Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ **Research Article**

AI-Driven Quantitative Models for Improving Bill-of-Materials Accuracy in Medical Device Product Lifecycle Management

Vivek Agrawal

Independent Researcher

ARTICLE INFO **ABSTRACT**

Medical device manufacturers continue to lose engineering hours to a bill-of-materials (BOM) that behaves as a static ledger rather than a governed source of truth, and the consequence is measurable: 87% of engineering leaders in one industry-wide survey report needing hours or days to retrieve the data required to justify a single design decision [1]. This article argues that the fix is not another integration layer but a change in what the BOM mathematically is — from a tabular record to a directed graph that machine learning models can traverse, compare, and reconcile. Drawing on graph-theoretic representations, phylogenetic tree reconciliation, spectral and hierarchical graph neural networks, Temporal Production Graphs, and the Orthogonal Procrustes alignment framework, the analysis works through the mathematical formulations that let a PLM platform detect labeling mismatches, topological discrepancies, and combinatorial drift between an engineering BOM and its manufacturing counterpart before those errors reach the production floor. The discussion extends the model to unsupervised anomaly detection, intelligent document processing for legacy drawing sheets, and the semantic infrastructure needed to synchronize CAD, PLM, ERP, and MES systems along a seven-level maturity spectrum. Because medical devices carry regulatory exposure that most manufacturing sectors do not, the article also positions these quantitative models against the FDA's shift from Computer Software Validation to Computer Software Assurance, the GAMP 5 Appendix D11 lifecycle for machine-learning validation, the Predetermined Change Control Plan framework, and the EU's draft Annex 22 restrictions on adaptive algorithms in GxP manufacturing. The contribution is a unified quantitative architecture — graph representation, learned reconciliation, and regulatory-aware validation — that reframes BOM accuracy as a computable, auditable property of the digital thread rather than a downstream quality metric.

Keywords: Bill of Materials, Graph Neural Networks, Digital Thread, Medical Device PLM, Orthogonal Procrustes, Computer Software Assurance, GAMP 5

1. INTRODUCTION

A medical device's engineering record ends, structurally, at its Bill of Materials (BOM) — the nested inventory of every physical component, mechanical part, electronic assembly, and software dependency needed to build the system [1]. Inside a Product Lifecycle Management (PLM) architecture, that BOM functions less like a parts list and more like the product's genetic code: it coordinates mechanical design, procurement, quality assurance, manufacturing execution, and regulatory compliance from a single structural record [2]. Treating it as anything less invites the exact failure modes this article sets out to solve mathematically.

In regulated device manufacturing, a BOM discrepancy is rarely a paperwork problem. It escalates into late-stage engineering change orders, delayed clinical trials, production halts, and — in the worst case — product recalls [3]. What is striking is not that these failures occur, but that they persist despite years of PLM investment: fragmented data environments, lagging change synchronization, and manual data entry keep reintroducing the same class of error into systems that were supposed to have engineered it away [3].The scale of the retrieval problem is documented rather than anecdotal. An industry-wide survey of engineering leaders found that 87% require hours or even days to

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ **Research Article**

retrieve the data needed to justify a single design decision [1]. That bottleneck alone extends design latency and raises the probability of human error during manual BOM compilation [3]. The pattern points to a structural cause: most PLM platforms still operate as systems of record — they version files and route approvals — without the structural reasoning needed to catch qualitative or logical mismatches between drawings, schematics, and the physical bill of materials [3].

This article's position is that closing that gap requires re-expressing the BOM itself in a form machine learning can operate on natively — a graph rather than a table — and layering AI-driven quantitative models, semantic infrastructure, and regulatory-aware validation on top of that representation [2]. Sections 2 and 3 develop the mathematical architecture: graph representations of multi-level hierarchies, phylogenetic reconciliation, spectral and hierarchical graph neural networks, Temporal Production Graphs, and the Orthogonal Procrustes alignment method. Section 4 extends the model into anomaly detection, document intelligence, and semantic synchronization across the enterprise stack, and situates the resulting capability inside the regulatory frameworks that govern AI in GxP manufacturing. Section 5 closes with the operational and strategic implications for device manufacturers building this capability into their digital thread.

2. RELATED WORK AND BACKGROUND

2.1 From eBOM to mBOM: Where Structural Divergence Originates

The transition from an Engineering Bill of Materials (eBOM) to a Manufacturing Bill of Materials (mBOM) is where most of this divergence originates [4]. Design engineers organize the eBOM around spatial sub-assemblies defined in 3D CAD, while manufacturing engineers restructure the mBOM around physical assembly steps, workstations, and production sequences [4]. The mBOM also carries manufacturing consumables — adhesives, packaging, sterilization agents — that rarely appear in the 3D CAD model at all, which means the two structures are never fully isomorphic by design [5].

Platforms such as Teamcenter Manufacturing Easy Plan attempt to close this gap by analyzing historical transformation patterns, extracting assembly data from design revisions, proposing Bill of Process sequences, and flagging structural discrepancies automatically [6]. Scoped accountability checks layered on top let manufacturing engineers concentrate validation effort on the assemblies that actually changed, which accelerates release cycles and keeps outdated configurations off the production floor [6]. These capabilities are useful, but they remain heuristic pattern-matching over historical transformations rather than a formal structural comparison — which is precisely the gap the graph-theoretic methods in Section 3 are built to close.

2.2 Traceability, Provenance, and the Widening BOM Definition

Modern medical devices fuse hardware, embedded software, and increasingly, adaptive AI algorithms [2]. That fusion has pushed the definition of "BOM" outward: product safety now depends on maintaining a Software Bill of Materials (SBOM) alongside an AI Bill of Materials (AI-BOM) — a structured, machine-readable catalog of every dataset, model architecture, training configuration, and software dependency used to build and operate the AI system [7]. An AI-BOM built to schemas such as SPDX 3.0 or CycloneDX documents model details, dataset lineage, performance profiles, and cryptographic verification in one traceable structure.

```
AI-BOM Core Architecture (SPDX 3.0 / CycloneDX)
 |-- Model Details [Architecture, Versioning, Parameters]
 |-- Dataset Lineage [Origins, Annotations, Preprocessing]
 |-- Performance Profiles [Sensitivity, Specificity, F1 Metrics]
 \-- Verification [Cryptographic Hash, Digital Signatures]
```

This architecture matters operationally because it hardens the device against three specific vulnerability classes: data poisoning that corrupts training sets to manipulate predictions, adversarial manipulation of input signals designed to fool the network, and dependency vulnerabilities in outdated third-party software components [7]. Validating this metadata against schemas such as SPDX 3.0's AI and Dataset profiles establishes traceable model provenance for security teams [8], and folding these assets into continuous risk engines — C2A Security's EVSec platform is one industry example — enables ongoing vulnerability scanning across the full device lifecycle [9]. When a vulnerability

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ **Research Article**

surfaces, a properly linked digital thread lets the organization locate every affected physical device configuration immediately rather than searching for it [2].

2.3 Semantic Infrastructure as the Precondition for AI-Driven BOM Accuracy

None of the graph-based or learned reconciliation methods in Section 3 function without a unified semantic layer underneath them. End-to-end integration across engineering, manufacturing, and supply chain operations depends on a standardized vocabulary that translates data definitions across CAD tools, PLM platforms, ERP systems, and MES environments [10]. Establishing a common ontology gives parts, BOM structures, and change records platform-independent definitions [10]; without it, integration pathways stay brittle, held together by custom scripts that break at every system upgrade [10].

Organizations that build this semantic core out can be placed along a maturity spectrum that runs from disconnected spreadsheets to fully closed-loop, self-optimizing digital threads. Table 1 lays out that spectrum together with the role AI and semantic modeling play at each level.

PLM Maturity Level	Operational Characteristics	Role of AI and Semantic Modeling
Level 0: Foundations	Manual processes; data lives in disconnected spreadsheets and local files [10].	None — data is fragmented and prone to manual error [10].
Level 1: Process Digitization	Simple workflow automation; files are vault-managed with basic change-routing [10].	Automation limited to process routing; integrations remain point-to-point [10].
Level 2: System Integration	Core platforms (PLM, CAD, ERP, MES) connected; eBOM-to-mBOM changes tracked [10].	Controlled metadata exchange establishes a single source of truth [10].
Level 3: Data Intelligence	Active pattern identification; automated part classification and anomaly detection [10].	ML models analyze change histories to classify components and detect anomalies [3].
Level 4: Cognitive Change	Predictive impact analysis; change processes run automatically from upstream risk signals [10].	AI triggers change workflows from risk alerts, field failures, or regulatory shifts [3].
Level 5: Digital Thread	Traceability stretches from early requirements to active field deployment [10].	Semantic linkages connect design requirements, BOM elements, and field configurations [10].
Level 6: Agentic Autonomy	Automated workflow orchestration; independent AI agents coordinate changes across platforms [11].	Multi-agent systems coordinate across PLM, ERP, and MES to resolve design changes [11].
Level 7: Closed-Loop Autonomy	Self-optimizing lifecycles; digital twins continuously align design and manufacturing states [10].	Continuous learning loops reconcile field telemetry with future product designs [10].

Table 1. PLM maturity spectrum and the corresponding role of AI/semantic modeling at each level (adapted from industry sources [10], [11]).

3. METHODOLOGY: MATHEMATICAL FORMULATIONS FOR BOM REPRESENTATION AND ALIGNMENT

Applying machine learning to a hierarchical assembly system first requires translating the tabular BOM into a mathematical object that carries both component attributes and assembly relationships [12]. The formulations below build that object in stages: a graph representation, a reconciliation method borrowed from an unrelated science, a

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ **Research Article**

family of neural architectures that operate on the graph directly, a temporal extension for supply dynamics, and a closed-form alignment method for reconciling two divergent structures.

3.1 Graph Representation and Phylogenetic Tree Reconciliation

A multi-level BOM is mathematically a directed acyclic graph (DAG), or equivalently a hierarchical tree, G = (V, E) [1]. Nodes v in V represent components, raw materials, or sub-assemblies; edges e in E encode assembly dependencies and quantitative linkages [1]. Every node carries a high-dimensional attribute vector — taxonomic metadata, physical measurements, electrical properties, supplier credentials — so the graph is attributed as well as structured [13].

Comparing structural variation across design cycles or product options is where this article's central methodological borrowing happens: phylogenetic tree reconciliation, developed to reconstruct evolutionary histories and compare genetic sequences, maps directly onto the problem of comparing a query BOM against a validated reference design [1]. The topology of the query is aligned against the reference topology, and the mapping surfaces three distinct classes of structural deviation.

Labeling mismatches occur where a component's classification differs from its parent assembly's structural parameters — a point-by-point attribute shift rather than a structural one [14]. Topological discrepancies are changes in parent-child relationships, such as a component reassigned to a different level of the physical assembly tree [14]. Combinatorial shifts are the hardest to catch by inspection: both the underlying parts and the linking structure diverge simultaneously across configurations [14]. Formulating the reconciliation as an optimization problem that minimizes tree edit distance turns this into a solvable comparison rather than a manual audit, giving design engineers a visual signal for unauthorized structural variation before a design is released to manufacturing [1].

3.2 Spectral and Hierarchical Graph Neural Networks

Flat neural networks assume grid-structured input; a BOM graph is irregular by nature, which is why spectral Graph Neural Networks (GNNs) — computing convolutions in the Fourier domain — are the relevant architecture rather than a convolutional network adapted after the fact [13]. Hierarchical Graph Neural Networks (HGNNs) extend this further with multi-scale graph pooling — top-k selection, edge contraction, spectral decimation — to produce multi-resolution abstractions of the device structure at part, module, and system granularity simultaneously [15].

Vertical and horizontal message-passing lets scale-specific GNN layers propagate component representations across variable depths of the hierarchy [15], which is what allows the same trained network to assess integrity at the part level and the system level without separate models for each. Figure 1 sketches the pooling and unpooling relationship between a coarsened module level and its constituent raw components.

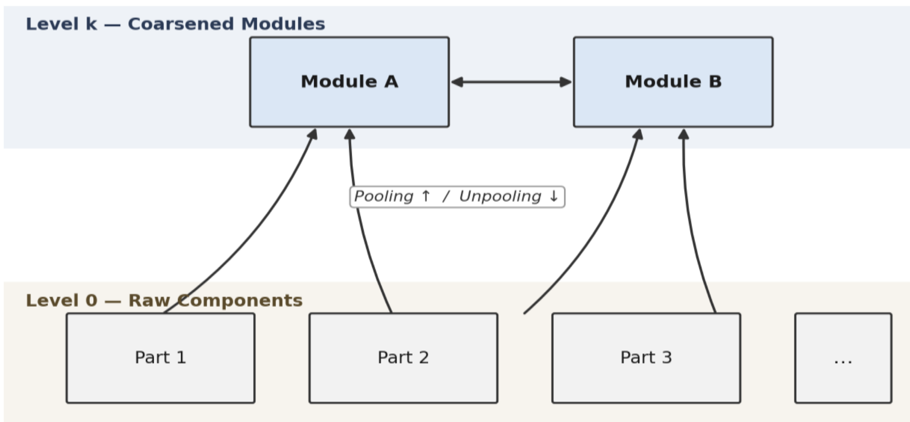

Figure 1. Hierarchical pooling between raw-component and coarsened-module graph levels [15].

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ **Research Article**

Supply networks add a dimension this static graph cannot capture on its own: time. Modeled dynamically, they become Temporal Production Graphs (TPGs), which — unlike static representations — capture time-varying transaction flows and dependencies where edges are governed by latent production functions [16]. Combining temporal GNNs with a dedicated inventory module lets the system trace how an external component shortage or a design change propagates through the supply chain over time [16]. The inventory state of a firm i at time step t is a function of the time-varying transactions received from suppliers and the deliveries shipped to downstream buyers, with attention weights mapping input supply configurations to output consumption rates [16] — which is what lets the model forecast which device assemblies will be affected by component obsolescence or supply disruption before the shortage actually hits the line.

3.3 Algebraic Harmonization via Orthogonal Procrustes Alignment

Reconciling an engineering design configuration with the actual manufacturing setup is fundamentally a spatial alignment problem, and the Orthogonal Procrustes framework gives it a closed-form solution [1]. Let X and Y be two coordinate matrices of dimension n x d, derived by embedding the structural features and node properties of the eBOM and mBOM respectively [1]. Resolving the dimensional and spatial misalignment between them means rotating and reflecting X under orthogonal constraints until it matches Y as closely as possible [1], formulated as the minimization:

min over orthogonal R of the Frobenius norm ||XR - Y||_F , subject to R^T R = I

where R is the orthogonal transformation matrix, I is the identity matrix, and the Frobenius norm measures the residual distance between the rotated eBOM embedding and the mBOM embedding [1]. This has a closed-form solution: apply Singular Value Decomposition (SVD) to the cross-correlation matrix X^T Y = U Sigma V^T, giving R = U V^T [17].

The resulting residual distance is a deterministic anomaly signal, not a heuristic one [1]. High residual values flag structural anomalies — mismatched subassemblies, omitted physical components — and because the metric is continuous rather than binary, quality control teams can rank and target their review effort at the sections of the assembly tree the metric identifies as anomalous, rather than auditing the full tree uniformly [1].

3.4 Unsupervised Detection Without Labeled Ground Truth

Labeled BOM-error datasets are scarce in practice, which pushes deep-tier discrepancy detection toward architectures that combine clustering with generative modeling rather than supervised classification [1]. Isolation Forest algorithms partition component attribute matrices to surface outliers — parts with mismatched physical descriptions, incorrect supplier designations, anomalous material properties [18]. For sequential or versioned configurations, Variational Autoencoders (VAEs) compress the graph structure into a lower-dimensional latent space and attempt to reconstruct the original matrix from that compressed form [18]; components with high reconstruction loss are the ones the model is effectively unable to explain, which is itself the anomaly signal [19].

When the same reasoning is applied to telemetry and transaction sequences during microservice rollouts or system releases, detection performance is governed by three hyperparameters [19]. Window duration W sets the length of the monitored interval — larger windows smooth transient variation but delay detection of genuine faults [19]. Window stride S sets how far the window advances between computation steps; setting S less than W creates overlapping intervals, trading higher computational cost for finer detection granularity [19]. Detection threshold K is the number of consecutive windows that must flag anomalous behavior before a system-wide alert fires — a higher K suppresses false positives at the cost of slower response to intermittent errors [19]. Functionally, the model behaves like a physics-based spelling checker: it scans physical and logical relationships and flags cases such as a component's electrical rating falling outside safety limits or two parts scheduled for assembly that cannot physically coexist in the same production envelope [5].

3.5 Intelligent Document Processing for Unstructured Legacy Records

A large share of BOM inaccuracy traces back to data that was never structured in the first place — legacy PDF drawing sheets, 2D drawings, supplier equipment datasheets [2]. Intelligent Document Processing (IDP) systems address this

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ Research Article

with high-resolution Optical Character Recognition paired with transformer-based Natural Language Processing [2], following the pipeline in Figure 2.

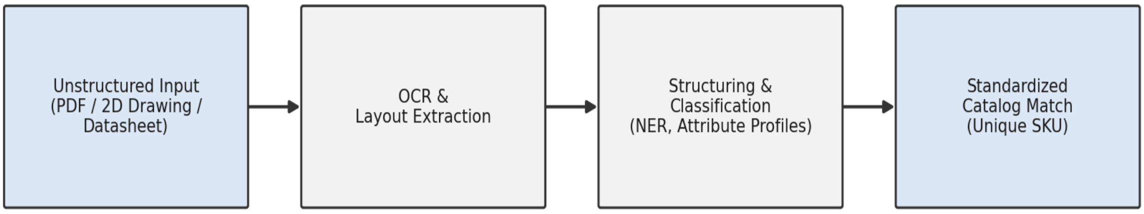

Figure 2. Intelligent Document Processing pipeline from unstructured schematic to standardized catalog match [2].

Once a document is parsed, models classify each page by visual layout and text density, and NLP algorithms decompose technical description strings into standardized attributes that are then mapped against the master catalog to assign SKUs, flag duplicates, and detect discrepancies between schematic notes and downstream PLM listings [5]. This closes a specific gap between manual document review and dynamic design systems [5] — and adoption data suggests the gap is closing quickly: only 28% of manufacturing organizations currently use AI-powered 2D drawing review tools, but 52% of engineering leaders plan to acquire the capability within six months and another 20% within two years [1]. That trajectory reflects an industry-wide push to eliminate manual blueprint validation, which is itself a major source of downstream engineering change orders [3].

3.6 Context Engineering for Autonomous BOM Agents

Deploying AI workflows across enterprise engineering databases exposes a limitation in traditional prompt engineering, which optimizes the user's query rather than the data the query runs against [20]. Context engineering inverts that priority: it structures the underlying product data — engineering metadata, change records, supplier specifications — so that an AI agent can act on it safely [20]. Fast inference without structured context simply produces fast mistakes at scale [20]. A robust context fabric exposed through standardized APIs is what lets organizations deploy autonomous agents that execute sequential tasks — launching stress simulations, validating drawing annotations, submitting change reviews — while keeping their recommendations tied to the actual product configuration rather than a stale snapshot of it [11].

4. RESULTS AND DISCUSSION: REGULATORY VALIDATION AND INDUSTRY IMPLEMENTATION

4.1 From Computer Software Validation to Computer Software Assurance

Any software used in medical device design, production, or quality management must be validated for its intended use [21]. The historical approach — Computer Software Validation (CSV) — leaned on rigid, document-heavy test scripts that generated large volumes of paperwork while doing comparatively little to prevent actual system defects [22]. The FDA's Quality Management System Regulation, which incorporates ISO 13485:2016 by reference, moves the paradigm toward Computer Software Assurance (CSA): critical thinking and risk-based testing replace scripted exhaustiveness, with automated testing tools doing the heavy lifting [22].

Validation effort under CSA scales with the software's actual impact on product quality and patient safety [21]. Systems supporting safety-critical tasks — verifying essential process parameters, making automated quality decisions — still require rigorous testing [22]; non-critical database tools, by contrast, can rely on exploratory testing and supplier verification documentation, which meaningfully reduces validation overhead without loosening scrutiny where it matters [23]. Figure 3 sketches this risk-scaled validation pathway.

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ **Research Article**

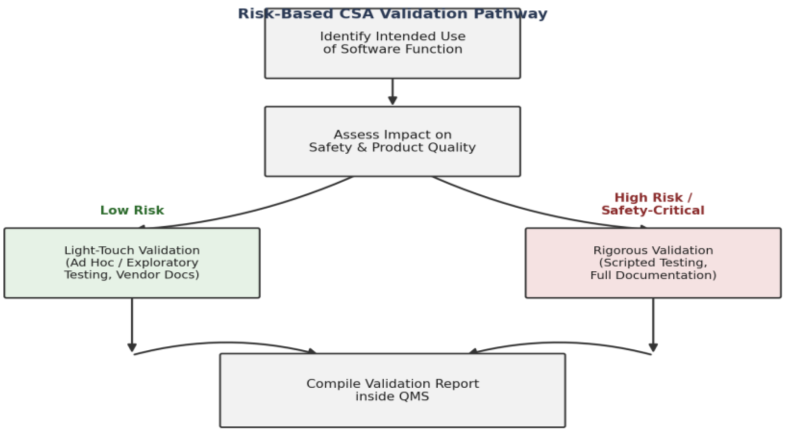

Figure 3. Risk-scaled Computer Software Assurance validation pathway [22], [24].

4.2 GAMP 5 Appendix D11 and the Machine-Learning Validation Lifecycle

Machine-learning systems operating in regulated (GxP) environments need validation that extends beyond static verification into the lifecycle of a data-driven model that keeps learning after deployment [24]. The ISPE GAMP 5 Second Edition, specifically Appendix D11, is the framework that defines that lifecycle for AI and ML systems [24], and it requires the validation protocol to address data integrity across three functionally distinct data categories [25]: training data, which defines model weights and algorithmic relationships; tuning data, sequestered profiles used to optimize parameters and refine architectures; and testing data, an independent, sequestered dataset — collected from multiple separate clinics or manufacturing sites — used to calculate final performance metrics such as sensitivity and specificity, and which must never influence training [26].

Executing validation across the AI lifecycle means running distinct activities at each phase, with different teams responsible for each. Table 2 summarizes this validation structure.

Validation Phase	Team Responsible	Purpose and Execution	Key Operational Considerations
Data Validation	Data Scientists & Process Owners	Ensure quality, completeness, and accuracy of training and testing datasets [27].	Verify ground-truth annotations, enforce ALCOA+ integrity controls, check for bias [26].
Model Validation	System Developers & Infrastructure Providers	Evaluate the mathematical accuracy of the trained model [27].	Run evaluations across sequestered data profiles to assess predictive stability [26].
Algorithm Validation	R&D Specialists	Confirm the algorithm's mathematical operations perform consistently [27].	Verify mathematical functions and algorithmic logic [27].
API Validation	Software & Systems Engineers	Secure integration pathways connecting ML systems to PLM databases [4].	Implement input-output filtering, data anonymization, and security testing [27].

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ **Research Article**

Infrastructure/Application Validation	Quality Assurance & Validation Teams	Validate underlying hardware, physical databases, and user interfaces [27].	Apply GAMP Category 4/5 protocols (IQ/OQ/PQ) to verify deployment [23].
Performance Validation	Clinical Experts & Quality Managers	Assess operational performance in real-world clinical environments [27].	Track performance KPIs (sensitivity, false-positive rate) to identify drift [26].

Table 2. GAMP 5 Appendix D11 validation phases across the AI/ML lifecycle [27], [26].

4.3 Marketing Submissions, Predetermined Change Control Plans, and Cross-Border Constraints

Software functions embedded in clinical devices force regulatory approval paradigms to adapt to algorithms that continue to learn and evolve after clearance [21]. In January 2025, the FDA issued draft guidance on AI-Enabled Device Software Functions covering lifecycle management and marketing submission recommendations [28], integrating with the Predetermined Change Control Plan (PCCP) guidelines finalized in late 2024 [21]. A PCCP describes planned algorithm modifications, validation methods, and post-market tracking protocols in advance [21]; once authorized within an initial marketing submission — a 510(k) or a PMA — the manufacturer can implement pre-approved algorithm updates without filing again for each one [21]. Figure 4 traces this execution pathway.

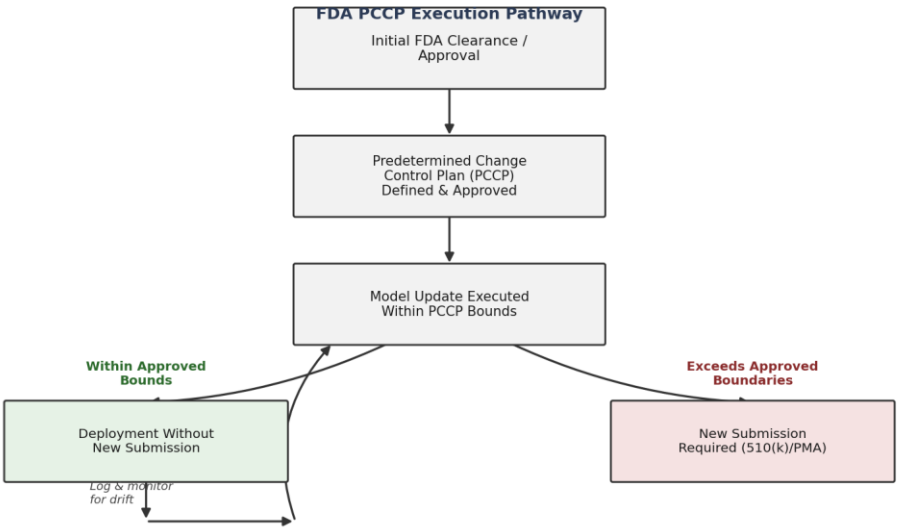

Figure 4. FDA Predetermined Change Control Plan execution pathway [21].

International regulators are drawing their own boundaries in parallel. The EU's draft GMP EudraLex Annex 22 sets strict limits on AI use in pharmaceutical and medical device production [26]: only static, deterministic models are permitted in critical manufacturing steps, meaning identical inputs must yield identical outputs and dynamic systems that keep learning in production are prohibited outright [26]. Generative AI and large language models are excluded from critical GxP decisions entirely, and where they appear in non-critical tasks, human-in-the-loop review is mandatory [26]. Underlying both restrictions is a performance baseline requirement: the AI application must perform at least as well as the traditional or manual process it replaces [26]. Taken together, these constraints mean the graph-based reconciliation methods in Section 3 are best positioned as decision-support and anomaly-flagging tools that keep a human validator in the approval loop, not as autonomous replacements for that validator — a distinction that should inform how manufacturers scope any deployment of this architecture in a regulated environment.

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/ Research Article

4.4 Industry Implementations and Operational Outcomes

Three current platform implementations illustrate how these mathematical and regulatory principles translate into deployed capability. Makersite operates as a decision-intelligence layer adjacent to enterprise PLM and ERP systems, connecting internal BOM data with more than 150 verified environmental, regulatory, and financial databases to automate lifecycle assessments, environmental declarations, and compliance tracking [3]. Its AI engine uses industry-trained agents for background database matching and gap-filling [3], and when primary supplier data is missing, it infers material properties, likely manufacturing steps, and regional emission factors from the product attributes that are available [3] — which is what allows product teams to simulate design changes and run cost evaluations in real time [29].

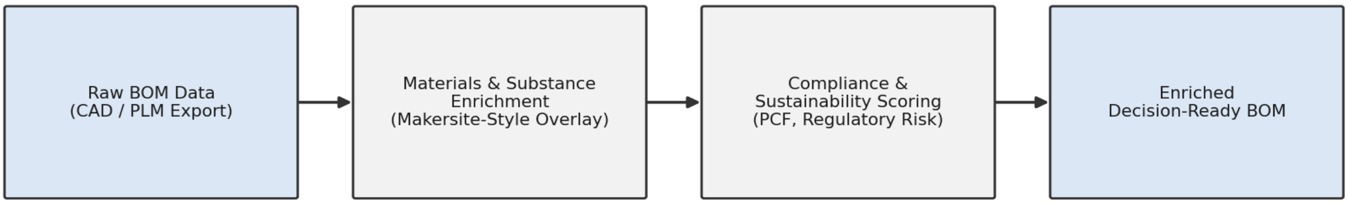

Figure 5. Makersite data enrichment pipeline for BOM-adjacent decision intelligence [3], [29].

PTC Windchill takes a narrower but operationally significant approach, embedding machine learning-based classification and parts rationalization directly into the PLM platform to scan enterprise databases for duplicate parts and redundant files [11]. By evaluating 3D geometry alongside metadata attributes, the system can suggest existing parts to engineers while they are actively designing rather than after the fact [11], and when duplicates surface, change workflows consolidate inventory and clean up catalog data automatically [11] — lowering inventory carrying costs and procurement complexity while accelerating development [11].

Siemens Teamcenter Easy Plan 2512 addresses the eBOM-to-mBOM reconciliation problem directly through a conversational AI assistant that lets engineers generate Manufacturing Bills of Materials and Bills of Process using natural language [6], automatically extracting legacy process data to standardize production routing steps [6]. Teamcenter's Smart Discovery analytics add scoped accountability checks on top, letting planning teams target validation effort at modified subassemblies specifically so that engineering changes propagate correctly to the manufacturing floor [4].

Read together, these three implementations validate the article's central claim from a different angle than the mathematics in Section 3: the market is already moving toward graph-aware, context-sensitive BOM intelligence, even where vendors have not formalized the underlying representation as a DAG or applied Orthogonal Procrustes alignment explicitly. The mathematical architecture developed here is less a departure from current practice than a formalization of where that practice is already heading — with the added benefit of an auditable, regulator-legible basis for the anomaly flags these systems generate.

5. CONCLUSION

Embedding artificial intelligence and quantitative graph models into medical device PLM is not a peripheral tooling upgrade; it marks the point where the digital thread starts to become self-checking. Representing hierarchical product structures as dynamic graphs, and applying closed-form alignment methods such as Orthogonal Procrustes, turns the Bill of Materials from a static record into an active source of decision intelligence that manufacturing, quality, and regulatory teams can all query against the same structural ground truth.

These models automate data cleansing, anticipate lifecycle anomalies before they reach production, and streamline eBOM-to-mBOM reconciliation — but none of that capability is deployable in a medical device context without validation workflows that satisfy GAMP 5 Appendix D11, the EU's draft GMP Annex 22, and the FDA's Predetermined

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Change Control Plan framework simultaneously. Regulatory alignment is not a constraint bolted onto the technical architecture after the fact; it has to be designed into the same graph structure that carries the engineering data.

Manufacturers that build these quantitative models on validated, well-structured product context stand to gain on three fronts at once: shorter time-to-market, lower operational risk exposure, and a stronger evidentiary basis for the safety and quality claims that regulators and clinicians both depend on. The mathematics exists; the remaining work is organizational — building the semantic layer and validation discipline that let these models operate inside a regulated environment rather than alongside it.

REFERENCES:

[1] Product structure (BOM)-based product similarity measures using orthogonal procrustes approach | Request PDF - ResearchGate, https://www.researchgate.net/publication/220384899_Product_structure_BOM-based_product_similarity_measures_using_orthogonal_procrustes_approach
[2] The AI Revolution in BOM Management: Driving Smarter Manufacturing and Product Development - CQLsys Technologies, https://cqlsystechnologiesblog.wordpress.com/2025/10/15/the-ai-revolution-in-bom-management-driving-smarter-manufacturing-and-product-development/
[3] 9 AI-Powered PLM Software Solutions for Manufacturers in 2026 - Makersite GmbH, https://makersite.io/insights/9-ai-powered-plm-software-solutions-for-enterprise-manufacturers-in-2026/
[4] Types of BOM in Heavy Industries: Understanding EBOM, MBOM, & More - Markovate, https://markovate.com/types-of-bom/
[5] Automated Intelligent Document Processing Software for BOM - Verdantis, https://www.verdantis.com/intelligent-document-processing/
[6] What's New in Teamcenter Manufacturing Easy Plan 2512, https://blogs.sw.siemens.com/teamcenter-manufacturing/2026/01/28/whats-new-in-teamcenter-manufacturing-easy-plan-2512/
[7] AI Bill of Materials (AI-BOM) and Model Provenance: A New Approach to AI Supply Chain Security | TechIntelPro, https://techintelpro.com/articles/ai-bill-of-materials-ai-bom-and-model-provenance-a-new-approach-to-ai-supply-chain-security
[8] What Is an AI-BOM (AI Bill of Materials)? & How to Build It - Palo Alto Networks, https://www.paloaltonetworks.com/cyberpedia/what-is-an-ai-bom
[9] It's Time to Talk About AI/ML BOM (Artificial Intelligence Bill of Materials) And Vulnerability Management | C2A Security - The Only Risk-Driven DevSecOps Platform, https://c2a-sec.com/its-time-to-talk-about-ai-ml-bom-artificial-intelligence-bill-of-materials-and-vulnerability-management/
[10] Engineering Services | PLM Maturity with a Semantic Core - Infosys Blogs, https://blogs.infosys.com/engineering-services/5g-edge/plm-maturity-with-a-semantic-core.html
[11] AI for PLM in Engineering - CoLab Software, https://www.colabsoftware.com/post/ai-for-plm-in-engineering
[12] Leveraging Graph Data Science in BOMs for Optimized Decision-Making in Production and Supply Chain - OpenBOM, https://www.openbom.com/blog/leveraging-graph-data-science-in-boms-for-optimized-decision-making-in-production-and-supply-chain
[13] Graph neural networks for materials science and chemistry - PMC - NIH, https://pmc.ncbi.nlm.nih.gov/articles/PMC9702700/
[14] (PDF) A Data Mining Approach to Forming Generic Bills of Materials in Support of Variant Design Activities - ResearchGate, https://www.researchgate.net/publication/244953890_A_Data_Mining_Approach_to_Forming_Generic_Bills_of_Materials_in_Support_of_Variant_Design_Activities
[15] Hierarchical Graph Neural Networks (HGNN) - Emergent Mind, https://www.emergentmind.com/topics/hierarchical-graph-neural-network-gnn
[16] Learning production functions for supply chains with graph neural networks - arXiv, https://arxiv.org/html/2407.18772v1
[17] BOM in the AI Era, Context Engineering Matters in Modern PLM - OpenBOM, https://www.openbom.com/blog/plm-and-bom-management/bom-in-the-ai-era-context-engineering-plm
[18] How AI Improves Accuracy in Bill of Materials Processes - OutRight CRM, https://www.outrightcrm.com/blog/ai-in-bill-of-materials-process/

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/
Research Article

[19] Praxium: Diagnosing Cloud Anomalies with AI-based Telemetry and Dependency Analysis, https://arxiv.org/html/2603.23890v2
[20] Computer Software Assurance for production and quality system software - Kneat, https://kneat.com/articles/regulatory/understand-computer-software-assurance-production-quality-system-software/
[21] How the FDA Reviews AI and Machine Learning Medical Devices: Complete Guide, https://www.complizen.ai/post/fda-ai-machine-learning-medical-devices-review
[22] FDA's Computer Software Assurance 2026: Changes and What to Do Next, https://pscsoftware.com/resource-center/article/fda-computer-software-assurance-2026-changes/
[23] Computer Systems Validation: How To Avoid FDA Warning Letters CFR 820.70(i), https://www.greenlight.guru/blog/computer-systems-validation-how-to-avoid-fda-warning-letters
[24] GAMP 5 Second Edition Explained: Key Changes and Updates - PSC Software, https://pscsoftware.com/resource-center/article/gamp-5-second-edition-changing-validation/
[25] AI/ML Validation in GxP: A Guide to GAMP 5 Appendix D11 | IntuitionLabs, https://intuitionlabs.ai/articles/gamp-5-ai-validation-gxp
[26] AI/ML Validation in GxP: A Guide to GAMP 5 Appendix D11 - IntuitionLabs, https://intuitionlabs.ai/pdfs/ai-ml-validation-in-gxp-a-guide-to-gamp-5-appendix-d11.pdf
[27] FDA AI Action Plan Explained - Enlil, Inc., https://enlil.com/blog/fda-ai-action-plan-explained/
[28] FDA Releases Draft Guidance on Submission Recommendations for AI-Enabled Device Software Functions - King & Spalding, https://www.kslaw.com/news-and-insights/fda-releases-draft-guidance-on-submission-recommendations-for-ai-enabled-device-software-functions
[29] Artificial Intelligence in Software as a Medical Device - FDA, https://www.fda.gov/medical-devices/software-medical-device-samd/artificial-intelligence-software-medical-device