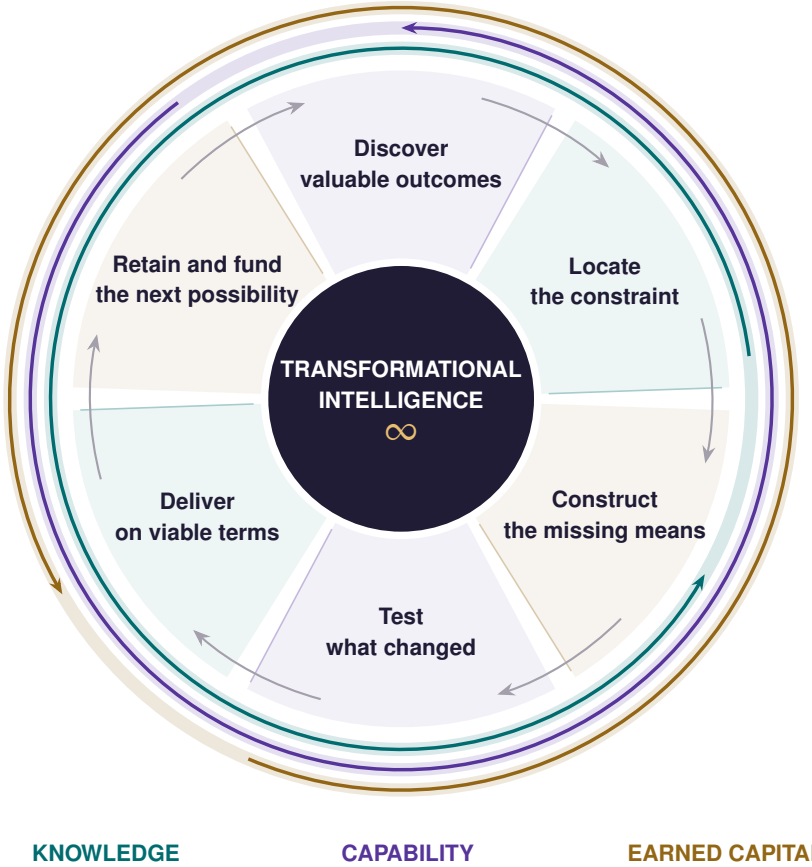


Figure 2. Three stocks, one testable succession. Knowledge improves selection; qualified means change what can be done; available earnings can finance the next action. The outer responsibilities may be revisited, combined, skipped, or performed in parallel. Arrows express hypotheses to test, not automatic growth. Every retained asset carries a scope; every new delivery requires current acceptance; every action requires current authority.

The architecture has three feedback paths. The **production path** improves a route to accepted output. The **economic path** selects viable exchanges and finances further work from permitted resources. The **discovery path** revises how useful arrangements are found and constructed. One path may improve while another stalls. Keeping them distinct permits a profitable service to continue while an unhelpful research method is retired.

5.1 A constructive search procedure

The difficult step in the wheel is making the missing means. A candidate begins with an accepted outcome and the strongest complete alternatives. The procedure identifies the alleged binding dependency and asks whether the outcome needs it, or only the inherited method does. It then proposes executable interventions: use an existing instrument, acquire an observation, repair an interface, change representation, reorganize production, or remove the dependency.

Structural correspondences propose candidates; verified objections turn into construction requirements. Each candidate declares inputs, prerequisites, actions, effects, acceptance, and resource use. A finite search

either produces a qualified route, returns a scoped obstruction, or ends without one. The system retains the instrument, its applicability, its constructor, and its failures separately. It can also choose the incumbent, ordinary delivery, or no entry.

In the running example, source and target share the relation between latent state, observation, and requested functional. The invariant is a reconstructible target in the observation row space. The missing bridge is either a compatible reconstruction program or additional informative observation. A type mismatch defeats program transfer; a null-space witness defeats inference from the existing observations. One objection calls for a new program, another for new information. Treating both as a request for more search would waste resources.

This is a revisable research procedure, not a claim that the complete autonomous loop is implemented. The finite study implements a specified constructor inside it. Choosing a valuable unsupplied target, acquiring a genuinely missing observation, and negotiating a viable activity remain additional work.

5.2 Inheritance with a boundary

A successor may inherit an executable instrument, evidence of its earlier behavior, an applicability model, and a constructor of related instruments. These are different assets. Historical evidence can support reusable generic checks, but the new mission still needs checks against its actual inputs, outputs, environment, and requirements. Permission is evaluated for the current action; it is not inherited from the existence of a tool [5].

Reuse should invalidate when relevant code, dependencies, measurement semantics, environment, or acceptance changes. The correct response can be a targeted recheck rather than a full replay, provided the dependency model is itself adequate. An archive that cannot explain why an item is applicable cannot yet support a strong inheritance claim.

For the prospective historical sensor, the *instrument* would turn specified traces into a calibrated estimate with uncertainty. Its *applicability model* would identify which clock alignment, channel semantics, and observation conditions permit that estimate. A *constructor* would adapt the measurement process and its discriminating checks to a different target. A better *discovery procedure* would find such useful constructions more effectively from comparable starting means. One successful reconstruction establishes none of the later claims by itself; each requires its own target and comparison.

5.3 Retention includes deletion

The enterprise retains compact, executable value: what worked, where it applies, what failed, and what would change the decision. Redundant traces, obsolete adapters, and misleading associations impose retrieval and maintenance costs. Removing them can improve the system. Failed candidates remain in denominators and accounts even when bulky intermediate files are replaced by a reproducible failure witness and a short lesson. Compression is compatible with honest accounting; erasing incurred costs is not.

6 Mechanisms that change the starting point

The OUT research collection offers complementary ways to construct missing means. Table 3 is a selection guide, not a mandatory sequence. A competent system chooses the mechanism that changes the current constraint, including a strong established method with no claim of novelty.

6.1 Transfer the relation, then test the bridge

Structural analogy maps relations rather than merely matching surface attributes [13]. A candidate transfer should identify the source mechanism, the corresponding target quantities, the required invariants, and a falsifying case. If a correspondence fails, the system can reject the transfer, construct a connector that

Table 3. From a binding condition to a useful construction. Each mechanism retains its source's assumptions and validation limits.

Mechanism	What it can change	Decisive question
OUT-LEARN [6]	Construct observations or calibration that separate relevant possibilities.	Does an independent reference support the new information?
OUT-THINK [7]	Build a representation that preserves required answers and updates with less work.	Which present and future obligations survive compression?
OUT-DESIGN [8]	Construct a replacement or transition that makes needed capacity usable.	Can the system actually reach and operate the replacement?
OUT-STRATEGISE [9]	Choose information and construction together, or act across remaining uncertainty.	Does the complete plan work under the declared possibilities?
OUT-EXECUTE [10]	Reuse compatible boundaries while recomputing changed parts.	Are the interfaces sufficient and current outputs accepted?
OUT-EVOLVE [11]	Build means that help construct a different successor capability.	What loses feasibility or value when the inheritance is removed?
OUT-REFLECT [12]	Construct evidence with inspectable corruption and correction assumptions.	What shared error or violated fault bound defeats the assurance?

establishes the missing condition, or redesign the route so that the condition is unnecessary. An analogy proposes a search direction; the target's acceptance evidence decides whether it works.

6.2 Turn a verified objection into a construction

A useful critic identifies a source-grounded omission, a contradiction, a boundary case, or a discriminating experiment. Another process adjudicates that objection before repair. A confirmed defect can require more than revising the answer: it can identify an unavailable observation, a defective evaluator, or an unnecessary dependency. The constructor then searches over changes that address the established cause.

Agent agreement is not independence of evidence. Several agents can share the same false source or missing observation. Their agreement may help inspect a derivation, but it does not manufacture external information. When observations cannot identify the requested quantity, the appropriate action is to improve observation, narrow an authorized claim, change production so the ambiguity ceases to matter, or stop.

Proposition 1 (An information boundary)**.** *Let two admissible worlds produce identical complete available observations but require different correct outputs. Any deterministic procedure using only those observations returns the same output in both worlds and therefore cannot be correct in both. Under an equal prior on two worlds with distinct exact answers, a randomized procedure whose randomness carries no additional world-dependent information has success probability at most one half.*

Proof. Identical observations induce the same output distribution. The probabilities assigned to the two distinct correct outputs sum to at most one. Averaging over the two worlds gives the bound. The deterministic statement is the corresponding special case. □

This elementary observation is useful because it directs invention toward the information channel rather than toward unsupported confidence. It is conditional on the stated worlds and observations; finding no affordable test does not establish that no test exists. The flagship paper's intervention example makes the same distinction and explicitly reports an unrestricted conventional planner that matches the constructed policy [1].

The hypothesis set is also open to challenge. Both initial explanations can be wrong; model misfit may require a third explanation or a different observation contract. Robustness over the current set remains conditional on its coverage of the relevant possibilities.

6.3 Produce evidence as part of producing the result

Dependencies, versions, assumptions, and acceptance conditions can travel with the output. This may reduce later reconstruction and make targeted revalidation economical. The benefit depends on completeness and correct interpretation of those records. A source pointer is not a proof, and a self-consistent provenance graph can still omit a decisive external fact. Evidence-producing construction should therefore be tested against deliberate omissions and changed dependencies, as well as ordinary successful cases.

7 What becomes reachable

Let the enterprise state be

$$x_t = (K_t, M_t, D_t, C_t, O_t, A_t), \tag{9}$$

where K_t is retained knowledge, M_t usable productive means, D_t evidence and exposure history, C_t cash, O_t obligations and commitments, and A_t current authority. The discovery procedure θ_t is separately specified. Richer equipment and a better search method are different interventions.

7.1 Three frontiers that should not be confused

For a fixed outcome contract o, validator V_o, and vector resource envelope B, define the **conditional continuation set**

$$\begin{aligned}\mathcal{R}_{\text{cont}}(x; B, o) = \{y : \exists \pi \in \Pi(o, A),\ y = \text{Out}(x, \pi),\ V_o(y) = 1,\\ \text{Cost}_{\text{continuation}}(x, \pi) \preceq B,\ \pi \text{ is fundable throughout its trajectory}\}.\end{aligned} \tag{10}$$

The installed means in x are given at this decision point. Their earlier construction cost has not vanished: it belongs in a whole-history comparison. Resource components can include time, money, energy, and storage; a justified exchange rate between them is not assumed.

The **initial complete-policy frontier** instead evaluates every admissible complete policy from a common initial state, including policies that construct tools, collect receipts, and later act. The **discovered frontier** is the subset actually found and qualified by a specified procedure within its search budget. A fourth observation, realized accepted execution, remains distinct from any feasible plan.

Proposition 2 (Construction is already inside a complete policy class)**.** *Suppose the admissible policy class is closed under valid sequential composition. A prefix p starts at x_0 and reaches x_t; a continuation q from x_t produces accepted output y. If $p; q$ respects the charter, authority, complete resource envelope, and funding path, then y is in the initial complete-policy frontier for that envelope.*

Proof. The composed policy $p; q$ is an admissible witness from x_0. Its construction and continuation burdens are both included, and its trajectory includes both funding paths. □

Consequently, inventing an instrument does not place its output outside an omniscient initial frontier that already permits inventing that instrument. It can change the conditional continuation set, improve a bounded procedure's ability to find a route, or install capacity that did not previously exist. These are substantial, testable changes. Comparing equal future budgets from unequal installed states answers a continuation question; it does not prove a gain after ignoring the history that created those states.

This correction sharpens *what becomes possible*. The intended claim concerns an actual situated enterprise, a declared resource envelope, and discoverable executable routes. Strict improvement beyond all prior routes requires an exhaustive argument or valid lower bound over a specified class. A favorable comparison with selected algorithms is a relative empirical result.

7.2 Accumulation can reduce capability

A new tool can add a route while its integration, selection, qualification, and maintenance consume a binding resource. Old interfaces may disappear. Therefore unconditional set inclusion is unwarranted.

Even a useful library can become costly to search. The finite study exposes some of these burdens through validation, invalidation, serialization, and expression-search costs.

For stochastic systems, define required acceptance probability and resource-risk conditions over a specified distribution. A favorable realized trajectory cannot establish reliable feasibility. Neither a higher tool count nor a monotone best result proves a better typical programme.

7.3 Four causal relations across generations

Reused by. A later task invokes a qualified existing instrument. This can amortize investment without creating a distinct capability.

Constructed with. Inherited means participate in constructing a different instrument. Remove the relevant inheritance, allow a strong alternative to rebuild or avoid it, and compare complete histories. A trace proves use; an intervention estimates contribution under the declared alternatives. The frozen-library arm helps separate later construction from perpetual reuse of the first tool.

Financed by. A source-bound record shows which settled earnings, seed funds, or financing receipts paid for commitments. Available earnings can fund a tool that subsequently loses. Funding is neither causal technical benefit nor commercial validation.

Discovered better by. A revised procedure finds better arrangements on fresh campaigns from matched resources, information, permissions, and means. Improvements attributable only to a larger inherited library belong to the equipment effect.

These relations can form a causal lineage, including jointly necessary parent sets. Complements and substitutes make benefits nonadditive: two assets cannot each be credited with the entire same gain.

8 Economics that can finance a next step

Let recognized operating revenue and any separately recorded recovery income over an interval be R, and the complete incurred expense under a declared accounting policy be X. Then $\Pi = R - X$; financing proceeds are excluded from R. Separately,

$$C_t = C_0 + \text{nonfinancing receipts}_t + \text{financing received}_t - \text{cash outflows}_t. \tag{11}$$

Nonfinancing receipts include settled customer collections and collected recoveries. Current available cash is $U_t = C_t - Q_t$, where Q_t is the *nonoverlapping* protected amount for custody, commitments, required buffers, and obligations that must be funded before further receipts. Exact category definitions belong to the account implementation. A restriction on cash is not automatically another expense, and the same obligation must not be subtracted twice.

8.1 When a new class of activity becomes viable

For $N > 0$ accepted deliveries, suppose a declared arrangement has construction burden F, variable cost v per delivery, and collected price s. Under these simplified immediate-cost assumptions,

$$\Pi(N) = N(s - v) - F, \qquad s_{\text{break-even}} = v + F/N. \tag{12}$$

If a retained constructor reduces the complete adaptation burden to $F' < F$, it lowers this threshold by $(F - F')/N$. A small activity priced between the two thresholds becomes viable in this model. This is the economic reason to care about a constructor: it may open many low-volume activities whose repeated fixed construction was unaffordable.

If the constructor was acquired earlier, F' is a continuation burden; its origin is charged in the whole-programme comparison. The inequality does not supply N, s, F', or accepted demand. The finite study's arithmetic counts cannot be substituted for those monetary quantities. A field evaluation must include acquisition, calibration, failed attempts, checking, integration, support, and continuing obligations, then test whether the constructor's reduction survives.

Technology can be reused without being consumed by each use, a central distinction in models of endogenous technological change [14]. Yet replicable knowledge and a private profit stream are different assets. Rivals can imitate, buyers can self-serve, and prices can fall. Physical inputs, authorized data, calibration, distribution, and reliable service can remain scarce. Productivity gains may accrue to users or other firms even when the originating enterprise captures little profit. An economic flywheel therefore needs a viable participation and value-capture arrangement as well as a useful technical inheritance.

8.2 Four separate decisions

Commercial viability asks whether accepted exchange can produce complete-cost profit under the stated conditions. **Investment value** asks whether an incremental use of resources is worthwhile against its alternatives. **Affordability** asks whether current resources can fund it while preserving obligations. **Successor benefit** asks whether the inherited means actually help the subsequent task. A business can pass the first while failing any of the others.

Positive profit does not imply affordable expansion. Revenue can be recognized before collection; reserves can restrict settled cash; an attractive investment can exceed the remaining balance. The correct response is to defer or redesign the investment and preserve the profit result. Conversely, positive cash financed by opening equity can coexist with losses. Counting either balance as proof of commercial success would hide the relevant constraint.

8.3 Complete cost and honest scope

The enterprise must account for search and failed attempts, acquisition, onboarding, production, inference and tools, verification, correction, integration, maintenance, support, payment losses, financing terms, and taxes or provisions where applicable. Actual human work is included when it occurs. A hybrid operation is labeled as such; normal buyer participation is separately disclosed. Access through an included subscription does not establish zero economic resource cost. Unmeasured costs remain unknown or enter an explicitly labeled sensitivity range.

Costs are recognized once under a consistent policy; development expense and asset depreciation must not double-charge the same investment. Funding sources and outcome attribution remain distinct. The existing native account is the operational authority; analytical views and pedagogical examples do not create another spendable balance.

8.4 Voluntary participation and strategic response

For a participant i, a modeled participation condition is

$$\mathbb{E}[u_i(\text{arrangement}) \mid \mathcal{I}_i] \geq \mathbb{E}[u_i(\text{outside option}) \mid \mathcal{I}_i]. \tag{13}$$

This is a design condition, not observed consent. Information, alternatives, payment timing, loss exposure, and the right to decline matter. A complete offer may need to change verification or risk allocation before any attractive price exists. Competitors and self-service can erode prices even when production is inexpensive.

The primary productivity benchmark compares competent frontier AI or agents with competent entirely human workers using conventional software on the same complete task. Human-plus-AI providers, autonomous rivals, self-service, and no purchase remain additional commercial alternatives. Human labor cost is not a guaranteed demand curve. Price, scope, acceptance probability, payment terms, capacity, and capital should be selected jointly rather than inferred from a fixed wage discount.

Claims of game-theoretic optimality must identify the game, information, feasible strategies, mechanism, participant responses, and objective. The bilateral-trade impossibility results of Myerson and Satterthwaite [15] show why desirable efficiency and incentive properties cannot simply be assumed to coexist under private information without the stated qualifications. A finite modeled optimum is useful when its assumptions apply; it is not a universally optimal enterprise.

8.5 The scale objective is an observer

For a fixed endpoint T after a full operating year, define

$$\Pi_{365}(T) = R([T - 365\text{d}, T)) - X([T - 365\text{d}, T)). \tag{14}$$

Revenue, valuation, financing proceeds, cumulative multiyear profit, and annualized short runs are different quantities. Report initial capital, received financing, remaining obligations, accounting policy, and simulation or actual-commerce status alongside this observer. Prospective capital can be chosen jointly with the arrangement; historical fixed-capital experiments retain their original conditions [16]. Neither more funding nor a larger numerical target creates demand, an informative measurement, or an accepted product.

9 A funding illustration: capability and cash

The following example isolates a possible complementarity between retained means and earnings. Its integer units and opportunities are authored for explanation. They are not market estimates, native account records, or evidence that a system discovered the arrangement autonomously. All costs in this finite model are immediate; no financing, taxes, depreciation, hidden maintenance, or obligations exist unless a case explicitly introduces them.

A predecessor starts with 10 units, spends 8 on its entire accepted delivery including instrument construction, and collects 14. Its profit is 6 and its cash is 16. The next accepted outcome would collect 20 after delivery. The cheapest route in the declared finite alternative set requires 18 units without the instrument and 15 with it. An unrestricted conventional constructor is also allowed to build the same instrument during the predecessor; it can tie the result.

Proposition 3 (A complementary funding threshold)**.** *Fix the accepted successor, timing, permissions, and declared alternatives. Let c_0 and c_1 be the minimum immediate funding requirements without and with an inherited instrument. Let b be reference available cash and $e \geq 0$ collected unrestricted predecessor earnings. With no additional financing, define $S_{ij} = \mathbf{1}\{c_i \leq b + je\}$ for instrument access i and earnings access j. If*

$$b < c_1 \leq b + e < c_0, \tag{15}$$

then $S_{00} = S_{10} = S_{01} = 0$ and $S_{11} = 1$.

Proof. The first inequality excludes both methods at cash b. The last excludes the no-instrument method at cash $b + e$. The middle inequality admits the instrument method at that cash. These are funding decisions; delivery still requires independent acceptance. □

Table 4. The same predecessor history, four successor access conditions. Quarantining earnings changes availability, not recognized profit. All amounts are illustrative units.

Instrument	Earnings access	Available	Needed	Admission
Withheld	Withheld	10	18	Defer
Retained	Withheld	10	15	Defer
Withheld	Available	16	18	Defer
Retained	Available	16	15	Fundable

The combined arm can spend 15 and, conditional on acceptance and collection, finish with $16 - 15 + 20 = 21$. Cumulative profit is $(14 - 8) + (20 - 15) = 11$, so final cash equals initial cash plus profit. The instrument's original construction is already included in 8; it is not charged again. The binary interaction $S_{11} - S_{10} - S_{01} + S_{00}$ is one. This is one threshold witness, not a general law of supermodularity or exponential growth.

9.1 Failure cases give the example its meaning

Withhold the 6 earned units from successor use and keep them explicitly restricted: recorded predecessor profit remains 6. Delay the 14-unit collection and, under the example's recognition rule, profit can still be 6 while cash is only 2. Reserve 2 of the settled 16 and only 14 remains available, again deferring the 15-unit action. If the funded successor fails acceptance after spending 15, it collects nothing; cash is 1 and total profit is -9. Failed work does not disappear.

If a conventional alternative needs only 15 without the inherited instrument, the alleged necessity of that instrument vanishes. If the conventional constructor creates the same instrument at the same predecessor cost, it matches the entire successful sequence. Both controls belong in the interpretation. The example demonstrates how capability and liquidity can interact; it establishes no unique advantage of AGI ALPHA and no improvement in the inventor.

The companion `examples/succession.py` evaluates these cases with exact arithmetic and checks cash conservation, preserved profit under cash restriction, and the conventional tie. It is a pure explanatory calculation and opens no business accounts. The four access treatments isolate a mechanism after a common predecessor. A separate whole-campaign comparison from equal initial resources is required to establish that discovering this arrangement was better than the strongest alternative.

10 How to test an improving inventor

An impressive output does not identify the cause of improvement. Table 5 separates claims that are often compressed into a single success label. These are evaluation categories, not a mandatory ladder for every useful transaction.

Table 5. The claim determines the experiment.

Claim	Required comparison or observation	What does not establish it
Useful construction	Current independent acceptance in its intended environment.	Code existence, an attractive diagram, or self-approval.
Commercial success	Accepted exchange, collections, complete incurred costs, and covered obligations.	Funding, revenue alone, or forecast demand.
Retained successor benefit	Fresh successor with and without inheritance; strong equal-content alternatives.	Reuse frequency or a later successful timestamp.
Better discovery	Fresh campaigns crossing discovery procedure and starting capabilities.	A better-equipped descendant alone.
Physical productive expansion	Installed, accepted output or capacity with lifecycle evidence.	A simulation or an engineering drawing.

10.1 First compare complete systems; then cross the components

The primary discovery test compares complete systems from matched initial histories and resource envelopes on fresh campaigns. It includes the cost of acquiring each system's means and revising its discovery procedure. This establishes an end-to-end result without first pretending that software and memory have a unique natural boundary.

A diagnostic factorial experiment then declares an executable interface. Let θ contain the search algorithm and its assigned parameters; let M contain the typed tool library and other assigned equipment. Explicitly assign learned abstractions, prompts, caches, data exposure, selection rules, and adapters to one component or a fixed common input. A procedural library can encode a search method, so silently placing it in either box changes the claim.

Let $Q(\theta, M)$ be a registered outcome vector, or a justified scalar, over the full discovery campaign. Compare:

	Original means M_0	**Retained means** M_1
Original procedure θ_0	$Q(\theta_0, M_0)$	$Q(\theta_0, M_1)$
Revised procedure θ_1	$Q(\theta_1, M_0)$	$Q(\theta_1, M_1)$

All four cells must execute through compatible interfaces under the declared intervention. Adapter development and qualification are charged. If a revised method cannot operate with the original means, the missing cell is not an automatic failure score; the decomposition does not identify the desired effect without a separately justified intervention.

With histories and other inputs controlled, the $M_0 \to M_1$ comparison under fixed θ estimates the equipment effect, and the $\theta_0 \to \theta_1$ comparison under fixed M estimates the assigned procedure effect. For a justified scalar, the interaction is

$$Q(\theta_1, M_1) - Q(\theta_1, M_0) - Q(\theta_0, M_1) + Q(\theta_0, M_0). \tag{16}$$

Its sign is empirical. The decomposition is relative to the declared architecture, not an intrinsic separation of intelligence from knowledge. The instrument-foundry study holds its supplied constructors fixed; it does not report this discovery-improvement experiment.

10.2 Three comparison boundaries

Purpose. Hold the complete outcome and material acceptance conditions common. If the purpose changes, register the new comparison. **Means.** Match initial resources, access, information, financing menus, and authority, and allow competent alternatives to construct different means. **Evidence.** Keep confirmation separate from exposed development data, include unsuccessful registered candidates, and evaluate the complete discovery burden.

Equal-content ordinary reuse is essential: the proposed retained method should not win solely because the control never received the same useful information. A direct specialist is essential when the task has strong known structure. An unrestricted complete alternative is essential when the claim concerns transformation rather than escape from an artificially restricted baseline. These comparisons can be expensive; the strength of a claim should match the comparison actually conducted.

10.3 Finite search and interpretable stopping

Before a confirmation campaign, fix its candidate population or generation rules, resource envelope, independent outcome checks, comparison arms, and stopping conditions. Include search and verification cost. Preserve the result when a budget is exhausted, prerequisites fail, or the incumbent wins. A subsequent campaign can revise the construction, but its new cases and resources are explicit. An endless search stopped at the first favorable result supplies neither a calibrated success rate nor evidence of reliable discovery.

Repeated tasks within one campaign can share causes and are not automatically independent samples. When uncertainty estimates are justified, use the campaign or another defensible unit of independence. Report exposure and selection. A model improved using a benchmark cannot treat its repeated exposure as fresh evidence of transfer.

10.4 An operating cycle that can choose not to invent

The practical cycle is to identify an important accepted outcome, compare complete routes, locate the binding condition, execute a finite discriminating construction, and evaluate both delivery and its costs. Collect only under the applicable acceptance and settlement rules. Preserve obligations. Retain useful instruments and short applicability lessons. Choose the next action from the actual updated state, including ordinary delivery, a different business, a new experiment, or holding resources.

This cycle is a policy interface, not an assertion that the entire research program is implemented. Repository components provide bounded mapping, construction, criticism, reuse, and native simulated accounting; their call paths and current limitations are documented publicly [16]. A named module does not prove that an autonomous agent discovered the method embodied in it.

11 What the public evidence currently supports

The evidence review is pinned to repository revision `e67aa0903158`. The three studies below answer different questions and their denominators are not pooled. They are reported published records inspected for this synthesis, not new market experiments or independent reruns performed by this paper.

11.1 A working arrangement with a conventional tie

Experiment 063 constructs accepted synthetic flexible-load planning artifacts, independently checks them, and connects them to simulated commercial accounting [17]. Individual subthreshold resources satisfy 0 of 4 technical requests; both ordinary unrestricted HiGHS pooling and the AGI ALPHA construction satisfy 4 of 4. A separately delegated generalist also constructs an accepted fresh two-event successor. These are computational planning results, not physical device operation.

The central authored account collects US$250, incurs US$179.35 of modeled total expense including its declared income provision, and reports US$70.65 net profit. Copycat and no-purchase scenarios each lose US$128. In a supplier-cost and failed-delivery scenario, US$2.40 profit does not satisfy the specified complete commercial acceptance. Actual demand and complete host/model billing remain unmeasured.

Crucially, the inherited successor route costs US$18 against US$3 for the ordinary route, with the same accepted work. The extra US$15 investment is funded from available modeled earnings, but its realized successor benefit is absent. The appropriate conclusion is to retain useful base delivery while rejecting that incremental investment claim. This record directly separates *financed by* from *helped by*; it does not establish a new industry.

11.2 A measurement witness that redirects invention

The next-series manufacturing study constructs a local observation operator and an ambiguity witness within declared elastic and residual-eigenstrain models [18]. In three authored interior geometries, two admissible constructed stress fields have central σ_{xx} values of -100 and $+100$ MPa while their declared full-support uniform through-direction averages vanish. The test exposes information absent from that particular measurement operator; it is not an observed 200 MPa error in a physical specimen.

The source application remains on hold because the intended local absolute-stress use needs a qualified measurement contract, calibration, matching spatial kernels, and independent same-specimen truth. No raw-row accuracy confirmation or physical qualification occurred. A learned reconstruction cannot recover a distinction that the assumed observations erase. The retained asset is a scoped identifiability instrument and a discriminating next measurement requirement, not a licensed physical service or measured commercial advantage.

11.3 Capital alone did not repair the finite portfolio

The R2 public portfolio reports 110 separately funded authored native scenarios over 36 modeled months [19]. The best final trailing-365-day profit is US$0 from non-entry; the worst is negative US$36,000. Neither the US$1 billion nor the US$3 billion profit milestone is attained. Twenty-three accounts retain uncovered funding conditions; arithmetic acceptance does not convert their obligations into funded success.

The scope matters as much as the result. Each account has eight authored prospects and at most three scheduled purchases per prospect across three years, with a maximum registered price of US$40,000. This

imposes a loose US$960,000 gross-contract bound per account over the entire interval. The US$1 billion initial-equity arm leaves demand unchanged. Consequently the study cannot validate a route to billions in annual profit, and its failure is not a proof that all possible businesses fail. It shows that enlarging capital without changing the limiting arrangement does not create the desired scale.

11.4 What these records change

The evidence supports usable computational constructions, accounting distinctions, strong conventional comparators, and precise reasons to reject or defer proposed gains. It supplies no demonstrated open-ended improvement of discovery, no physical industrial expansion, and no actual commercial profit for these experiments. This is actionable knowledge: stop buying a tool that loses, obtain the measurement that would distinguish the target, and seek a different complete arrangement when the registered market cannot express the objective.

12 Relation to prior work

The research does not originate transformational creativity, structural analogy, reusable programs, open-ended search, or recursive self-improvement. Boden's account distinguishes combination, exploration, and transformation of conceptual spaces [20]. Gentner formalizes relational structure mapping [13]. DreamCoder jointly develops reusable program abstractions and search capability [21]. POWERPLAY searches for tasks and solver modifications together while retaining previously acquired skills [22]. POET co-generates challenges and solutions and uses transferred solutions as stepping stones [23]. These precedents make task generation, abstraction, and transfer established starting points rather than new claims of this paper.

More recent systems directly investigate editable agents and improvement procedures. The Darwin Gödel Machine maintains and evaluates self-modified coding agents [24]; Hyperagents makes both task and meta agents editable and reports accumulating meta-level improvements [25]. SkillCraft evaluates formation and reuse of executable higher-level tool compositions across tasks [26]. These are close precedents for research on reuse and improving the inventor. AGI ALPHA studies how such improvement procedures could interact with accepted production, continuing obligations, and fundable succession. Those economic and physical outcomes require evidence beyond the reported agent benchmarks.

Within the AGI ALPHA series, the foundational substrate connects agents, jobs, validators, tools, memory, settlement, and governance [27]. *Transformational Intelligence* treats the formulation as an object of discovery [1]; *Invent the Economy* develops voluntary viable arrangements [3]; *The World as an Output* and *Beyond the Production Frontier* extend the means of discovery and production [28, 29]. The OUT collection supplies more specialized constructions with their own conditional results. This paper integrates those ideas into a single account of successive enterprise states and asks for separate evidence of funding, enablement, and improved discovery.

The contribution is therefore an operational synthesis and evaluation design: couple qualified inheritance to real resource constraints and complete productive arrangements, then follow the benefit through a successor. The elementary propositions clarify that synthesis, and the executed program-construction study exposes its technical inheritance mechanism and stronger conventional alternative. They are not presented as new mathematics, a new learning algorithm, or proof that this architecture dominates other organizations.

13 The next decisive demonstration

The running opportunity provides a concrete progression from this study to consequential work. A prospective measurement foundry first selects a valuable question that cannot be served affordably by its strongest complete alternatives. The target stays fixed; the means remain open.

Acquire the missing distinction. Identify existing traces or permitted new measurements and compare their information with the required quantity. A null-space or other ambiguity witness specifies what is missing. If no informative observation can be acquired, stop or propose a different authorized outcome. For retrospective work, evidence must already survive.

Build and qualify the instrument. Construct the actual observation and reconstruction process, including units, kernels, calibration, noise, and version boundaries. Use independently withheld external truth where applicable. For a physical process, install and test it; a simulator's correctness does not certify a specimen or sensor. Compare the full route with competent existing instruments, laboratories, software, and providers.

Deliver a complete activity. Test a real beneficiary's acceptance, participation, and payment terms within existing authorization. Count acquisition, construction, failed work, checking, correction, integration, and support. A finite research programme can contain funded losses: an enabling instrument need not have its own buyer. Its investment belongs to the programme and must be justified by observed or explicitly uncertain downstream value.

Construct a distinct successor. Choose a materially changed request, generate its instrument, and independently qualify it. Compare whole histories against reconstruction from scratch, frozen first-generation reuse, and the strongest conventional constructor. Retain the method that performs best under the outcome and resource criteria.

Test the inventor. Only after defining a revised discovery procedure and compatible component interventions should a fresh campaign claim that the system has become better at finding such transitions. Human selection, repairs, and other assistance remain visible in the evidence record. Increasing automation is a target to measure.

This programme can also discover that another complete arrangement is better: a cooperative use of existing resources, a revised interface or contract, or a different industry. A prospective capital choice is made jointly with that arrangement. More capital can fund a missing means; it cannot establish that the means works or that buyers value the result.

13.1 What infinity means

The infinity mark names an open-ended research horizon. New useful outcomes and means can enter the search; current industries and benchmarks do not define its final boundary. Each actual experiment still has finite resources and a fixed endpoint. Memory is selective, information is incomplete, and accumulation can have negative returns.

A bounded best-so-far score $q \leq q_{\max}$ cannot accumulate net gains of a fixed $\varepsilon > 0$ more than $\lfloor (q_{\max} - q_0)/\varepsilon \rfloor$ times. Long-run progress must be assessed through newly accepted capabilities and changing opportunity sets, with the burden of finding and sustaining them included.

The central hypothesis is that useful construction can reduce the burden of future useful construction often enough to sustain consequential succession. Search, qualification, maintenance, imitation, and resource scarcity may absorb those gains. The flywheel can accelerate, stall, simplify, or be replaced. Its direction is a result to observe.

14 Conclusion

AGI ALPHA FLYWHEEL asks an enterprise to make useful invention a repeatable productive activity. It searches for an important outcome, constructs what is missing, qualifies the result, and tests whether something valuable survives into a different task. The object of invention is the complete arrangement that makes the activity possible.

The revised construction study gives this proposal an executable centre: specifications become typed programs, certificates and fresh-input checks determine acceptance, and controlled libraries expose the effects of inheritance across generations. Established compilers remain full competitors. Construction over existing observations is separated from obtaining new information.

The formal and economic distinctions then determine what may be claimed. A continuation state can improve without expanding an omniscient initial policy frontier. A tool can help without making its constructor a better inventor. Productive value can grow without private profit, and profit can exist without currently available cash. Preserving these distinctions makes the larger ambition investigable.

The next consequential result must connect a genuinely useful activity, its qualified enabling means, a distinct successor, and complete costs. A further result must show improved discovery of those transitions. The US$3 billion objective remains ahead of the evidence. The programme advances by constructing the next necessary means and letting comparative evidence determine whether it deserves to survive.

Build the next starting point. Then test what it makes possible.

A Reproduce, inspect, and extend

The publication package contains editable LaTeX, a scholarly bibliography, the vector flywheel, the exact instrument-foundry study, the preserved funding illustration, and a public source map. Build from `docs/agi-alpha-flywheel` with `make`. A TeX Live installation with pdfLaTeX, BibTeX, the recommended fonts, and the listed LaTeX packages is sufficient. The standalone source archive can also be uploaded to a LaTeX editor with `paper.tex` selected as its main file.

Run the finite construction study and its independent acceptance controls with:

```
python3 examples/check_foundry.py
```

The public protocol was committed before substantive execution at `fa7a1117b913`. The original executable and first result are preserved. The portable replay compares every scientific field, including source, protocol, workload, artifacts, and costs, while reporting Python-version metadata and excluding variable timing. Its source and results are bound in the publication manifest. The workload is authored and public, not a concealed holdout. Runtime observations must be read with the declared cost scope.

The preserved funding illustration runs with:

```
python3 examples/succession.py -check
python3 examples/succession.py
```

This needs only the Python standard library. It performs no model calls, network requests, external transactions, or native account mutations. Its output is an illustrative model result, not a business receipt. The existing native financial engine remains the authority for repository operating experiments.

For the reviewed experimental records, use the exact source links in the bibliography and the commands in each original report. Reproducing their software may require their pinned dependencies and frozen source. This paper's example does not reproduce those experiments, validate physical claims, or turn the public records into live commercial operations.

A.1 Minimal evidence record for a future claim

A future connected demonstration should bind its intended outcome, complete alternatives, source revisions, exposure history, finite protocol and stopping rule, acceptance evidence, complete incurred costs, cash restrictions and obligations, and the actual successor. Record separately whether each link was funded, causally enabling, or evidence of improved discovery. Include the outcome that would invalidate reuse and the result of the strongest conventional route. These fields make a result inspectable without requiring disclosure of private documents, customer data, or execution credentials.

A.2 A note on authorship and evidence

This manuscript presents Vincent Boucher's AGI ALPHA research program. Included AI assistance was used for synthesis, drafting, code preparation, and technical checking. The finite constructor and workloads are authored software and specifications. Their executable outputs are measured within that domain; the study does not claim that an autonomous enterprise invented its own constructor or discovered industrial demand. Source-specific findings remain attributed to their public papers and records. Publication and reproducible arithmetic do not establish peer review, independent market validation, or completion of the broader program.

References

[1] Vincent Boucher. Transformational intelligence: Learning to change the game. Technical report, MONTREAL.AI, October 2026. URL https://github.com/MontrealAI/AGI-Alpha-Invention-Flywheel-Lab/blob/e67aa09031583f976a39c8406ae86585b02cde76/docs/research/transformational-intelligence-202610.pdf. Research manuscript.

[2] Vincent Boucher. OUT-ACCELERATE: Intelligence that accelerates the means of acceleration, 2026. URL https://github.com/MontrealAI/AGI-Alpha-Invention-Flywheel-Lab/blob/d6198584eeb0372564f1ae5cbfb50e1b288e508b/docs/research/out-accelerate.tex. 2 November 2026. Research edition 1.0; source snapshot d6198584eeb0372564f1ae5cbfb50e1b288e508b.

[3] Vincent Boucher. Invent the economy: Transformational intelligence for creating viable markets. Technical report, MONTREAL.AI, October 2026. URL https://github.com/MontrealAI/AGI-Alpha-Invention-Flywheel-Lab/blob/e67aa09031583f976a39c8406ae86585b02cde76/docs/research/invent-the-economy-202610.pdf. Research manuscript.

[4] Vincent Boucher. Beyond the original problem: Transformational intelligence for expanding what is possible. Technical report, MONTREAL.AI, October 2026. URL https://github.com/MontrealAI/AGI-Alpha-Invention-Flywheel-Lab/blob/e67aa09031583f976a39c8406ae86585b02cde76/docs/research/beyond-the-original-problem-20261019.pdf. 19 October 2026. Revised research manuscript.

[5] MONTREAL.AI. Transformational autonomous enterprises: Mission-relative revalidation for capability inheritance across four execution systems. Technical report, MONTREAL.AI, October 2026. URL https://github.com/MontrealAI/AGI-Alpha-Invention-Flywheel-Lab/blob/e67aa09031583f976a39c8406ae86585b02cde76/docs/research/transformational-autonomous-enterprises-v1.pdf. 23 October 2026. Research edition 01; prepared by Writer 4; SUCCESSOR Omega harness.

[6] Vincent Boucher. Out-Learn: Change what you need to know. Technical report, MONTREAL.AI, October 2026. URL https://github.com/MontrealAI/AGI-Alpha-Invention-Flywheel-Lab/blob/e67aa09031583f976a39c8406ae86585b02cde76/docs/research/out-learn-20261021.pdf. 21 October 2026. Revised research manuscript.

[7] Vincent Boucher. Out-Think: Transformational intelligence for inventing the space of reasoning. Technical report, MONTREAL.AI, October 2026. URL https://github.com/MontrealAI/AGI-Alpha-Invention-Flywheel-Lab/blob/e67aa09031583f976a39c8406ae86585b02cde76/docs/research/out-think-20261021.pdf. 21 October 2026. Research manuscript.

[8] Vincent Boucher. Out-Design: Creating the freedom to transform. Technical report, MONTREAL.AI, October 2026. URL https://github.com/MontrealAI/AGI-Alpha-Invention-Flywheel-Lab/blob/e67aa09031583f976a39c8406ae86585b02cde76/docs/research/out-design-20261021.pdf. 21 October 2026. Revised research manuscript.

[9] Vincent Boucher. Out-Strategise: Transformational intelligence for constructing the conditions of success. Technical report, MONTREAL.AI, October 2026. URL https://github.com/MontrealAI/AGI-Alpha-Invention-Flywheel-Lab/blob/e67aa09031583f976a39c8406ae86585b02cde76/docs/research/out-strategise-20261020.pdf. 20 October 2026. Research manuscript.

[10] Vincent Boucher. Out-Execute: Constructing the conditions of completion. Technical report, MONTREAL.AI, October 2026. URL https://github.com/MontrealAI/AGI-Alpha-Invention-Flywheel-Lab/blob/e67aa09031583f976a39c8406ae86585b02cde76/docs/research/out-execute-20261021.pdf. 21 October 2026. Revised research manuscript.

[11] Vincent Boucher. Out-Evolve: Building the means of discovery and production. Technical report, MONTREAL.AI, October 2026. URL https://github.com/MontrealAI/AGI-Alpha-Invention-Flywheel-Lab/blob/e67aa09031583f976a39c8406ae86585b02cde76/docs/research/out-evolve-20261020.pdf. 20 October 2026. Revised research manuscript.

[12] Vincent Boucher. Out-Reflect: Transformational intelligence for constructing self-correcting evidence. Technical report, MONTREAL.AI, October 2026. URL https://github.com/MontrealAI/AGI-Alpha-Invention-Flywheel-Lab/blob/e67aa09031583f976a39c8406ae86585b02cde76/docs/research/out-reflect-self-correcting-evidence-20261021.pdf. 21 October 2026. Substantively revised research edition.

[13] Dedre Gentner. Structure-mapping: A theoretical framework for analogy. *Cognitive Science*, 7(2):155–170, 1983. doi: 10.1207/s15516709cog0702_3. URL https://doi.org/10.1207/s15516709cog0702_3.

[14] Paul M. Romer. Endogenous technological change. *Journal of Political Economy*, 98(5, Part 2):S71–S102, 1990. doi: 10.1086/261725. URL https://doi.org/10.1086/261725.

[15] Roger B. Myerson and Mark A. Satterthwaite. Efficient mechanisms for bilateral trading. *Journal of Economic Theory*, 29(2):265–281, 1983. doi: 10.1016/0022-0531(83)90048-0. URL https://doi.org/10.1016/0022-0531(83)90048-0.

[16] MONTREAL.AI. AGI ALPHA flywheel: Developer instructions, 2026. URL https://github.com/MontrealAI/AGI-Alpha-Invention-Flywheel-Lab/blob/e67aa09031583f976a39c8406ae86585b02cde76/AGENTS.md. Operating direction at source snapshot e67aa09031583f976a39c8406ae86585b02cde76; consulted 2 November 2026.

[17] MONTREAL.AI. Constructing the productive arrangement: AGI ALPHA research edition 063, 2026. URL https://github.com/MontrealAI/AGI-Alpha-Invention-Flywheel-Lab/blob/e67aa09031583f976a39c8406ae86585b02cde76/docs/transformational063/REPORT.md. 1 November 2026. Repository report; source snapshot e67aa09031583f976a39c8406ae86585b02cde76.

[18] MONTREAL.AI. Next experiment series: Executed results and admission decision, 2026. URL https://github.com/MontrealAI/AGI-Alpha-Invention-Flywheel-Lab/blob/e67aa09031583f976a39c8406ae86585b02cde76/experiments/next_series_20261102_365c3d6c/REPORT.md. 2 November 2026. N3 model-instrument and measurement-scope results; source snapshot e67aa09031583f976a39c8406ae86585b02cde76.

[19] MONTREAL.AI. AGI ALPHA: R2 complete finite comparison, 2026. URL https://github.com/MontrealAI/AGI-Alpha-Invention-Flywheel-Lab/blob/e67aa09031583f976a39c8406ae86585b02cde76/experiments/public_portfolio_20261102/evidence/economics-r2/report/REPORT.md. 2 November 2026. Report of 110 separately funded authored scenarios; source snapshot e67aa09031583f976a39c8406ae86585b02cde76.

[20] Margaret A. Boden. Creativity and artificial intelligence. *Artificial Intelligence*, 103(1–2):347–356, 1998. doi: 10.1016/S0004-3702(98)00055-1. URL https://doi.org/10.1016/S0004-3702(98)00055-1.

[21] Kevin Ellis, Catherine Wong, Maxwell Nye, Mathias Sablé-Meyer, Lucas Morales, Luke Hewitt, Luc Cary, Armando Solar-Lezama, and Joshua B. Tenenbaum. DreamCoder: Bootstrapping inductive program synthesis with wake-sleep library learning. In *Proceedings of the 42nd ACM SIGPLAN International Conference on Programming Language Design and Implementation*. ACM, 2021. doi: 10.1145/3453483.3454080. URL https://doi.org/10.1145/3453483.3454080.

[22] Jürgen Schmidhuber. PowerPlay: Training an increasingly general problem solver by continually searching for the simplest still unsolvable problem. *Frontiers in Psychology*, 4:313, 2013. doi: 10.3389/fpsyg.2013.00313. URL https://doi.org/10.3389/fpsyg.2013.00313.

[23] Rui Wang, Joel Lehman, Jeff Clune, and Kenneth O. Stanley. Paired open-ended trailblazer (POET): Endlessly generating increasingly complex and diverse learning environments and their solutions, 2019. URL https://arxiv.org/abs/1901.01753v3. Version 3, 21 February 2019.

[24] Jenny Zhang, Shengran Hu, Cong Lu, Robert Lange, and Jeff Clune. Darwin Gödel Machine: Open-ended evolution of self-improving agents, 2025. URL https://arxiv.org/abs/2505.22954v3. Version 3, revised 12 March 2026.

[25] Jenny Zhang, Bingchen Zhao, Wannan Yang, Jakob Foerster, Jeff Clune, Minqi Jiang, Sam Devlin, and Tatiana Shavrina. Hyperagents, 2026. URL https://arxiv.org/abs/2603.19461v1. Version 1, 19 March 2026.

[26] Shiqi Chen, Jingze Gai, Ruochen Zhou, Jinghan Zhang, Tongyao Zhu, Junlong Li, Kangrui Wang, Zihan Wang, Zhengyu Chen, Klara Kaleb, Ning Miao, Siyang Gao, Cong Lu, Manling Li, Junxian He, and Yee Whye Teh. SkillCraft: Can LLM agents learn to use tools skillfully?, 2026. URL https://arxiv.org/abs/2603.00718v2. Version 2, revised 10 March 2026.

[27] Vincent Boucher. AGI ALPHA: A scalable substrate for intelligence organizations. Technical report, MONTREAL.AI, April 2026. URL https://github.com/MontrealAI/AGI-Alpha-Invention-Flywheel-Lab/blob/e67aa09031583f976a39c8406ae86585b02cde76/docs/research/agi-alpha-foundation-20260430.pdf. 30 April 2026. Research manuscript.

[28] Vincent Boucher. The world as an output: Constructive sufficiency for reusable discovery. Technical report, MONTREAL.AI, October 2026. URL https://github.com/MontrealAI/AGI-Alpha-Invention-Flywheel-Lab/blob/e67aa09031583f976a39c8406ae86585b02cde76/docs/research/the-world-as-an-output-20261020.pdf. 20 October 2026. Research manuscript.

[29] Vincent Boucher. Beyond the production frontier: Transformational intelligence from verified discovery to stellar-scale civilization. Technical report, MONTREAL.AI, October 2026. URL https://github.com/MontrealAI/AGI-Alpha-Invention-Flywheel-Lab/blob/e67aa09031583f976a39c8406ae86585b02cde76/docs/research/beyond-the-production-frontier-20261019.pdf. 19 October 2026. Research manuscript.