Persistent | Google Cloud

Mainframe Modernization for the Agentic Era

A decision framework for boards, CIOs and business leaders responsible for putting AI into production on top of systems of record.



Table of content

THE BOARDROOM QUESTION
What happens when an AI agent needs an answer from the system nobody is willing to change? 3

CHAPTER ONE
Access is not readiness 4

CHAPTER TWO
Decide by workload, not by ideology 6

CHAPTER THREE
Evidence over assertion 7

CHAPTER FOUR
The modernization engine: Google Cloud 8

CHAPTER FIVE
The accountability layer: Persistent 9

EVIDENCE
What this has looked like in practice 11

CHAPTER SIX
Measure what changes for the business 12

CHAPTER SEVEN
Is your core ready for the agentic era? 13

THE FIRST MOVE
Start small. Prove the path. Scale the system. 14

THE WIDER FRAME
Core, Context, Coordination 15

© 2026 Persistent Systems Ltd. All rights reserved.

THE BOARDROOM QUESTION

What happens when an AI agent needs an answer from the system nobody is willing to change?

The question has moved out of architecture reviews and into board conversations. Enterprises are being asked to put AI into production, improve customer and employee experience and show returns inside the fiscal year. The business rules, transaction history and operating context that AI depends on sit inside decades-old code, batch windows and tightly coupled data structures.

The familiar version of this problem is cost and it is now the less important one. The real exposure is that the AI agenda is moving faster than the core can safely support it, and the gap is widening in a direction that is expensive to reverse.

This playbook is built on a single claim: most enterprises will succeed at connecting AI to the mainframe and still fail to get value from it. Reaching the data was never the hard part. Knowing what the data means is.

> **An enterprise can buy a more capable model in weeks. It cannot recreate decades of undocumented business logic after the people who understood it have gone.**

We offer a decision framework, not a migration pitch. It sets out how to choose a destination for each workload, how to compress the work of understanding a complex estate, how to prove behavioral equivalence with evidence rather than assurance and how to make each wave of change fund and de-risk the next.

It draws on the joint Persistent and Google Cloud partnership for mainframe modernization: an AI-native engine that makes a legacy estate legible and a cutover provable, paired with the domain context, engineering integration and governance required to act on what it finds.



CHAPTER ONE

Access is not readiness

Nearly every large enterprise already has a plan for connecting AI to the mainframe. Almost all of those plans solve the wrong half of the problem.

The prevailing approach is sensible on its face. Expose the core through APIs, replicate priority data into the cloud, stream events outward and then leave the transaction engine alone, because it works. The reasoning is that the mainframe does not need to change if the enterprise can simply reach it. That reasoning holds when the consumer is human. A dashboard shows a number to an analyst who already knows what the number means, which caveats apply and when to distrust it. Three decades of institutional judgment sit between the field and the decision.

An agent has none of that. It acts on the value it receives, at machine speed, across thousands of cases and it produces a confident answer whether or not it has understood the field correctly.

What an integration layer cannot tell an agent

Exposing a system makes it reachable, not interpretable. The knowledge an agent needs in order to act safely is almost never present in the interface:

- Which business rules are current policy and which are vestigial workarounds that no one has dared remove.
- Which of the exceptions encoded in the code path are regulatory obligations and which are historical accommodations for a client relationship that ended years ago.
- What a field actually means today, after twenty years of overloading — a status code reused for three unrelated purposes across two lines of business is not an edge case, it is the norm.
- Where the transaction boundary sits, what is atomic and what happens to downstream state when a batch job fails halfway.
- Which data is authoritative and which is a convenience copy that goes stale between batch windows.

None of that lives in the data. It lives in code, in the batch schedule, in operational runbooks and, increasingly, in the working memory of a small and shrinking group of people. Every retirement in that group removes documentation that was never written down and no amount of API surface brings it back.

> **An agent connected to an uninterpreted core does not fail loudly. It produces plausible, confident, auditable decisions that turn out to be wrong at scale.**

Why AI programs plateau after the pilot

This is the mechanism behind a pattern many enterprises have already lived through. A pilot performs well because its scope is narrow enough that a human being can vet the context by hand. Production scope removes that safeguard, the volume of interpretation required exceeds what any team can check and the initiative stalls without a single component having failed.

© 2026 Persistent Systems Ltd. All rights reserved.

The usual response is to add another integration layer. Each one increases operational fragility without addressing the underlying deficiency and the estate becomes harder to reason about than it was before.

What this changes

If access were the constraint, integration would be the answer and modernization could be deferred indefinitely. Because meaning is the constraint, the sequence inverts. The work of recovering, documenting, validating and governing business logic is not a step that follows the decision to modernize. It is the thing that makes any subsequent decision — including the decision to leave a workload exactly where it is — defensible.

This is also why an AI-ready core is not a synonym for a relocated one. Plenty of workloads should stay on the mainframe. The test is not where the workload runs, rather, whether the enterprise can state, with evidence, what it does.

Google Cloud's August 2026 modernization guidance makes a compatible argument from the engineering side: this is not a COBOL-to-Java conversion problem. The hard constraints are tightly coupled data and logic, undefined application boundaries, complex transaction paths and the requirement to prove functional equivalence before cutover.



© 2026 Persistent Systems Ltd. All rights reserved.

CHAPTER TWO

Decide by workload, not by ideology

The binary between keeping everything and moving everything is obsolete. A future-ready estate is a deliberately designed portfolio in which different workloads reach different destinations for stated reasons.

Once meaning has been recovered, the destination question becomes answerable. Some capabilities should be retired outright, some need only to participate in modern channels, some benefit from a new runtime, some carry technical debt that must go while behavior stays identical. A few are strategically differentiating enough that the business capability itself should change.

Path	Choose it when	What it delivers
Retire	The capability is redundant, duplicated elsewhere or no longer creates value.	Removes cost and complexity permanently.
Augment	The core is stable and correct, but data and functions must participate in modern channels and AI workflows.	Unlocks value without relocation — valid only once the logic is documented.
Rehost or replatform	The workload benefits from a new runtime with limited functional change.	Improves economics and operations while containing scope.
Refactor	Business behavior must stay stable but technical debt has to be removed.	Preserves fidelity while restoring maintainability.
Rewrite	The application is strategically differentiating and the future experience must change.	Creates new capability rather than a cleaner replica.

> **The destination should follow the business outcome, the risk profile, data gravity and regulatory obligation. It should never follow a platform preference formed before the estate was understood.**

Augment deserves particular scrutiny, because it is the path most enterprises default to and the one most often chosen for the wrong reason. Leaving a workload in place is a legitimate strategic answer. Leaving it in place because nobody knows what it does is not a decision — it is a deferral and it accrues interest.



Where monolith-to-microservices earns its place

For workloads that do warrant structural change, decomposition is where the case moves from risk reduction to competitive advantage. Independently deployable services collapse the cost and elapsed time of routine change, because a single feature no longer requires regression-testing and redeploying an entire monolith. A defect is contained within one service rather than taking down the application. Recovery is faster because smaller services are quicker to roll back and redeploy. And consumption-based cloud economics replace a cost floor that rises with MIPS usage.

Enterprises running phased, parallel-validated migrations have reported MIPS consumption reductions in the region of 30% over a multi-year program without production outages. Although directional, a defensible number is the one built from the client's own capacity and licensing position.

© 2026 Persistent Systems Ltd. All rights reserved.

CHAPTER THREE

Evidence over assertion

The value of AI in modernization is not autonomous code conversion. It is the compression of the work of understanding, documenting, transforming and validating a complex estate, with human judgment retained at the points where being wrong is expensive.

Automation changes the economics of discovery in particular. Work that once took a specialist team months of archaeology can now be done in weeks, which makes it viable to understand an estate before committing to a direction rather than after. That single shift is what converts modernization from a leap of faith into a sequence of evidence-backed decisions.

It does not remove the need for judgment. A tool can extract a rule. It cannot tell the enterprise whether that rule is still commercially relevant, whether it encodes an obligation to a regulator, or whether removing it would breach a contract nobody has read since signing.

Stage	What Google Cloud supplies	What Persistent is accountable for
Discover	The Mainframe Assessment Tool reads the estate at scale — mapping dependencies, extracting business rules, generating documentation and identifying domain boundaries.	Setting scope and criticality, securing data access, and holding the evidence to a standard a risk committee will accept.
Decide	Portfolio clustering and modernization-path recommendations generated across the full estate rather than a sample.	Translating those recommendations into a destination per workload against business value, risk appetite and regulatory obligation.
Transform	Agentic workflows built with Gemini produce specifications, target models, backlogs, code and tests; deterministic modernization preserves external behavior exactly where fidelity governs.	Target architecture, rule validation, exception handling, acceptance criteria and integration across application, data, security and operations.
Prove	Dual Run processes live workloads through legacy and modern environments simultaneously and compares outcomes; Mainframe Connector moves and converts data for cloud services and progressive offload.	Setting tolerance thresholds, resolving exceptions, running the cutover gate and carrying named accountability for the go decision.
Scale	Reusable automation, patterns and knowledge capture across successive waves.	The operating model, value tracking, controls and the factory backlog that funds the next wave.

> **Neither half works alone. An engine without domain judgment automates the wrong decisions faster. Judgment without an engine is a consulting exercise that runs out of budget before it reaches production.**

© 2026 Persistent Systems Ltd. All rights reserved.

CHAPTER FOUR

The modernization engine: Google Cloud

Google Cloud brings AI reasoning, cloud platform and mainframe-specific products that attack the constraints named in chapter one directly — meaning, boundaries and equivalence — rather than treating discovery, transformation, data movement and cutover as four disconnected projects.

The distinction matters commercially. A toolkit gives an enterprise five products and leaves it to stitch the evidence together. An engine carries a shared evidence base forward, so what discovery learns about a business rule is the same artifact that validation later tests against.



Mainframe Assessment Tool: the meaning-recovery layer

MAT reads the estate and produces what most enterprises have never had — a current, machine-generated account of what the core actually does. Dependencies are mapped, business rules are extracted from code, documentation is generated where none survived and application domain boundaries are identified rather than guessed.

This is the capability that makes the argument in chapter one actionable. Work that once required a specialist team and months of archaeology becomes viable in weeks, which is what allows an enterprise to understand an estate before committing to a direction instead of after.



Agentic workflows built with Gemini: compressing the build

Once the evidence base exists, Gemini-powered workflows built with Gemini carry it forward into specifications, target architecture, user stories, implementation plans, transformed code and generated tests. Human review points are configured deliberately rather than assumed away, which is what distinguishes an accelerated program from an unsupervised one.



Deterministic modernization: change without interpretation

Some workloads cannot tolerate an interpreted rewrite. For these, deterministic modernization targets structural change and technical debt while preserving external behavior exactly. Fidelity becomes the governing constraint and the enterprise still exits with a maintainable estate.



Dual Run: proof before commitment

Dual Run is the capability most likely to change how a risk committee reads a modernization proposal. Live production workloads are processed through both the legacy and the modern environment at the same time and the outputs are compared. What emerges is a match rate, an exception log and a documented resolution trail — produced before anything is retired.

In practice this is what removes the big-bang outage window. Migration proceeds module by module, and each cutover is backed by demonstrated equivalence rather than a scheduled leap.



Mainframe Connector: data where the work is

Mainframe Connector copies and converts mainframe data for use by Google Cloud services such as BigQuery and Spanner, making core data available to analytics and new cloud-based functions while progressively offloading processing from the mainframe itself. It is the mechanism behind an augment strategy that is genuinely incremental rather than a parallel estate maintained indefinitely.

© 2026 Persistent Systems Ltd. All rights reserved.

CHAPTER FIVE

The accountability layer: Persistent

Tooling can reveal and transform code at remarkable speed. It cannot decide which rules still matter commercially, which exception is a regulatory obligation, where a phased cutover protects revenue, or how teams should operate on the other side of the change. And it cannot be held accountable for any of it.

Persistent operates as the strategic orchestrator of the program across four layers. Each answers a question the engine raises but cannot close.



Business context

Modernization choices are connected to industry workflows, regulatory obligations, customer journeys and value pools. This is what turns an extracted business rule into a decision — whether it reflects current policy, encodes an obligation to a regulator, or is a workaround whose original purpose ended years ago.

In banking, financial services and insurance that judgment rests on more than three decades of sector work and a global practice of over 7,000 BFSI practitioners. It also rests on knowing the hard constraints rather than discovering them at audit: DORA, Basel reporting, PCI-DSS, HIPAA, HL7 and FHIR, and real-time payment mandates are treated as architectural inputs with compliance sign-off built into the delivery model.

Engineering judgment

The right pattern is selected workload by workload, and the program is integrated across application, data, cloud, security, testing and operations rather than run as parallel workstreams that reconcile at the end. Portfolio decisions made in chapter two only hold if someone is accountable for the engineering consequences of each one.



Governance and assurance

Decision rights, evidence standards, human review points, cutover gates and value tracking are established before the first wave, not retrofitted when an auditor asks. Dual Run output is only as useful as the tolerance thresholds and approval discipline wrapped around it — and the sign-off record is what a regulator will eventually ask to see.



The modernization factory

A program designed as a single heroic migration concentrates risk at the end and delivers value nowhere else. Persistent runs modernization as a repeatable, governed pipeline instead, so each wave produces two returns rather than one. The first is the business outcome — lower run cost, faster change, improved resilience or a better customer and employee experience. The second is the intelligence retained: validated rules, mapped dependencies, regression tests, recorded decisions and reusable patterns that make the next wave faster and safer than the last.

© 2026 Persistent Systems Ltd. All rights reserved.

The second return is the one that compounds, and it is routinely left off the business case.

- Discover and baseline - estate map, extracted rules, dependency view, cost and risk baseline. Leadership learns what exists and where the highest-value move is.
- Prioritize and design - workload segmentation, target architecture, roadmap and governance. Leadership learns what will move, what will stay and why.
- Modernize in waves - refactored or reimagined services, modern data models, tests and APIs. Value arrives incrementally rather than at program end.
- Validate and cut over - dual-run evidence, exception resolution and an approval record. The cutover decision is evidence-led.
- Optimize and repeat - retired capacity, realized value, reusable knowledge and a funded factory backlog. Savings pay for the next wave.

Every wave should leave behind a business outcome and a body of validated knowledge. A program that delivers only the first is buying the same discovery twice.



Industry IP that modernizes the business function, not just the code

Alongside the platform engine, Persistent brings its own accelerators built for the workflows a modernized core has to serve — reusable solutions for claims adjudication, automated underwriting through the Intelligent Underwriting Workbench, and lending through Agentic Loan Origination. The Agentic BFSI Suite extends this with autonomous compliance and AML agents, iClaims for FNOL-to-payout claims handling, and iAURA, which automates data governance, quality reconciliation and natural-language querying across the BFSI data lifecycle.

These exist for the reason chapter one describes. A modernized system that reproduces a legacy workflow faithfully has moved technical debt, not removed it. Industry patterns are what make the target state fit for purpose rather than a cleaner replica of the thing being replaced.



© 2026 Persistent Systems Ltd. All rights reserved.

EVIDENCE

What this has looked like in practice

The argument in this playbook is testable. Three programs illustrate what recovering meaning, choosing by workload and proving equivalence produce in production.

Sector	What was delivered	Outcome
Tier 1 US bank	A 16-year application development and maintenance relationship spanning core-banking modernization and AFS to LoanIQ interface migration, including data integration across more than 150 systems during a major merger.	80% efficiency gain in data onboarding.
Leading wealth management firm	A reverse-engineering loop that reads COBOL, extracts business logic and rules, and forward-engineers them into object-oriented Java; GenAI-powered search deployed to lift agent productivity.	Business logic recovered from code as documented, reusable assets rather than lost to turnover.
US insurer	Cloud-native modernization of policy administration, with business rules extracted and rebuilt against current policy rather than reproduced as found.	68% of business rules automated.

The wealth management program is the clearest demonstration of the case this playbook makes. The deliverable was not a language migration. It was the conversion of undocumented institutional knowledge into an asset the enterprise owns, can audit and can hand to an agent — which is what an AI-ready core actually means.

The partnership behind it

- A 13-year Google Cloud partnership, operating under a Strategic Partner Agreement.
- 2025 Google Cloud Infrastructure Modernization Partner of the Year.
- More than 500 Google Cloud engagements delivered, with over 2,100 Google Cloud certifications, 150+ Google Gemini-certified partner associates and 1,800 Google-certified professionals.
- 10 Google Partner Expertise designations, including industry expertise in BFSI.
- In 2026, Persistent achieved seven Google Cloud competencies, including Gemini Enterprise, Infrastructure, Financial Services, Healthcare & Life Sciences, Manufacturing & Industrial, Application Modernization, and Chrome.
- Direct collaboration with Google's Centralized Mainframe Solutions Group, giving clients engineering support and tooling access for complex, regulated modernization programs.

© 2026 Persistent Systems Ltd. All rights reserved.

CHAPTER SIX

Measure what changes for the business

The strongest modernization business cases connect technical activity to a small number of executive outcomes and establish the baseline before transformation begins rather than reconstructing it afterward.

Outcome	Leading indicators	Business measures
AI readiness	Priority data and functions exposed through governed interfaces; documented rule coverage; context quality.	Time to deploy a new AI use case; share of target workflows supported.
Change velocity	Automated test coverage; deployment frequency; reusable components.	Release lead time; cost per change; time to market.
Operational economics	MIPS offload; license retirement plan; automation rate.	Run-cost reduction; cost per transaction; budget released for innovation.
Resilience and risk	Dual-run match rate; exceptions closed; documented logic coverage.	Production incidents; recovery time; audit findings; key-person dependency.
Customer and employee experience	Real-time data availability; straight-through processing; self-service completion.	Resolution time; abandonment; employee effort; satisfaction.



© 2026 Persistent Systems Ltd. All rights reserved.

CHAPTER SEVEN

Is your core ready for the agentic era?

Score each statement from 0 to 2. Zero means not true. One means partly true. Two means consistently true across the estate, not just in the best-run domain.

Dimension	Readiness statement	Score
Business priority	The core processes where modernization would unlock measurable growth, cost or risk outcomes can be named.	0 1 2
Estate understanding	Current visibility exists into application boundaries, dependencies, rules and owners.	0 1 2
Data and context	Priority data can be accessed with the timeliness, quality, lineage and permissions that AI workflows require.	0 1 2
Meaning and provenance	For priority workloads, the enterprise can state which rules are current policy, which exceptions are regulatory and what each critical field means today.	0 1 2
Portfolio choices	Workloads can be distinguished by whether they should be retired, augmented, replatformed, refactored or rewritten.	0 1 2
Validation	A credible method exists to prove behavioral and data equivalence before cutover.	0 1 2
Governance	Decision rights, human approvals, evidence standards and cutover gates are defined.	0 1 2
Continuity	Mission-critical core changes could still be executed if the primary legacy specialists were unavailable.	0 1 2
Operating model	Teams, skills, funding and value tracking can sustain multiple modernization waves.	0 1 2

Reading the score

- 0 to 6 - The AI agenda is almost certainly carrying hidden core risk. Start with discovery and a quantified baseline before committing to any destination.
- 7 to 12 - Foundations exist, but portfolio choices or assurance are inconsistent across the estate. Select a lighthouse workload and prove the method end to end.
- 13 to 18 - The enterprise is positioned to industrialize a modernization factory and scale by wave.

This diagnostic is a structured conversation starter. It is not an audit or a formal risk rating.

© 2026 Persistent Systems Ltd. All rights reserved.

THE FIRST MOVE

Start small. Prove the path. Scale the system.

The first commitment should buy clarity, not lock the enterprise into a destination.

Persistent and Google Cloud begin with a fixed-fee assessment and a lighthouse proof of value on a bounded scope, designed to replace assumptions with evidence before any larger decision is required.

- **Discovery and assessment** — a structured, fixed-fee engagement of four to eight weeks. The Mainframe Assessment Tool maps dependencies and extracts rules while Persistent sets scope and validates what the output means. Leadership receives a quantified business case and a heat-mapped, prioritized roadmap.
- **Executive review and validation** — current-state cost baselines, risk profile and modernization sequencing are presented for scrutiny. Leadership authorizes the next step against evidence rather than a proposal.
- **Lighthouse proof of value** — a selected high-impact workload is modernized through workflows built with Gemini and validated with Dual Run against live production behavior, under Persistent-defined tolerance thresholds. Leadership receives proof of functional fidelity, delivery economics and operating controls.
- **Strategic scaling** — the method, artifacts and lessons become a governed factory backlog with milestone gates and a value-tracking cadence. Leadership receives a wave plan and reusable assets.

Leave with four decisions, not another strategy deck

- Which workloads should move, change, connect or retire.
- Which business outcome will justify the first wave.
- Which evidence and controls are required before cutover.
- How the first wave becomes a repeatable modernization factory.

Book an AI-ready core assessment

A fixed-fee engagement of four to eight weeks that identifies where modernization unlocks priority AI use cases, quantifies the risk currently carried by undocumented logic and builds the economic case for the first wave.

© 2026 Persistent Systems Ltd. All rights reserved.

THE WIDER FRAME

Core, Context, Coordination

Mainframe modernization is not an end state. It is the first of three layers an enterprise needs before agentic AI does useful work at scale.

Core is the secure, maintainable foundation — the work this playbook describes. Context is the data fabric that gives an agent something meaningful to reason over. Coordination is the orchestration layer that lets agents act across workflows rather than answer questions inside one.

The sequence matters. Coordination built on unrecovered Core produces exactly the failure described in chapter one, at greater speed and wider blast radius. Enterprises that treat modernization as an IT cost exercise tend to fund Core and stop. Enterprises that treat it as an AI-readiness program fund all three and get compounding returns from each wave.

For a board, this is the reframe that matters: modernization stops being a line in the IT budget and becomes the precondition for the AI strategy already committed to.

About Persistent

Persistent Systems (BSE: 533179 and NSE: PERSISTENT) is a global services and solutions company delivering AI-led, platform-driven Digital Engineering and Enterprise Modernization to businesses across industries. With over 27,500 employees located in 21 countries, the Company is committed to innovation and client success. Persistent offers a comprehensive suite of services, including software engineering, product development, data and analytics, CX transformation, cloud computing, and intelligent automation. The Company is part of the MSCI India Index and is included in key indices of the National Stock Exchange of India, including the Nifty Midcap 50, Nifty IT, and Nifty MidCap Liquid 15, as well as several on the BSE such as the S&P BSE 100 and S&P BSE SENSEX Next 50. Persistent is also a constituent of the Dow Jones Best-in-Class World Index. The Company has achieved carbon neutrality, reinforcing its commitment to sustainability and responsible business practices. Persistent has also been named one of America's Greatest Workplaces for Inclusion & Diversity 2025 by Newsweek and Plant A Insights Group. As a participant of the United Nations Global Compact, the Company is committed to aligning strategies and operations with universal principles on human rights, labor, environment, and anti-corruption, as well as take actions that advance societal goals. With 468% growth in brand value since 2020, Persistent is the fastest-growing IT services brand in 'Brand Finance India 100' 2025 Report.

USA
Persistent Systems, Inc.
2055 Laurelwood Road Suite 210
Santa Clara, CA 95054
Tel: +1 (408) 216 7010
Email: info@persistent.com

India
Persistent Systems Limited
Bhageerath
402 Senapati Bapat Road
Pune, 411016
Tel: +91 (20) 6703 0000

Persistent
Re(AI)magining the World

www.persistent.com