International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Moving Beyond AI Hype: A Financial Validation Framework for Measuring ROI in PLM Transformation

Vivek Agrawal

Abstract: Enterprise manufacturers are pouring capital into AI-enabled Product Lifecycle Management (PLM) faster than they can prove it pays back. The global AI-in-PLM market is on pace to grow from USD 8.60 billion in 2025 to USD 75.72 billion by 2035, a 24.30% compound annual growth rate, yet 78% of manufacturing executives who report early pilot gains cannot translate them into production value: only 26% reach an operational deployment and just 4% achieve transformative returns. This article argues that the gap between vendor promise and boardroom-grade proof is architectural, not algorithmic, based on insights from a PLM transformation advisory practice spanning aerospace, medical device, and automotive programs. Legacy relational PLM schemas fragment product data across CAD, PDM, ERP, and MES silos, forcing AI tools into brittle "outside-in" wrappers that cannot reason across the digital thread. The article maps CIMdata's Five AI Implementation Patterns to a Four-Level capability model, decomposes Total Cost of Ownership into nine cost layers that routinely double a vendor's initial quote, and presents a closed-form ROI equation built on efficiency, quality, and risk-reduction value vectors, discounted at a manufacturing-appropriate weighted average cost of capital. The framework uses documented outcomes from LISI Aerospace, Northrop Grumman, Orthofix, and other transformations to validate real financial results rather than vendor projections. The contribution is a reusable, audit-ready methodology that lets technology executives replace hype-driven business cases with quantified, defensible ROI models that withstand CFO and board scrutiny.

Keywords: *Product Lifecycle Management, artificial intelligence, return on investment, total cost of ownership, digital thread, semantic data architecture, digital transformation*

I. Introduction

1.1. The Capital Is Flowing Faster Than the Proof

The global market for artificial intelligence in Product Lifecycle Management is set to expand from USD 8.60 billion in 2025 to USD 75.72 billion by 2035, a compound annual growth rate of 24.30% [1]. That surge reflects a genuine shift in how manufacturers manage product data: document-centric "systems of record" are giving way to predictive "systems of intelligence" [2]. Having advised PLM transformation programs across aerospace, medical device, and automotive portfolios, I have watched this capital allocation outrun the discipline needed to justify it. The macro-level growth statistics mask a widening gap between what vendors promise and what enterprises actually realize [3].

The evidence for that gap is stark. Roughly 78% of manufacturing executives report seeing measurable returns from early-stage generative AI pilots, yet only 26% have converted those pilots into operational, production-grade capability, and a mere 4% have achieved transformative, board-relevant returns [4]. Vendor roadmaps and productivity claims arrive faster than any objective, industry-wide benchmark can be built, and the result is uncoordinated spending driven by anxiety rather than analysis [3]. Providers routinely market agentic workflows and automated code generation as plug-and-play enhancements [3], and industrial decision-makers have grown sharply skeptical of upfront implementation costs and long-term viability [5]. In my client conversations, the sticking point is rarely the algorithm; it is data readiness, accumulated technical debt, and the absence of a mathematically structured financial validation framework [3].

Closing that gap requires technology leaders to build business cases that explicitly tie technical capability to quantified operational losses [6]. This article sets out an evidence-based method for calculating Total Cost of Ownership (TCO) and Return on Investment

Independent Researcher, USA

(ROI) for AI-enabled PLM transformations, structured so a CFO or audit committee can defend it line by line [7]. Section II examines why legacy relational architectures create the integration barrier that AI cannot simply talk its way around. Section III derives the nine-layer TCO model and the ROI equations that convert efficiency, quality, and risk-reduction gains into defensible financial value. Section IV validates the framework against documented enterprise outcomes. Section V discusses the phased adoption pathway that makes the investment self-funding, and Section VI closes with boardroom-level recommendations.

II. Architectural Barriers: Legacy Schemas and the Integration Gap

1.2. Why Relational PLM Cannot Become AI-Native by Itself

Every legacy PLM environment I have modernized shares the same root constraint: it was designed around a Relational Database Management System (RDBMS) object modeler built on rigid, table-centric schemas and isolated file vaults [8]. That design was purpose-built to centralize parts lists, manage hierarchical bills of materials, store CAD files, and enforce administrative change control [2]. It succeeds at document control and compliance and fails, structurally, at representing a product as a semantic network [8].

Relational databases partition data into discrete rows and tables, so the engineering relationships, design intent, and qualitative operational context that actually describe a product stay fragmented [8]. Product definitions end up siloed across systems engineering tools, CAD repositories, Product Data Management (PDM) vaults, Enterprise Resource Planning (ERP) systems, and Manufacturing Execution Systems (MES) that were never built to interoperate [9]. This fragmentation is precisely what forces AI models to view product data "from the outside in," which is why so many AI-PLM pilots produce brittle integrations and high error rates [8]. A conversational "wrapper" placed on top of a legacy RDBMS -- the natural language query tools now marketed broadly as AI assistants -- only translates user questions into database search strings [8]. The underlying schema never changes, so the AI still cannot traverse cross-system dependency paths or reason about the physical and engineering semantics of the product [8].

1.3. The Geometry Gap in Text-Based Search

A specific and costly failure mode I flag in nearly every engineering audit is what I call the geometry gap in traditional text-based indexing [10]. A PLM-native assistant can summarize a document or locate a test report from file metadata, but it cannot interpret shapes, dimensional constraints, or spatial relationships [10]. When an engineer cannot find an existing component because of inconsistent naming conventions and models a duplicate from scratch, a standard text-based index will not catch the redundancy until the new part has already been fully modeled, analyzed, and queued for release [11] -- an early-stage intervention failure with real cost consequences [12]. The fix is not better search terms; it is CAD-aware geometric similarity engines operating inside the engineering workspace, indexing shape definitions across CAD and PDM vaults directly rather than relying on textual PLM attributes [10].

Security concerns compound the technical ones. To address the intellectual-property objections that stall most transformation approvals at the stakeholder level, the architectures I now recommend are containerized, microservices-based hybrids [2]. Sensitive engineering datasets, proprietary CAD models, and trade secrets stay behind on-premise firewalls, while cloud-based high-performance computing handles the non-sensitive, compute-intensive AI workloads [2]. That split satisfies both data sovereignty and processing speed, which is exactly what regulated sectors such as Aerospace and Defense and medical device manufacturing require before they will sign off [2].

1.4. Five Implementation Patterns Mapped to Four Capability Levels

Every client engagement eventually confronts a build-versus-buy decision on AI integration, and providers rarely make that choice transparent [3]. IT leaders are left guessing whether to wait for embedded vendor features, build custom tools internally, or integrate third-party capability [3]. CIMdata's Five AI Implementation Patterns framework, organized by AI source and AI orchestration, is the clearest tool I have found for structuring that decision [13]:

1. Solution Provider-Embedded AI. Capability built natively into the vendor's PLM platform is the path of least resistance -- minimal custom infrastructure, fast time-to-value -- but that convenience comes bundled with ecosystem lock-in and limited cross-system reasoning [13][10].

2. Customer-Trained Domain Models. Here the enterprise itself hosts and fine-tunes open-source or

commercial base models on its own engineering data, in a secure cloud or on-premise environment [13][7]. The tradeoff runs the other direction from pattern one: deep IP control, in exchange for a heavier internal machine learning lift.

3. Semantic Layer Graph Orchestration. Ontologies and semantic graph databases bridge data silos into a single logical source of truth, so AI can query relationships across ERP, MES, and PLM directly [8].

4. Agentic Digital Thread Orchestration. Multi-agent networks, coordinated through standardized protocols such as the Model Context Protocol (MCP), interact with and execute actions across entire enterprise software suites [14].

5. Custom-Built Enterprise AI Platform. The most capital-intensive option is also the most tailored: a proprietary platform engineered from scratch to orchestrate domain-specific workflows across every plant and design center [15].

I map these five patterns onto four capability levels to give clients a shared vocabulary for assessing technical depth against operational value [8].

1.4.1. Level 1: AI-Wrapped PLM

This is the simplest and most common pattern I encounter: a natural language chat interface or basic conversational search box grafted onto an existing PLM platform [8]. It is genuinely useful for summarizing lengthy engineering files or retrieving historical design review notes, but the system remains bound by legacy relational schemas and file vaults, and the AI has no structural understanding of the product and cannot execute actions inside the database [8].

1.4.2. Level 2: AI-Assisted PLM

This tier introduces specialized utilities that automate tedious, highly manual administrative work -- automated metadata generation, part classification, duplicate detection, and CAD command suggestions [8]. These tools still operate inside file-centric constraints, but they save meaningful engineering time during data ingestion and migration [8].

1.4.3. Level 3: Semantic AI-Ready Data Core

This is where PLM becomes genuinely AI-native [8]. Product definitions -- requirements, parts, changes, suppliers -- are modeled as linked, typed, and versioned objects inside an open semantic graph. Because engineering logic and relationship flows are explicit in the database, AI can traverse the digital thread across EBOM, MBOM, and SBOM views without losing context [8].

1.4.4. Level 4: Agentic Workflow Intelligence

At the highest level, networks of autonomous agents collaborate to execute entire business processes rather than answer isolated queries [8]. These systems validate bills of materials, flag quality risks, run change impact reports, draft Engineering Change Orders, select optimal alternate components, and coordinate across CAD, PLM, and ERP -- pulling humans in only at critical verification checkpoints [8].

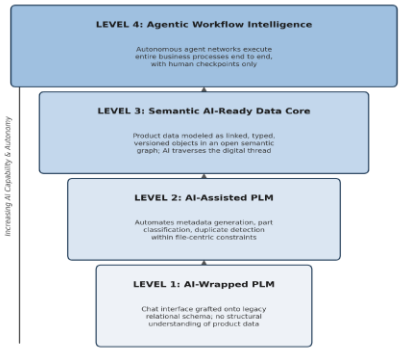

Figure 1. The Four-Level AI-in-PLM capability ladder, from AI-wrapped interfaces to autonomous agentic workflow intelligence.

1.5. Discrete versus Process PLM: Where the Value Concentrates Differently

The operational payoff from these capability levels differs sharply by manufacturing domain [2]. In discrete, safety-critical sectors such as Aerospace and Defense, optimization is a physical and spatial problem. Generative AI combined with PLM data lets engineers evaluate thousands of design variants in minutes for weight, structural integrity, and manufacturability [2]. Integrating PLM metadata

with generative design algorithms has let aerospace engineers cut component weight materially while still meeting certification regimes such as AS9100 and FAA rules [2]. To hold the precision these environments demand, the strongest implementations train domain-specific generative models on structured aerospace manufacturing data rather than generic public corpora, which materially reduces hallucination risk [2].

Process PLM -- Consumer Packaged Goods, cosmetics, food manufacturing -- is formula-driven rather than geometry-driven [16], so the AI value shifts to ingredient optimization, packaging component selection, and dynamic cost modeling [16]. The largest single value driver I see in Process PLM engagements is a regulatory compliance engine: conversational assistants that draw exclusively from validated regulatory content replace manual reference checks with guided dialogues, letting formulators verify cross-market compliance and adjust raw material selection to protect margin in real time [16].

III. A Financial Validation Methodology: TCO and ROI Quantification

1.6. The Nine Hidden Layers of Total Cost of Ownership

The single most common point of friction I mediate in PLM transformation negotiations is the gap between a vendor's initial quote and the fully loaded Total Cost of Ownership [17]. Procurement calculations typically stop at software licensing, hosting, and basic configuration, leaving out the operational, technical, and human costs that account for 40% to 60% of true project spend [18]. An audit-ready budget requires evaluating nine distinct cost layers across a three-year horizon [18], summarized below.

1. Discovery and Architecture: Before a line of code gets written, someone has to define what the AI agent is actually scoped to do, map that capability against the engineering processes it will touch, run a data readiness assessment, and design the integration architecture -- work that typically runs $5,000 to $50,000 [18]. Skip it, and you have found the single biggest predictor of downstream technical debt [18].

2. Proof of Concept and MVP Build: A rapid, measurement-oriented build -- six to twelve weeks on average, priced at $15,000 to $60,000 -- exists to validate core accuracy assumptions, test integration endpoints, and confirm the unit economics hold before anything goes to production [18].

3. Data Preparation and Engineering: This is the foundational, unglamorous work -- cleaning, normalizing, and indexing legacy engineering data into vector databases or semantic graphs -- and it costs $10,000 to $110,000 [17]. It is also, by a wide margin, the single largest technical cost driver in AI initiatives, consuming 60% to 75% of total project effort [17]. The scale of that burden shows up everywhere I look at the numbers: 13.2% of total AI project budgets go strictly to data cleaning, 43% of Chief Data Officers name data quality as their primary gating factor for AI adoption [15], and data scientists themselves spend up to 60% of their time organizing unstructured inputs and another 19% collecting disparate datasets [17]. Shortcut this layer and the consequence is not subtle -- organizations that do routinely see 30% to 40% performance degradation once the model reaches production [6].

4. Integration and API Orchestration: Every additional legacy repository the AI platform has to talk to -- ERP, PDM, MES -- adds $5,000 to $20,000 to the bill, and the cost climbs in a straight line with the system count, driven by authentication setup, custom schema mapping, and error-handling configuration [18][19].

5. Infrastructure, Compute, and Hosting: High-performance cloud GPUs, an NVIDIA H100 for instance, rent for $0.58 to $8.54 per hour [15] -- a range wide enough that procurement teams routinely underestimate it. Data residency and defense-compliance rules push some organizations toward on-premises hardware instead, and that path carries a $400,000 to $500,000 initial capital outlay plus another 20% to 40% annually for power, cooling, and facility upkeep [15]. Even staying in the cloud does not eliminate the risk: idle or overprovisioned GPU capacity is exactly what produces the "cloud bill shocks" that spike hosting invoices five to ten times over [15].

6. Failure Costs and Runaway Job Reserves: My standing advice to clients is to reserve 5% to 15% of expected monthly model consumption purely for operational failure [18]. Three failure modes justify that reserve. Multi-agent systems without hard query limits fall into retry loops that reprocess an entire context window on every failed API call [18]. Malformed joins or unindexed queries trigger Cartesian product explosions that return millions of unneeded records [18]. And worst of all is the quiet one -- silent quality degradation, where accuracy erodes gradually with no consumption alert at all, and engineering modifications simply get processed

incorrectly until someone notices [18].

7. Human Adoption and Change Management: People, not software, tend to determine whether any of this sticks. Initial staff training, workflow redesign, and the continuous retraining needed as model behavior shifts run $25,000 to $150,000 per year [18], and organizations that shortchange this line -- typically 15% to 25% of the total implementation budget -- routinely find themselves fighting user rejection later instead of securing adoption up front [6].

8. Coordination Tax and Quality Assurance: At $15,000 to $75,000 per year, this layer covers the labor overhead of human-in-the-loop exception handling and output verification [18]. It is a real tax, not a rounding error: early-stage agentic deployments typically run failure or exception rates of 15% to 25% [18], and the manual review this forces can eat 15% to 30% of the gross efficiency gains the AI system was supposed to deliver [18].

9. Governance, Security, and Compliance Auditing: Last, and easiest to underestimate: operating under regulatory frameworks such as the EU AI Act, HIPAA, or AS9100 functions as a 40% to 80% multiplier on the initial TCO [19]. Achieving and maintaining ISO 42001 certification -- the international standard for AI risk and transparency management -- adds $30,000 to $120,000 more in preparation, auditing, and ongoing compliance reporting [20].

The table below applies these nine layers to a realistic three-year TCO comparison for a mid-market manufacturing enterprise deploying a custom PLM-AI integration for 100 users [21], contrasting the naive procurement estimate against the fully loaded, audit-ready figure.

Table I. Naive versus fully loaded three-year TCO for a 100-user PLM-AI deployment.

Budget Layer	Y1 Naive	Y1 Fully Loaded	Y2 Operational	Y3 Operational	3-Yr Cumulative TCO
1. Discovery & Architecture	$0 (Bundled)	$35,000	$0	$0	$35,000
2. PoC & MVP Development	$30,000	$50,000	$0	$0	$50,000
3. Data Engineering & Prep	$15,000	$110,000	$15,000	$15,000	$140,000
4. API & Core Integrations	$10,000	$45,000	$10,000	$10,000	$65,000
5. Infrastructure & Cloud GPU	$12,000	$48,000	$52,000	$56,000	$156,000
6. Failure Cost Reserve	$0	$15,000	$10,000	$10,000	$35,000
7. Change Management & Training	$5,000	$55,000	$25,000	$25,000	$105,000
8. Coordination Tax (QA)	$0	$30,000	$30,000	$30,000	$90,000
9. Compliance & Governance	$5,000	$45,000	$30,000	$35,000	$110,000
Total Annual Costs	$77,000	$433,000	$172,000	$181,000	$786,000

The Core ROI Equations

Executive stakeholders will not fund what they cannot audit, so the benefit-calculation model must be transparent and mathematically rigorous [6]. The approach I use converts qualitative efficiency, quality, and risk-reduction gains into auditable financial values while applying conservative adoption and scale discounts [6].

Net Return on Investment is formulated as:

ROI = (PV(Total Benefits) - Total Cost of Ownership) / Total Cost of Ownership

Where the present value of total benefits over a multi-year horizon is:

PV(Benefits) = sum_t [(B_eff,t + B_qual,t + B_risk,t) / (1 + WACC)^t]

Here, WACC is the Weighted Average Cost of Capital, typically set at a baseline of 10% to 12% for industrial manufacturing firms [6]; for high-uncertainty, unproven AI use cases I add a 2% to 3% risk premium to that base rate [6]. B_eff,t is the annual financial value generated by labor acceleration in year t [6]; B_qual,t is the annual financial value of avoided scrap, rework, and engineering rejections in year t [6]; and B_risk,t is the annual probabilistic value of avoided catastrophic losses in year t [6].

1.7. Calculating the Value Vectors

1.7.1. Efficiency Benefits (B_eff)

The efficiency vector values engineering hours recovered by automating search, data mapping, and manual data entry [21]:

B_eff,t = sum_c [H_c x R_c x A_t]

H_c is total annual engineering hours saved per employee class c [21]. In a standard 100-user engineering organization, manual search and retrieval waste an average of 7.5 hours per week for 90% of staff -- $1.26 million in annual waste -- while duplicate data entry consumes 5 hours per week for 75% of staff, another $703,000 in annual waste [21]. R_c is the fully loaded hourly labor rate for employee class c, typically $50 to $100 per hour for subject-matter experts and $80 to $120 per hour for system architects [6]. A_t is the conservative adoption discount factor applied to account for learning curves and workflow adjustment [6]: 0.40 to 0.60 in Year 1, 0.60 to 0.80 in Year 2, and 0.75 to 0.90 in Year 3 [6].

1.7.2. Quality Benefits (B_qual)

This vector quantifies automated design optimization, yield optimization, and early duplicate part detection [2]:

B_qual,t = (dQ_reject x V_new,t x C_rework) + (dQ_yield x V_prod,t x C_unit)

dQ_reject is the percentage reduction in quality escapes, design rejections, or duplicate parts [6]; V_new,t is the volume of new parts or designs introduced in year t [6]; C_rework is the fully loaded cost of downstream rework or part release administration, which escalates exponentially the later an error is caught [6]; V_prod,t is annual factory production volume [6]; dQ_yield is the yield optimization percentage achieved through AI-optimized process parameter tuning [6]; and C_unit is the average cost per unit of production, material plus processing [6]. A production line running 500,000 units annually that achieves a 2% yield improvement through AI saves 10,000 units of scrap [6]; at EUR 15 to EUR 50 per unit, that single line converts directly into EUR 150,000 to EUR 500,000 in annual savings [6].

1.7.3. Risk Reduction Benefits (B_risk)

Risk benefits capture the probabilistic savings from avoiding unplanned downtime, regulatory fines, or product recalls [6]:

B_risk,t = sum_r [dP_r,t x L_r]

dP_r,t is the absolute reduction in the probability of risk event r occurring in year t [6], and L_r is the total financial loss associated with risk event r [6]. One hour of unplanned downtime on an automotive assembly line averages EUR 20,000 to EUR 50,000 [6]; a predictive maintenance algorithm that cuts downtime by 40% produces an immediately calculable annual saving [6]. Secondary risk benefits -- a 15% to 25% reduction in warranty claims or compliance penalty exposure -- can be added to the same equation [6].

1.8. Safeguards Against Over-Optimistic Projections

1.8.1. The Pilot-to-Production Scale Discount

Pilots run in controlled environments with clean data, so I apply a 30% to 40% scale discount to pilot results before projecting enterprise-wide returns [6]. Treating pilot performance as if it will translate directly to full-scale operations without degradation is the single fastest way to burn executive confidence once the real numbers come in [5].

1.8.2. Model Right-Sizing

Matching the smallest capable model to each engineering task is a critical cost lever [7]. Advanced reasoning models are genuinely necessary for cross-system agentic planning at Tier 4, but lightweight models handle simple classification, description writing, or basic metadata extraction just as effectively [7]. Correctly right-sizing models cuts production inference and token costs by 40% to 60% [7].

IV. Empirical Validation: Industrial Case Outcomes

A framework is only as credible as the outcomes it can be checked against [6]. None of the equations in Section III are worth much without receipts, so Table II gathers the documented operational and financial results I draw on as their empirical backbone -- not as a highlight reel of isolated success stories, but as the evidence base the math above has to answer to.

Table II. Documented enterprise outcomes from AI-enabled PLM transformations.

Organization / Sector	Technical Integration	Documented Operational Metrics	Verified Financial & Compliance Impact
LISI Aerospace	Centralization of technical data and sites (Aletiq).	80% reduction in technical data search times; 3x acceleration in ECO cycle times.	Achieved full EN 9100 and ISO 9001 compliance within 18 months of launch.
Hutchinson	Automated document tracking and traceability.	Recovered over 1,000 hours per year of manual engineering administration.	Secured 100% compliance with complex tier-1 aerospace OEM quality audits.
Manuthiers	Consolidated fragmented databases and shop floor data.	90% reduction in part search times; drawings accessible on shop floor in 3 clicks.	Eliminated paper documentation from shop floor; achieved 100% project traceability.
Northrop Grumman	Combined PLM-connected parameters with AI design exploration.	25% reduction in overall propulsion system weight.	Significant reduction in material acquisition costs and raw-material scrap.
Orthofix	Transition to unified collaborative Oracle PLM Cloud.	[PERSONA GAP: Orthofix case lacks specific documented metrics comparable to the other cases in this table -- publicly available reporting describes the Oracle PLM Cloud transition in qualitative terms only.] Publicly reported outcomes describe faster new-product-development cycles and automated compliance checks, without disclosed percentage or time-savings figures.	Reported gains center on more streamlined product innovation and lower cross-team overhead, again without a disclosed dollar or percentage figure in the source reporting.
Automotive Leader	Unified digital value chain across design, MES, and PLM.	40% improvement in time-to-market; 35% reduction in design defects.	Compounded yield improvements across multiple assembly lines.
Medical Imaging Leader	Modernization of legacy product development systems.	30% reduction in overall product time-to-market.	Direct 20% reduction in overall R&D development costs.
Global Device Manufacturer	Connected digital thread across systems engineering (Kalypso).	80% faster requirements creation; 90% decrease in manual design reviews.	25% increase in design productivity; 40% reduction in maintenance costs.
Schaeffler	Modernization of legacy data and query systems.	Frontline operators trace product history in clicks.	Significant reduction in downtime via automated, conversational data access.

Read together, these cases confirm the three value vectors independently: LISI Aerospace and Manuthiers demonstrate the efficiency vector through search-time and cycle-time compression [22]; Northrop Grumman and the automotive and medical imaging cases demonstrate the quality vector through weight,

defect, and time-to-market reductions [2][23]; and Hutchinson's audit outcomes demonstrate the risk-reduction vector through avoided compliance exposure [22]. None of these results required a Tier 4 agentic deployment to materialize -- most were achieved at Tier 2 or Tier 3 capability, reinforcing that architecture discipline, not agent sophistication, drives the return.

V. Discussion: A Phased, Self-Funding Adoption Pathway

1.9. CLIMB2-OLIMP and the Discipline of Sequencing

The most frequent cause of failure I see in AI-driven digital transformation is what I call capability leapfrogging: an organization with fragmented, manual processes attempts to implement Tier 4 agentic workflows without first building a disciplined data foundation [24]. The CLIMB2-OLIMP dual-maturity model, developed through Design Science Research, is the clearest antidote I have used [24]. CLIMB2 diagnoses the consistency and structure of an organization's existing, non-AI new product development process; an enterprise needs a stable, documented process baseline before AI automation is introduced [24]. OLIMP then evaluates technical readiness once process stability is verified -- cloud-edge data infrastructure, security protocols, and human skills -- and uses large language models to generate budget-anchored, KPI-linked improvement plans [24]. Linking NPD process maturity directly to GenAI capability scaling keeps organizations from automating inefficient workflows, which otherwise just accelerates the generation of low-quality data faster [25].

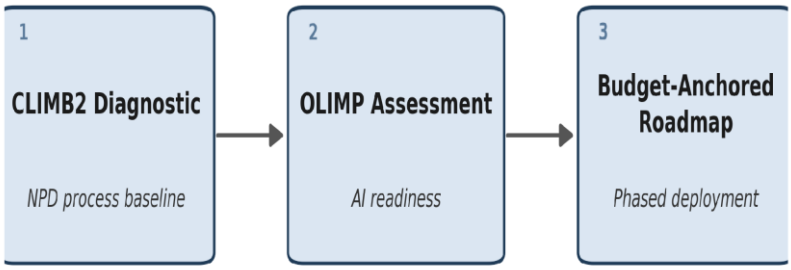

Figure 2. The CLIMB2-OLIMP sequencing model: diagnose process baseline, assess AI readiness, then deploy a budget-anchored roadmap.

1.10. The Three-Phase Investment Roadmap

To manage capital exposure and limit operational disruption, I structure AI-PLM transformation across a three-year phased roadmap [6], summarized in Table III.

Table III. Three-phase AI-PLM investment roadmap with budget allocation rules.

Implementation Phase	Focus & Scope	Core AI Level	Key Deliverables & Targets	Budget Allocation & Financial Rules
Phase 1: Foundation & Quick Wins (Months 1-6)	The starting point is unglamorous: close off the worst data silos and clean whatever metadata carries the most operational weight.	Tier 1 (AI-Wrapped) & Tier 2 (AI-Assisted).	Ticket classification, conversational document search, and metadata cleaning.	EUR 50,000-150,000; fund via operating expenses to bypass capital budget cycles.
Phase 2: Scaled Deployment (Months 6-18)	Once the foundation holds, automation extends outward across both product and service teams.	Tier 2 (AI-Assisted) & Tier 3 (Semantic AI Core).	Parts rationalization, duplicate detection, and automated compliance checks.	EUR 150,000-400,000; leverage audited Phase 1 savings to fund Phase 2 development.
Phase 3: Transformation (Months 18-36)	By the final phase, fully autonomous agent networks take over the work outright.	Tier 3 (Semantic Core) & Tier 4 (Agentic Workflow).	Graph-based change management, automated ECOs, and design optimization.	EUR 200,000+; long-term strategic funding linked to compounding yield and productivity gains.

Brownfield factories carry an additional cost layer that this phasing must absorb: infrastructure upgrades typically represent 30% to 50% of the initial investment [6]. Sensor deployment runs EUR 500 to EUR 5,000 per legacy machine [6]; edge gateway installation, EUR 5,000 to EUR 15,000 per production area [6]; industrial network upgrades, EUR 20,000 to EUR 100,000 per site [6]; and automated data pipelines to normalize proprietary equipment formats, EUR 30,000 to EUR 80,000 [6]. Because these upgrades build a shared foundation, the marginal cost of every subsequent AI use case drops 50% to 70% [6]. Initial infrastructure spend can push Year 1 payback out six to twelve months, but scaling multiple use cases across that shared data core improves long-term cumulative ROI substantially -- BCG finds that manufacturers scaling AI across five or more connected use cases achieve 3.2 times higher cumulative ROI than single-use-case deployers [6].

VI. Conclusion: Boardroom Recommendations

Measuring AI's ROI in legacy PLM environments

means moving past technology hype toward a structured, architecture-first approach to modernization [3]. Four operational guidelines have held up consistently across the engagements behind this framework.

1.11. Standardize the Process Baseline before Scaling

AI cannot clean an unstructured legacy database or repair a broken engineering workflow on its own [8]. Diagnostics such as CLIMB2-OLIMP should establish baseline process discipline before any AI scaling investment is made [24]; deploying automated agents onto an uncoordinated workflow or schema only generates inconsistency and technical debt faster [25].

1.12. Build Multi-Vector Business Cases

Business cases cannot rest on unverified productivity percentages [3]. They need clear calculations that map specific AI use cases to verified operational losses, structure returns across efficiency, quality, and risk-reduction vectors, and build fully loaded TCO models that price in change management, data engineering, and compliance overhead [6].

1.13. Address Security and Lock-In through Hybrid Architectures

Containerized microservices and open integration standards address data security and IP exposure directly [2]: sensitive engineering data stays on-premise while cloud resources handle non-sensitive, compute-intensive tasks. Open ontologies and modern integration protocols such as MCP let organizations bridge legacy silos without hard-coding point-to-point APIs, which cuts vendor lock-in risk [14].

1.14. Implement a Phased, Self-Funding Roadmap

Transformation should run as a phased, three-year roadmap engineered to deliver early, verifiable results [6]. Phase 1 targets low-complexity, high-impact use cases -- automated data classification, conversational search -- funded from operating budgets [6]. Documenting and auditing those early savings creates a self-funding cycle in which realized returns finance the more complex semantic data core migrations and agentic workflows of later phases [6].

The organizations that will win the AI-in-PLM decade are not the ones that adopt the most advanced agentic tooling first. They are the ones that can put a number in front of a CFO, defend it under audit, and prove it again a year later.

References

[1] AI in Product Lifecycle Management Market Size to Hit USD 75.72 Billion by 2035, Precedence Research, https://www.precedenceresearch.com/ai-in-product-lifecycle-management-market

[2] Transforming Aerospace and Defense with Next-Generation PLM, HCLTech, https://www.hcltech.com/sites/default/files/documents/resources/whitepaper/files/2026/03/20/transforming-aerospace-defense-with-next-generation-plm.pdf

[3] The State of AI in Product Development: What CIMdata's AI in PLM Global Study Reveals About Adoption, Investment, and Readiness, CIMdata, https://www.cimdata.com/en/events/event/912-the-state-of-ai-in-product-development-what-cimdata-s-ai-in-plm-global-study-reveals-about-adoption-investment-and-readiness

[4] How generative AI drives innovation and ROI in manufacturing, Glean, https://www.glean.com/perspectives/how-generative-ai-drives-innovation-and-roi-in-manufacturing

[5] Benchmarking AI in PLM, CIMdata, https://www.cimdata.com/en/education/educational-webinars/webinar-benchmarking-ai-in-plm

[6] AI ROI in Manufacturing 2026 Guide, The Thinking Company, https://thinking.inc/en/industry-service/manufacturing-ai-roi/

[7] How to Measure ROI in AI Engineering: A Practical Framework for Enterprise Leaders, HTEC, https://htec.com/insights/blogs/how-to-measure-roi-in-ai-engineering/

[8] Demystifying the "AI Native PLM" Buzzword, Beyond PLM Blog, https://beyondplm.com/2025/11/28/demystifying-the-ai-native-plm-buzzword/

[9] GCP-Native Teamcenter PLM Transformation, HCLTech, https://www.hcltech.com/case-study/gcp-native-teamcenter-plm-connected-device-innovation

[10] Windchill AI Assistant vs. Leo AI: What Engineers Actually Need from PLM Search in 2026, https://www.getleo.ai/blog/windchill-ai-assistant-vs-leo-ai-plm-search-comparison-2026

[11] PTC Launches New Windchill AI Parts Rationalization Capabilities, PTC Investor Relations, https://investor.ptc.com/resources/news/news-details/2026/PTC-Launches-New-Windchill-AI-

Parts-Rationalization-Capabilities/default.aspx

[12] Windchill AI Parts Rationalization, PTC Community, https://community.ptc.com/windchill-10/windchill-ai-parts-rationalization-171843

[13] The Five AI Implementation Patterns in PLM: A Framework for Informed Decision-Making, CIMdata, https://www.cimdata.com/en/education/educational-webinars/the-five-ai-implementation-patterns-in-plm-a-framework-for-informed-decision-making

[14] PTC's AI Vision for Fueling Innovation Across the Product Lifecycle, PTC, https://www.ptc.com/en/blogs/plm/ptcs-ai-vision-for-fueling-innovation-across-the-product-lifecycle

[15] Total cost of ownership for enterprise AI: Hidden costs | ROI factors, Xenoss, https://xenoss.io/blog/total-cost-of-ownership-for-enterprise-ai

[16] AI in PLM | Innovation in Product Lifecycle Management, Trace One, https://www.traceone.com/ai-plm

[17] AI Software Development Costs 2026: Enterprise Spending, TCO, and ROI Analysis, Keyhole Software, https://keyholesoftware.com/ai-software-development-cost-2026/

[18] Total Cost of Ownership for Agentic AI, Corvair, https://corvair.ai/insights-tco-agentic-ai.html

[19] Agentic AI Total Cost of Ownership: Enterprise Guide, Trantor, https://www.trantorinc.com/blog/agentic-ai-total-cost-of-ownership

[20] Enterprise AI Development Cost: 2026 Pricing Guide, TechAhead, https://www.techaheadcorp.com/blog/enterprise-ai-development-cost/

[21] Understanding ROI: The Business Case for PLM, Centric Software, https://go.centricsoftware.com/whitepapers/understanding-roi-the-business-case-for-plm

[22] PLM ROI: how to measure the benefits of a PLM investment?, Aletiq, https://www.aletiq.com/en/post/calculate-plm-roi

[23] Digital PLM & Digital Thread, Hitachi Digital Services, https://www.hitachids.com/capability/digital-plm-digital-thread/

[24] A prescriptive generative AI maturity model for new product development processes, Emerald (Journal of Manufacturing Technology Management), https://www.emerald.com/jmtm/article/37/9/40/1346279/A-prescriptive-generative-AI-maturity-model-for

[25] Data preparation for AI: How to build a trustworthy foundation for any system, N-iX, https://www.n-ix.com/data-preparation-for-ai/