

Original Article

Architecting Quantum-Ready ERP Platforms: A Modernization Framework for Manufacturing

Sankalp Shrivastava

Enterprise Architect, USA.

Corresponding Author : shri.sankalp@ieee.org

Received: 17 July 2026

Revised: 26 August 2026

Accepted: 10 September 2026

Published: 28 September 2026

Abstract - ERP modernization is no longer only a version upgrade or a move to cloud infrastructure. In many programs, the harder architecture work is deciding what stays in the transaction core and what should sit outside it. Quantum tech adds an unusual twist to that decision. A few planning and optimization problems may become candidates for combined quantum-classical processing. At the same time, the security effect of quantum computing - especially migration away from vulnerable asymmetric cryptography - has to be considered much earlier. This paper proposes a vendor-neutral architecture for that situation. ERP continues to record and process business transactions, while optimization happens behind a workload broker layer that smoothly connects to classical, quantum-inspired, or quantum services—without altering the ERP posting process. This approach also supports crypto-agility, using flexible interfaces, certificates, signatures, and data flows. Manufacturing is chosen as the example because it deals with real-world problems encountered during scheduling, logistics, inventory, and planning. These are areas where better optimization can have a real impact. The approach is to improve integration and security now, and keep the option open to use quantum services in the future if they become useful.

Keywords - Crypto-Agility, Hybrid Computing, Post-Quantum Cryptography, Supply Chain Optimization, Digital Core.

1. Introduction

ERP modernization has one constraint that ordinary application modernization often does not: the business cannot simply stop while the architecture is being redesigned. Most financial transactions, production orders, inventory changes, procurement paperwork, sales transactions, and asset details need to stay consistent while everything is being updated. That is why most ERP programs evolve the landscape around a stable transaction core rather than rewrite everything at once. Cloud platforms, cleaner service distinctions, API calls, events, and side-by-side extensions are now common patterns [1], [2]. During these changes, it is essential to keep financial records, production orders, inventory updates, procurement paperwork, sales details, and asset information in sync. All of these areas need to stay in sync at every step of the process.

Quantum tech computing belongs outside that core. There is little architectural value in asking a quantum processor to save a purchase order or post an accounting document. The options worth considering appear later, when ERP data needs to help pick the best option from a huge range of possible plans - for example, a production sequence, a vehicle route, an inventory allocation, or a network design.

Research in logistics and supply-chain optimization already uses hybrid workflows in which classical software prepares the problem and constraints, a specialized solver performs the difficult search, and the business application receives the result [5]-[9]. In an ERP environment, that final hand-off is especially important. The solver may advise that ERP still owns the transaction.

Security creates a different timetable. A sufficiently capable error-correcting quantum computer would threaten the public-key schemes used in RSA- and elliptic-curve-based systems. NIST has already standardized the first set of post-quantum algorithms and is asking organizations to determine vulnerable cryptographic dependencies before migration becomes urgent [13]-[17].

For ERP teams, those dependencies are rarely confined to one server. They are found in TLS connections, middleware, APIs, partner gateways, digital signatures, identity services, secure file transfer, database clients, backups, and records that may need protection for many years. In other words, the chance to use advanced optimization can wait until there is clear proof it adds value, but building a complete cryptographic inventory cannot be postponed.



The current research work has identified the capability of quantum computing in ERP, supply chain planning, routing and production scheduling [3], [5]-[12]. Also, separate standards and guidance address crypto-agility and post-quantum migration [4], [13]-[19]. This approach makes all the areas remain largely disconnected. The literature gives less focus on only a single ERP architecture, which keeps the transaction processing inside the digital core. Such an architecture should route selected planning problems to classical, quantum-inspired, or quantum services through a common broker. It should validate solver results before execution and manage cryptographic changes across the same enterprise boundary. For an ERP, architects face two practical tasks. They need to connect new compute services without changing finance, inventory, production, or order processing. They need to replace cryptographic methods across interfaces without redesigning each business process. This paper addresses both tasks within one ERP modernization design.

Manufacturing is the main focus here because it makes the boundaries easy to see. This paper does not claim that quantum computing brings immediate benefits, introduce a new algorithm, or suggest that today's quantum hardware should handle ERP transactions. This paper focuses on a practical ERP problem. Manufacturing companies need a safe way to test quantum computing for planning and scheduling tasks. ERP must still control every business

transaction. The same architecture must support future changes to cryptographic standards. The paper uses the term “quantum-ready” for an ERP design that supports these needs without changing core transaction processing. In this context, “quantum-ready” means being able to identify which workloads are a good fit, switch between different computing methods, check results before they are used, and update cryptographic tools—all without having to rewrite core business processes. The novelty of this paper lies in bringing these areas into one ERP architecture. The proposed design keeps transaction control within ERP. Most published work stops at the solver. Weinberg et al. [7] study logistics, and Awasthi et al. [8] study production scheduling. Their work does not define what ERP should do with a solver result. Research on PQC focuses on cryptographic migration [4], [13]-[19]. This paper connects these two areas. Prior work gives useful results for logistics and production scheduling [7], [8]. It does not explain what should happen after a solver returns a plan to ERP. A schedule can look better mathematically and still violate plant capacity, approval rules, or current stock data. Post-quantum studies address a different problem, mainly cryptographic migration [4], [13]-[19]. The proposed framework puts these concerns in one ERP design. The solver proposes a plan. ERP checks it and posts the approved change. PQC changes stay outside the business logic. This combination is the main contribution of the paper.

ERP TRANSACTION CORE Orders Finance Inventory Production	DATA & INTEGRATION SERVICES APIs Events Data Products
DECISION / OPTIMIZATION SERVICES AI Classical Optimization Simulation	SPECIALIZED QUANTUM SERVICES Quantum-Inspired Hybrid QPU

Fig. 1 Separation of transactional and specialized compute roles

2. Background and Related Work

2.1. ERP Modernization as an Architecture Change

ERP modernization is often presented as an upgrade to a newer version or migrating to cloud infrastructure. In practice, the architectural changes are usually more important than the hosting decision. Direct point-to-point interfaces can be reduced, extensions can be separated from core transaction processing, and shared API, event, and data services can be introduced. Research on cloud ERP and microservice-based ERP links these patterns to greater flexibility, scalability, easier maintenance, and better interoperability, but it also notes the extra operational and security complexity of distributed systems [1], [2].

This matters concerning quantum readiness for a simple reason: it is much easier to swap out experimental computing services when they are placed behind a clear, well-defined interface. If planning logic is embedded in custom code that reads and updates ERP tables directly, testing another solver can disturb the posting path. If the same planning request is

exposed through a governed service, the solver can be changed while the transaction logic stays where it is. A normal modernization decision - whether to decouple a planning capability - can therefore create a useful option for future specialized compute.

2.2. Quantum Computing in an Enterprise Context

For enterprise applications, quantum calculation technology is relevant only to certain kinds of problems. A large search space is not enough by itself; the problem also has to be formulated in a way that a quantum, hybrid, or quantum-inspired method can use.

Routing, scheduling, logistics, and inventory optimization appear frequently in the literature [5]-[9]. These are decision problems, not transaction-processing problems. A QPU is a specialized compute resource, not a substitute for the database, application server, workflow engine, or controls that maintain business state.

Current hardware makes a hybrid pattern the sensible starting point. Reviews of logistics applications describe classical preprocessing along with constraint management around quantum or quantum-inspired optimization [5]. Weinberg et al. [7] demonstrated a hybrid workflow for a commercially relevant multi-truck routing problem. Awasthi et al. [8] studied quantum-classical optimization for production scheduling, and Fedouaki et al. [9] surveyed mixed approaches across several supply-chain problem categories. None of those examples requires the enterprise application itself to become quantum. The bounded optimization problem is what crosses the service boundary.

Industry experiments point in much the same direction. IBM has discussed routing, disruption, and supply-chain optimization as possible quantum use cases [10], [11]. SAP and Fujitsu connected enterprise applications to Fujitsu's quantum-inspired Digital Annealer in a proof of concept that included manufacturing scheduling and vehicle-space optimization [12]. These examples are not meant to prove that quantum hardware is already the best choice for production. Instead, their real value is in showing that things like breaking workloads into parts, making clear integration boundaries, and swapping out solvers can already be tested today.

Table 1. How computing responsibilities are divided in a quantum-ready ERP system

Compute role	Best suited to	ERP relationship	Architectural rule
ERP transaction processing	Postings, workflows, master data, controls	System of record and execution	Keep deterministic and isolated from experimental compute
AI/analytics	Prediction, classification, anomaly detection, forecasting	Advisory and analytical services	Consume through governed APIs and data products
Classical optimization	Mature scheduling, routing, and planning solvers	Primary optimization path	Remain the benchmark and default when it meets business needs
Quantum / hybrid optimization	Selected combinatorial or simulation problems	Specialized external service	Call through an abstraction layer; never allow direct transactional writes

Source: Author synthesis. ERP modernization patterns are informed by [1], [2]; quantum and hybrid optimization examples are drawn from [5]-[12].

2.3. The Post-Quantum Security Dimension

The security problem does not depend on whether a company ever sends an optimization job to a QPU. Shor's algorithm establishes the threat to public-key cryptosystems based on integer decomposition and discrete logarithms [20]. Grover's algorithm has a different effect on symmetric search and does not create the identical direct break of modern symmetric encryption [21]. For a large ERP estate, the practical issue is timing. Asymmetric cryptography is embedded in many products and integrations, so replacing it is likely to be a multi-year dependency-management exercise rather than a single security patch.

NIST finalized FIPS 203 (ML-KEM), FIPS 204 (ML-DSA), and FIPS 205 (SLH-DSA) in 2024 [13]-[15]. Its migration material places strong emphasis on discovery, cryptographic inventory, prioritization, and staged transition [16], [17]. CISA has also addressed automated discovery and inventory for PQC migration [18]. An ERP inventory, therefore, needs to reach beyond the ERP server itself. Clients, integration runtimes, identity platforms, partner gateways, and operational-technology interfaces may all contribute cryptographic dependencies to the same end-to-end business flow.

Product support is starting to turn these ideas into real solutions that architects can actually see and work with. For

example, in its Q2 2026 release, SAP HANA Cloud added post-quantum protection for TLS by using hybrid ML-KEM key establishment with clients that support it, while still supporting older clients with standard methods [19]. This implementation serves as an architectural example—it is not meant to suggest that every ERP product has this feature. What it demonstrates is a practical way to migrate: add PQC beneath the application layer, maintain compatibility during the transition, and provide enough connection details so you can tell which security path was used.

2.4. What Prior Work Leaves Open

Work on quantum computing in manufacturing has concentrated on difficult planning problems. Production scheduling and supply-chain logistics appear often in the published studies [7]-[9]. These papers show that quantum or hybrid methods can take part in the calculation. They say much less about what happens after the calculation reaches an ERP system.

A solver can return a schedule that looks correct but cannot be executed. ERP may show that a machine is unavailable or that a required component is short. In that case, the schedule must be rejected or revised. The paper assigns this decision to ERP, not to the external solver. It then places post-quantum security controls within the same architecture.

3. Materials and Methods

The approach I took to draw the architecture was more qualitative, rather than using experimental quantum techniques. The reason is simple: this is a design problem, not a hardware performance problem. My goal was to take proven ERP modernization strategies, quantum optimization research, and PQC migration advice and turn them into a reference design that real transformation programs can use. Instead of benchmark figures, the outcome is a well-defined set of boundaries, roles, and adoption options.

The source material was organized into three main evidence sets: ERP modernization ([1], [2]), quantum and hybrid optimization ([5]–[12]), and post-quantum migration and crypto-agility guidance ([4], [13]–[19]). Grouping the material in this approach made it easier to organize the findings. Moreover, made it easier to create a practical reference design. A typical manufacturing environment was divided into four main areas: transactions, integration and data, decision computing, and security. Each manufacturing workload was assigned to the layer best suited to handle it. Transaction integrity remained within the ERP system, while more advanced calculations were managed outside of it. The design was evaluated using four practical tests: The ERP transaction path should be consistent, and the solver should be replaceable without requiring changes to the business process. Any solver output must be validated before becoming an actual transaction, and cryptographic controls should be easy to locate and update without changing the core business logic. These are not formal proofs—they are

real-world architecture checks that any project team should consider before letting an experimental service interact with production systems.

Whenever something changes—like a sale, a production run, or a new purchase—all of the related records need to stay in sync. That way, information in finance, inventory, and assets always matches up, and nothing is missed.

4. Results and Discussion

4.1. Design Principles for Quantum-Ready ERP

The reference design is built on six main ideas, but they are not meant to be strict, separate rules. The most important point is that the ERP system keeps its role as the main record-keeper and manages all key transactions. Sophisticated decision-making problems are exposed through services or governed data products, not by letting apps dive straight into the tables. The application does not ask for a particular quantum provider—it simply requests the business outcome it needs. Choosing which solver to use happens in a broker layer, and the solver's answer is only a suggestion until it is checked against the company's own rules. Crypto-agility is not managed inside a single ERP module; instead, it is built into the trust and integration services that connect everything. There is always a classical (non-quantum) route available as a backup. Without this safety net, relying on experimental compute services could put business continuity at risk, which is exactly what this architecture aims to prevent.

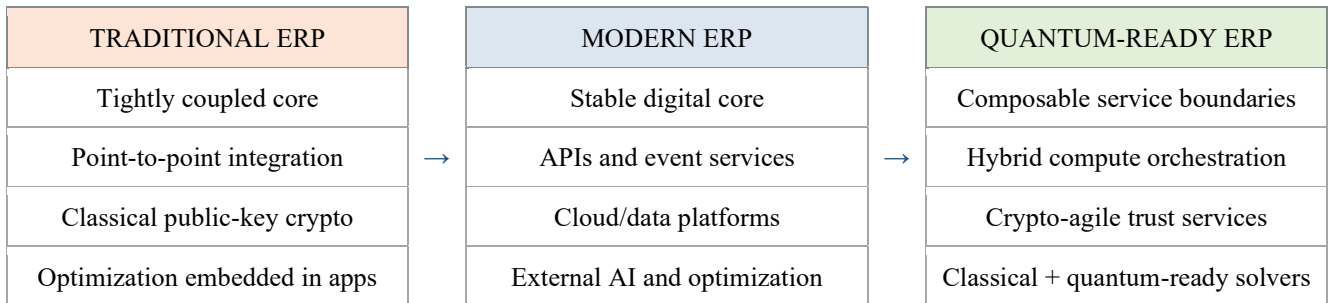


Fig. 2 ERP modernization path toward quantum readiness

4.2. Quantum-Ready ERP Reference Architecture

Figure 3 shows where the main architectural boundary sits. The ERP digital core—including finance, procurement, manufacturing, inventory, sales, and asset management—remains responsible for business execution. Only approved data and services leave this core, passing through the combination and data layer. The business problem is packaged before it ever reaches the decision or workload broker. Quantum capabilities, as a result, are positioned beneath the orchestration layer—not alongside ERP modules as just another app. This setup guarantees the business system stays in control, even when an external solver is involved in planning.

The workload broker acts as a buffer between the ERP system and the technology used for solving problems. For example, a production planning request might go to a standard classical optimizer most of the time, use a quantum-inspired solver for a special test, rely on a simulator during development, or connect to a quantum processor in the future. The ERP system does not have to make a different call for each option—the only thing it provides is the scheduling problem and the result it needs. By keeping this contract consistent, it is easier to compare different solvers, since each one is tested with the same request, and the rest of the process does not have to change.

BUSINESS & INDUSTRY CHANNELS Users Suppliers Customers Plants Warehouses OT / IoT		
ERP DIGITAL CORE Finance Procurement Manufacturing Inventory Sales Asset Management		
INTEGRATION & DATA PLANE API Management Events Integration Services Governed Data Products		
DECISION / WORKLOAD BROKER Problem packaging Solver selection Policy Cost / latency checks Result validation		
CLASSICAL SOLVER MIP heuristics simulation	QUANTUM-INSPIRED annealing specialized optimization	QUANTUM SERVICE simulator hybrid runtime QPU
ENTERPRISE VALIDATION GATE Feasibility business rules authorization explainability fallback		
SECURITY & TRUST PLANE Identity TLS PQC readiness Certificates Key management Cryptographic inventory / CBOM Observability		

Fig. 3 Proposed quantum-ready ERP reference architecture

The security and trust layer is designed to span the entire system. If cryptography were seen as just the network team's job, too many important areas would get overlooked. In an ERP environment, cryptography is not only used for TLS—it also protects interface authentication, signed documents, code signing, encrypted files, partner certificates, client libraries, key management, and connections with banks or shipping companies. That is why the architecture brings together identity, certificates, crypto policies, inventory, monitoring, and migration controls in a single layer. Simply setting up TLS on a server does not cover the full range of security needs.

4.3. Hybrid Workload Execution Pattern

Figure 4 makes more sense when you look at it as a series of steps rather than just a set of boxes. Imagine demand changes and a planner needs a new production schedule. The ERP system starts by collecting all the necessary business inputs—open orders, available materials, capacity, changeover limits, service priorities, and any commitments that cannot be broken. That integration layer then bundles this information as a request that works with any solver. Only after this step does the broker pick which path to use for execution. From the business perspective, all that matters is receiving a workable schedule—it does not need to know whether the calculation was done with a classical approach or a quantum-powered service.

A classical approach remains part of the design, even after a quantum option is added. If a new solver is not available, is too expensive, takes too long, or does not meet the required standards, the process can always revert to the classical method. This is an intentional, careful choice—production planning should not stop just because an experimental service cannot provide what is needed on time.

Solver output also goes through a validation checkpoint. Even if a schedule makes sense on paper, it could still violate a business rule, miss a regulatory requirement, or be based on information that's no longer current. That is why feasibility, authorization, policy, and other enterprise controls must be checked before any recommendation turns into action in the ERP system. Only the ERP is allowed to create the final production order, stock transfer, purchase order, or transport instruction. This marks the boundary between decision support and true authority over transactions. Each solver output passes through an enterprise validation gate before ERP execution. The enterprise validation gate checks every solver result before ERP accepts it. A returned schedule is not ready for posting. ERP compares it with the latest plant data. A machine may be unavailable, or a required component may be short. Such a proposal is rejected. The plan then goes back for review or another solver run. ERP records the solver name and the reason for acceptance or rejection. Only an approved plan can create the production order.

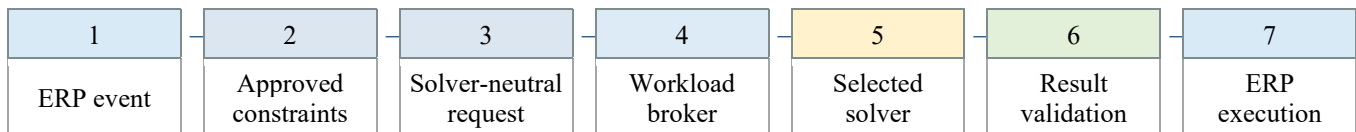


Fig. 4 Hybrid workload execution pattern

4.4. Manufacturing Industry Application

Manufacturing makes this separation clear, since the same environment has to handle both rigid transactional processes and the messier, more unpredictable challenges of planning. A plant network may have to decide which facility makes a product, sequence jobs across machines, respect labor and material limits, work around maintenance windows, meet customer dates, and reduce changeover or transport cost. ERP is well-suited to holding those objects and their current state. Searching through the combinations created by all of those constraints is a different computing task.

The framework clearly separates these two roles. The ERP system provides all the necessary information—like demand, inventory, capacity, routings, lead times, and commitments—for a specific optimization run. The optimization service gets only the data it needs for the problem and sends back a proposed plan. Before that plan can affect actual operations, it is compared to results from the established classical approach. This comparison matters: “quantum-ready” means the architecture is flexible and can use different methods—it does not mean there is always a preference for the quantum route. Research in production scheduling and supply chains supports this kind of carefully bounded hybrid setup.[7]-[9].

Table 2. Manufacturing use cases and their position in the architecture

Business area	ERP provides	Optimization problem	Possible compute path	ERP receives
Production scheduling	Orders, routings, work centers, capacities	Sequence jobs while reducing lateness and changeovers	Classical first; hybrid/quantum candidate	Validated production sequence
Network planning	Plants, DCs, lanes, demand, capacity	Allocate demand and flows across a constrained network	Classical or hybrid optimization	Recommended source and flow plan
Transportation	Deliveries, weights, time windows, carriers	Route vehicles and loads under constraints	Classical / annealing/hybrid	Route or load recommendation
Inventory allocation	Stock, service levels, demand, lead times	Allocate limited inventory across locations and priorities	Classical; selected quantum methods under study	Allocation proposal
Maintenance coordination	Asset plans, downtime, labor, production demand	Coordinate maintenance windows with production capacity	Classical/hybrid experimentation	Maintenance schedule proposal

Note: “Quantum-ready” describes a state of the architecture, not just a feature or add-on, not a preference concerning quantum execution. Solver selection should remain evidence-based and benchmarked against the established classical method.

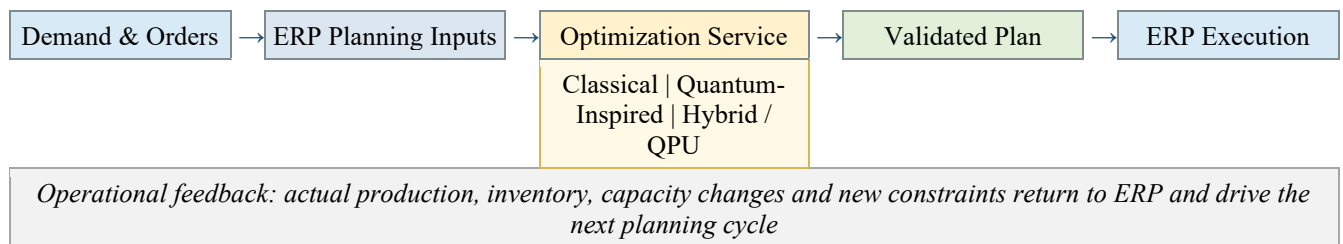


Fig. 5 Manufacturing optimization loop with ERP retained as the execution authority

4.5. Quantum-Safe Security within the ERP Modernization Program

The security work starts with discovery. Replacing algorithms before knowing where they are used is likely to produce a patchwork migration. NIST describes cryptographic inventory as a way to record algorithms, protocols, keys, certificates, components, and data flows that depend on them [16]. For ERP, the useful unit of analysis is often an end-to-end business interface rather than a single

server. Take a payment flow as an example—it might move through the ERP system, middleware, API management, network controls, a certificate authority, and finally a bank gateway. Updating just one part of this chain does not make the whole process quantum-safe if any other piece still relies on weak cryptography.

Modernization helps ensure that any future migration is smoother and less disruptive. New interfaces should avoid

tying cryptographic choices directly to business logic when the platform gives another option.

Trust stores, TLS policy, key-management services, certificate automation, signing services, and client libraries can be governed separately from the business function. That separation is what crypto-agility is intended to provide [4]. It does not remove the work of migration, but it reduces the number of places where a cryptographic change becomes a business-code change.

In SAP HANA Cloud Q2 2026, TLS 1.3 clients with the required support can negotiate hybrid ML-KEM [19]. A client without that support falls back to the classical path. During migration, both connection types can exist at the same time. The negotiated algorithm recorded for each connection becomes the key evidence. It shows which clients still rely on older cryptography. ERP teams can then trace those clients to the related library, middleware component, or partner interface. The work is not finished until those connections have been reviewed.

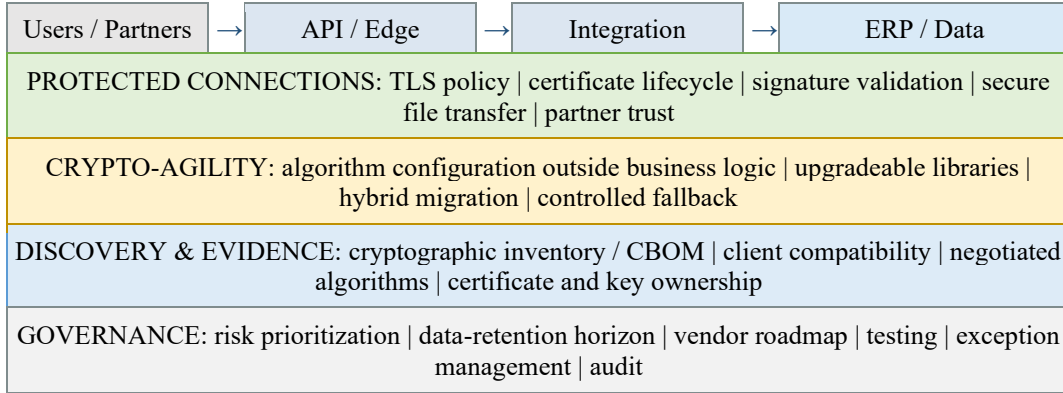


Fig. 6 Quantum-safe security insertion points across the ERP ecosystem

4.6. Adoption Roadmap

Quantum readiness is better handled as part of the overall modernization backlog, rather than as a separate “quantum program.” The first steps are standard architecture tasks: identify decision problems where advanced optimization could make a difference, and map out the cryptographic dependencies that already exist in the system. After that, decouple certain planning functions using APIs, events, or governed data products so that computation can take place outside the ERP core. At the same time, assign responsibility for certificates and keys, create a cryptographic inventory, and develop a roadmap for PQC migration. None of these steps requires a production-ready quantum computer, which is exactly why it makes sense to tackle them now, before one is needed.

Only after a stable boundary exists does a quantum pilot become meaningful. The same problem definition can then be run against a classical solver, a quantum-inspired method, a simulator, or hardware-backed execution. A one-off proof

of concept that depends on custom integration may demonstrate a solver, but it does not demonstrate an enterprise architecture. With the broker pattern, a future method that proves useful can be added as another execution option instead of forcing another ERP redesign.

Operationalization comes later and looks surprisingly conventional: service levels, cost controls, access policies, observability, solver and model versioning, incident procedures, and a documented fallback route. Microsoft’s Quantum Ready material is centered on assessing business impact, defining a strategy, and building organizational readiness [22]. IBM’s Quantum Business Foundations material similarly discusses industry use cases and the steps organizations can take to become quantum-ready [23]. The implication for ERP is simple. Before quantum compute becomes a production concern, readiness is mainly an architecture and operating model concern.

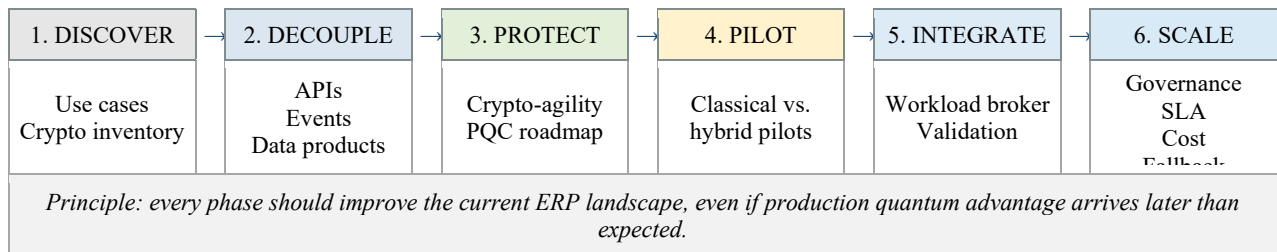


Fig. 7 Phased roadmap for quantum-ready ERP modernization

5. Architectural Trade-offs and Limitations

Optionality has a cost. The workload broker, external solver interface, validation service, cryptographic inventory, and observability layer all have to be operated, patched, monitored, and supported. Adding those components only to attach a ‘quantum-ready’ label to ERP would be difficult to justify. They earn their place when they also improve today’s integration discipline, solver portability, security visibility, or analytical capability. If they do not, the design has created complications without enough current value.

Choosing the right problems is another important limit—many ERP tasks should always stay on proven, classical systems. Things like financial postings, master data checks, workflow approvals, authorization, and most reporting are clear examples. Even when it comes to optimization, a well-tuned classical solver might still win out in terms of cost, speed, reliability, or quality of results. That is why the framework treats quantum computing as just one optional tool, not the default. A problem does not become a good fit concerning quantum just because it is challenging.

Moving data can be just as much of a bottleneck as solver performance. Sending a large set of transactional data to an outside service can be slow, costly, or run into security and data-sovereignty issues. Whenever possible, the broker should get only the specific variables needed for the optimization—not a full dump of master and transaction data from the ERP system. Sometimes, a simplified mathematical version of the problem is enough. Even as quantum hardware gets better, strict requirements for speed or data residency may still make a local classical or quantum-inspired solver the better choice.

PQC migration has the opposite problem: it impacts a broad ecosystem, and that ecosystem will not move in one release. Operating systems, databases, middleware, identity platforms, client libraries, network components, and partner systems all have their own upgrade schedules. A single ‘PQC complete’ milestone can therefore create false confidence. Readiness is better managed as a living inventory and compatibility program that is revisited as standards and products change [4], [16]-[19]. The literature review used here is targeted rather than systematic; its purpose is to

support the architecture synthesis, not to provide an exhaustive survey of quantum optimization research.

6. Conclusion

A quantum-ready ERP system is not just an ERP platform with a quantum computer attached. Instead, it is about how the whole architecture is set up. The ERP system still keeps track of business records and handles all controlled transactions. Quantum and quantum-based services operate separately and are only used for certain decision-making tasks. Even then, their results must pass through enterprise validation before anything alters in the system. This measured boundary is what allows for experimentation without letting experimental technology become part of the core transaction procedure.

A factory may keep its existing production planning process and test a new solver beside it. The same scheduling request can be sent to the current classical engine and to a quantum service. Their results can then be compared against the same plant data. ERP accepts only the result that passes its posting and approval rules. This lets the company test quantum computing without changing the transaction process.

The security timeline is more concrete. While post-quantum standards are already in place, picking a new algorithm is just one piece of a much bigger migration effort for large enterprises. The real challenge is finding every cryptographic dependency, figuring out how long sensitive data needs to stay protected, and coordinating upgrades across all connected systems—without disrupting day-to-day operations. By including cryptographic inventory and agility in the ERP modernization program, architecture and security teams gain a shared framework to handle this complex task. The key advice is to move forward carefully. Keep the core transaction systems secure. Handle complicated decisions through clearly defined services. Always have a classical backup in place. Make sure everyone understands where cryptography is used, and do not tie the business process to just one type of solver. These practical steps make it easier to add quantum services in the future, without forcing ERP systems to do things they were not built for.

References

- [1] Serhii Slivka, “Microservices Architecture for ERP Systems,” *Bulletin of Cherkasy State Technological University*, vol. 29, no. 4, pp. 32-42, 2024. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [2] Chulhyung Lee, Hayoung Fiona Kim, and Bong Gyou Lee, “A Systematic Literature Review on the Strategic Shift to Cloud ERP: Leveraging Microservice Architecture and MSPs for Resilience and Agility,” *Electronics*, vol. 13, no. 14, pp. 1-31, 2024. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [3] Poshan Kumar Reddy Ponnammreddy, “Quantum Computing Integration in Enterprise Resource Planning Systems: A Strategic Framework for Digital Transformation,” *Journal of Information Systems Engineering and Management*, vol. 10, no. 62s, pp. 406-415, 2025. [[CrossRef](#)] [[Publisher Link](#)]

- [4] Elaine Barker et al., “Considerations for Achieving Crypto Agility,” *Cybersecurity White Paper*, vol. 39, pp. 1-41, 2025. [[Google Scholar](#)]
- [5] Frank Phillipson, “Quantum Computing in Logistics and Supply Chain Management: An Overview,” *arXiv*, pp. 1-52, 2024. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [6] Hansheng Jiang, Zuo-Jun Max Shen, and Junyu Liu, “Quantum Computing Methods for Supply Chain Management,” *2022 IEEE/ACM 7th Symposium on Edge Computing (SEC)*, Seattle, WA, USA, pp. 400-405, 2022. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [7] Sean J. Weinberg et al., “Supply Chain Logistics with Quantum and Classical Annealing Algorithms,” *Scientific Reports*, vol. 13, no. 1, pp. 1-18, 2023. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [8] Abhishek Awasthi et al., “Real World Application of Quantum-Classical Optimization for Production Scheduling,” *2024 IEEE International Conference on Quantum Computing and Engineering (QCE)*, Montreal, QC, Canada, pp. 239-244, 2024. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [9] Fayçal Fedouaki et al., “Quantum Computing for Supply Chain Optimization: Algorithms, Hybrid Frameworks, and Industry Applications,” *Logistics*, vol. 10, no. 3, pp. 1-24, 2026. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [10] IBM Institute for Business Value, Exploring Quantum Computing Use Cases for Logistics, IBM, 2022. [Online]. Available: <https://www.ibm.com/thought-leadership/institute-business-value/report/quantum-logistics>
- [11] IBM Institute for Business Value, Quantum Computing in Supply Chain: Preparing for Near-Term Threat Management and Planning for Longer-Term Operational Advantage, IBM, 2023. [Online]. Available: <https://www.ibm.com/thought-leadership/institute-business-value/en-us/report/quantum-computing-supplychain>
- [12] Paul McElligott, A Quantum-Inspired Solution for Enterprise Optimization, SAP News, 2024. [Online]. Available: <https://news.sap.com/2024/03/quantum-inspired-computing-enterprise-optimization/>
- [13] G.M. Raimondo, and L.E. Locascio, “FIPS 203: Module-Lattice-based Key-Encapsulation Mechanism Standard,” *Federal Information Processing Standards Publications*, vol. 203, pp. 1-56, 2025. [[Google Scholar](#)]
- [14] Gina M. Raimondo, and Laurie E. Locascio, “FIPS-204: Module-Lattice-based Digital Signature Standard,” *Federal Information Processing Standards Publications*, vol. 204, pp. 1-65, 2025. [[Google Scholar](#)]
- [15] G.M. Raimondo, and L.E. Locascio, “FIPS-205: Stateless Hash-based Digital Signature Standard,” *Federal Information Processing Standards Publications*, vol. 205, pp. 1-61, 2025. [[Google Scholar](#)]
- [16] National Institute of Standards and Technology, National Cybersecurity Center of Excellence, Frequently Asked Questions about Post-Quantum Cryptography, 2026. [Online]. Available: <https://pages.nist.gov/nccoe-migration-post-quantum-cryptography/>
- [17] Dustin Moody et al., “Transition to Post-Quantum Cryptography Standards,” *NIST*, pp. 1-22, 2024. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [18] Cybersecurity and Infrastructure Security Agency, Strategy for Migrating to Automated Post-Quantum Cryptography Discovery and Inventory Tools, 2024. [Online]. Available: <https://www.cisa.gov/resources-tools/resources/strategy-migrating-automated-post-quantum-cryptography-discovery-and-inventory-tools>
- [19] Thomas Hammer, What’s New in SAP HANA Cloud - July 2026, SAP Community, 2026. [Online]. Available: <https://community.sap.com/t5/technology-blog-posts-by-sap/what-s-new-in-sap-hana-cloud-july-2026/ba-p/14419199>
- [20] Peter W. Shor, “Algorithms for Quantum Computation: Discrete Logarithms and Factoring,” *Proceedings 35th Annual Symposium on Foundations of Computer Science*, Santa Fe, NM, USA, pp. 124-134, 1994. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [21] Lov K. Grover, “A Fast Quantum Mechanical Algorithm for Database Search,” *Proceedings of the 28th Annual ACM Symposium on Theory of Computing*, NY, United States, pp. 212-219, 1996. [[CrossRef](#)] [[Google Scholar](#)] [[Publisher Link](#)]
- [22] Microsoft, Quantum Ready | Get Started, Microsoft Quantum, 2026. [Online]. Available: <https://quantum.microsoft.com/en-us/quantum-ready/get-started>
- [23] IBM Quantum Learning, Quantum Business Foundations, IBM. [Online]. Available: <https://learning.quantum.ibm.com/course/quantum-business-foundations>