International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 18(20s) 2026 pp. 1599-1607
https://doi.org/10.70917/ijcisim-2026-6020
Article

Architectural Controls for Secure Retrieval-Augmented Generation in GxP-Regulated Pharmaceutical Research: A Compliance-Driven Framework for Protecting Biomedical Intellectual Property

Rajashekar Reddy Gujjula

Independent Researcher, USA

Abstract: Pharmaceutical research and development organizations are adopting retrieval-augmented generation (RAG) systems for drug discovery knowledge retrieval, regulatory document analysis, and clinical literature synthesis — yet GxP-regulated deployments expose compliance vulnerabilities that general AI security frameworks leave unaddressed. Audit trail continuity under 21 CFR Part 11, cross-tenant intellectual property isolation between co-hosted research programs, and validated-system-state preservation impose obligations at each pipeline processing layer, not only at network perimeters. This paper develops a compliance-driven architectural control framework assigning discrete GxP controls to four RAG pipeline layers — ingestion, vector store, retrieval gate, and generation — and maps each control to the specific Part 11 provisions and ALCOA+ data integrity attributes it satisfies. Infrastructure as Code deployment patterns allow the complete pipeline to be qualified as a GxP computerized system under Computer Software Assurance principles. Layer-specific controls provide a more defensible regulatory posture than perimeter-only approaches precisely because GxP obligations bind to data provenance and audit trail continuity at the processing stage, not the access boundary.

Keywords: retrieval-augmented generation, GxP compliance, 21 CFR Part 11, pharmaceutical informatics, cloud security architecture, ALCOAI

1. INTRODUCTION

Compound libraries, genomic screening outputs, clinical trial records, and regulatory submission packages collectively constitute the most commercially sensitive data assets a pharmaceutical organization holds. Connecting large language models to these corpora through retrieval-augmented generation architectures offers a genuine productivity case: drug discovery precedent surfaced without fine-tuning on proprietary data, regulatory document queries answered from controlled internal sources, and clinical literature patterns synthesized at a speed manual review cannot match [1][2]. The architectural challenge is not whether RAG can work in pharmaceutical research — demonstrations in clinical knowledge encoding [3] and biomedical foundation models [4] establish that it can. The challenge is whether it can work without creating compliance exposure that outweighs the operational benefit.

GxP-regulated environments impose obligations on pharmaceutical AI systems that the AI security literature does not systematically address [5][6]. Under FDA's 21 CFR Part 11 [7], any computerized system that creates, modifies, or retrieves regulated electronic records must maintain secure, computer-generated, time-stamped audit trails at every stage of data processing — not simply at the system boundary. ALCOA+ data integrity principles [8] require that every retrieved data element be Attributable to a specific user or system, Contemporaneous with the action



Copyright: © 2026 by the authors

that produced it, and demonstrably Original. These requirements attach to the retrieval and generation stages of a RAG pipeline in ways that network-perimeter controls and model-level adversarial robustness measures [9] cannot satisfy. The audit trail gap between retrieval execution and generation output is a regulatory exposure unique to generative AI deployments in GxP contexts.

No published framework maps RAG architectural controls to Part 11 provisions and ALCOA+ attributes at the pipeline-layer level. This paper develops such a framework, organized around four discrete control layers: ingestion, vector store, retrieval gate, and generation. Each layer carries its own compliance obligations and requires its own set of engineered controls, independent of what other layers provide. Implementation patterns grounded in Infrastructure as Code and validated through Computer Software Assurance-aligned qualification protocols [10] make the framework operationally deployable within existing enterprise cloud platform engineering practices.

2. BACKGROUND AND RELATED WORK

A. RAG in Pharmaceutical Research

The convergence of human expertise and AI-enabled knowledge retrieval in clinical and biomedical settings [11] has created conditions for RAG adoption across pharmaceutical discovery, translational, and regulatory domains. Singhal et al. contend that large language models encode clinical knowledge at levels approaching expert performance on standardized assessments [3], establishing a capability baseline that makes RAG-augmented pharmaceutical research systems technically credible. Thirunavukarasu et al. argue that the shift from narrow diagnostic AI to general knowledge infrastructure in medicine is already underway, with implications for how organizations architect and govern AI deployments [12]. Moor et al. demonstrate that foundation models for generalist medical AI substantially outperform task-specific alternatives on retrieval-heavy benchmarks [4] — this held even when retrieval corpora were relatively small, which was not anticipated at the design stage of those evaluations. Rajpurkar et al. establish that AI in health and medicine has moved decisively beyond single-task applications toward knowledge management infrastructure [13], a framing that directly motivates the security architecture question this paper addresses. The compliance infrastructure for these deployments has received limited systematic treatment.

B. GxP Compliance and Validated Systems

FDA's 21 CFR Part 11 establishes electronic records requirements across the pharmaceutical enterprise [7]. The provisions most directly implicated in RAG deployments are §11.10(d), restricting system access to authorized individuals, and §11.10(e), requiring secure, time-stamped audit trails for all actions that create, modify, or delete regulated records. The ALCOA+ framework operationalizes Part 11 data integrity expectations into attribute-level requirements: records must be Attributable, Legible, Contemporaneous, Original, and Accurate, plus complete, consistent, enduring, and available [8]. FDA data integrity guidance posits that these attributes must be preserved regardless of system architecture or processing method — a standard that a probabilistic text generation system meets only if explicit architectural controls enforce attributability and contemporaneity at the generation stage. GAMP 5 second edition and the FDA's Computer Software Assurance approach [10] endorse risk-based, automation-compatible validation for complex software systems, replacing exhaustive scripted testing with policy-as-code enforcement and qualified building blocks. NIST AI RMF 1.0 [14] and NIST SP 800-53 Rev. 5 [15] provide complementary security control guidance; neither addresses GxP-specific compliance obligations for AI-enabled pharmaceutical systems.

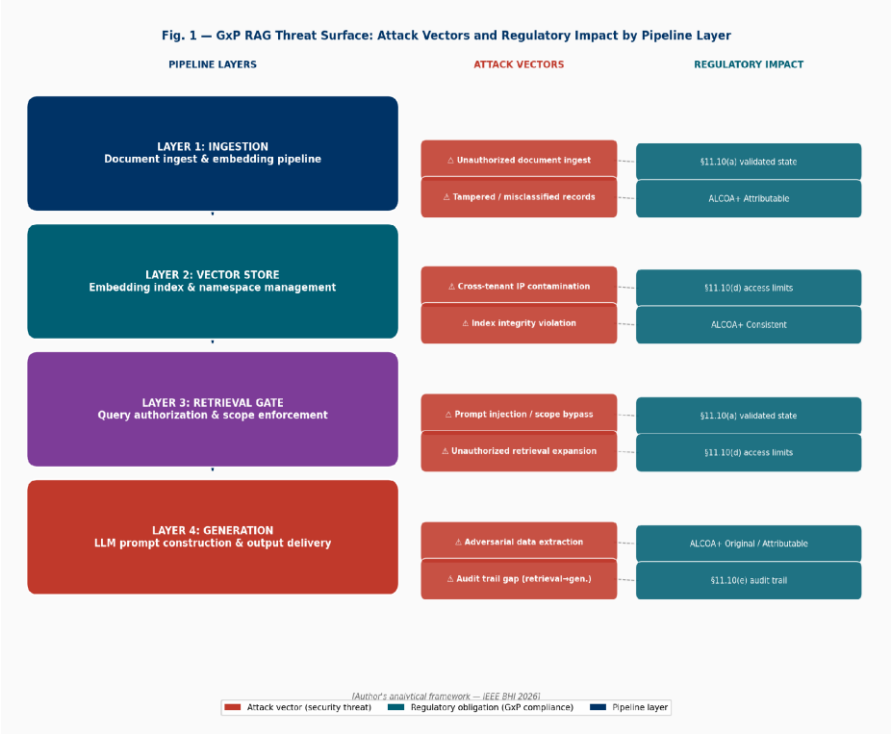
C. RAG Security Literature and the GxP Gap

Perez and Ribeiro establish that language models are susceptible to prompt injection attacks that redirect model behavior through adversarially constructed inputs [9] — a risk directly relevant to retrieval scope manipulation in pharmaceutical RAG deployments. Carlini et al. demonstrate that large language models can be induced to reproduce training data through targeted extraction attacks [5], raising a distinct IP exposure risk when pharmaceutical compound data has informed model training. Liu et al. survey privacy and security risks across the deep learning life cycle [6],

noting that data extraction, membership inference, and model inversion attacks represent active threat surfaces; none of these analyses engage with GxP compliance as a constraint on architectural responses. The harm taxonomy established by Suresh and Guttag [16] provides useful vocabulary for categorizing pipeline risks but was not developed for regulated data environments. Federated learning frameworks for health AI [17][18] address multi-institution data sharing constraints but target a different architectural pattern than centralized pharmaceutical RAG deployments. Large language model surveys [19] have noted the need for domain-specific deployment governance without specifying the architectural mechanisms required in regulated environments. The gap this paper fills is specific: an end-to-end GxP compliance mapping for RAG pipeline architecture, organized at the layer level.

3. THREAT MODEL FOR GXP-REGULATED RAG

Fig. 1. GxP RAG threat surface — pipeline layers annotated with attack vectors and regulatory impact.



Pharmaceutical RAG deployments face a dual threat surface. Conventional cybersecurity threats — unauthorized access, data exfiltration, service disruption — are present in any enterprise AI system. The second surface, distinctive to GxP-regulated environments, comprises regulatory compliance failures that trigger legal and inspection consequences independent of whether a security breach occurs. Both surfaces must be modeled and controlled.

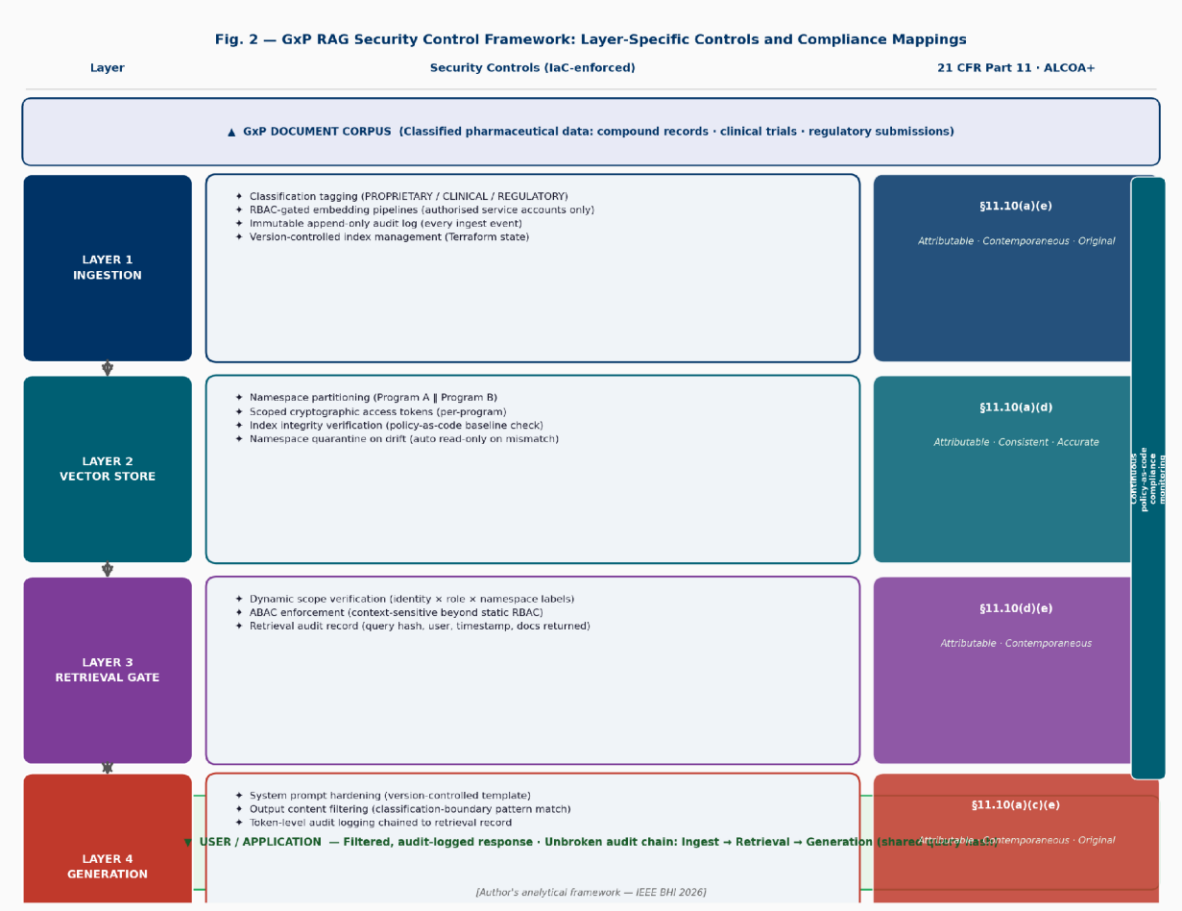
At the ingestion layer, unauthorized document ingest — adding unvetted, tampered, or misclassified records to the vector index — simultaneously introduces a security vulnerability and a validated-state integrity violation, because the index contents no longer correspond to the approved, qualified dataset. Cross-tenant contamination at the vector store presents a compound risk: retrieval requests from one research program surfacing documents belonging to another breach IP confidentiality and ALCOA's Attributable principle in the same event.

Prompt injection at the retrieval gate [9] is a mechanism for authorized users to exceed permitted retrieval scope without triggering the access control check a direct query would encounter. The generation layer adds risks absent from traditional pharmaceutical data systems: probabilistic synthesis can produce outputs containing protected compound information without those outputs appearing in the retrieval log, and adversarial extraction techniques [5] can elicit parametric reproduction even when retrieval scope is correctly restricted. Validated-state disruption is the

cross-cutting regulatory risk: any unauthorized pipeline configuration change creates a deviation from the qualified state that must be detected and resolved before the system can be used for GxP-relevant activities.

4. COMPLIANCE-DRIVEN ARCHITECTURAL CONTROL FRAMEWORK

Fig. 2. Four-layer RAG pipeline with GxP security controls annotated at each stage.



The framework assigns a distinct set of compliance-driven controls to each of the four RAG pipeline layers. Layer-level assignment reflects where specific Part 11 provisions and ALCOA+ attributes attach in data processing. Perimeter controls cannot satisfy an audit trail obligation that arises inside the pipeline between retrieval execution and generation output.

A. Ingestion Layer Controls

Every document entering the vector index receives a classification label at ingest: proprietary compound data, clinical trial record, regulatory submission artifact, or analogous category. This label propagates into embedding metadata and is evaluated at retrieval time — making the Attributable and Accurate ALCOA+ attributes verifiable throughout downstream processing. RBAC-gated embedding pipelines restrict ingest operations to authorized service accounts with defined roles; no ad-hoc or direct write to the vector index is permitted. Every ingest event writes an immutable audit record to an append-only log store: requestor identity, timestamp, document identifier, classification label, source system reference, and the IaC pipeline version that executed the ingest — satisfying Part 11 §11.10(e) at the ingestion stage. Cryptographic sealing of log records prevents retrospective modification. Version-controlled index management enforced through Terraform state files under peer-reviewed Git workflows ensures the vector index can be restored to any prior validated state [10].

B. Vector Store Controls

Research program namespacing partitions the vector store at the logical level: each program occupies an isolated namespace with independent access boundaries enforced by the storage service, not managed as a runtime policy susceptible to query manipulation. Cryptographic access tokens are scoped to the requesting user's authorized namespace set — a token for Program A carries no entitlement to query Program B's namespace, regardless of query content. Index integrity verification executes as a policy-as-code gate before any query is permitted to an environment: embedding model version, index schema version, and namespace configuration are validated against the approved baseline in the version control system. Namespace quarantine — automatic read-only restriction on query access — triggers when any integrity check returns a mismatch, preventing use of a potentially compromised index while investigation is pending.

C. Retrieval Gate Controls

Dynamic scope verification executes before any retrieval call reaches the vector store. The gate evaluates the requesting user's identity, current role assignment, session-level authorization context, and the classification labels in the target namespace — then permits or denies retrieval before any document is accessed. Attribute-based access control at this layer allows context-sensitive retrieval permissions that role-based controls alone cannot express: a user authorized for Phase II study data may be denied retrieval from Phase III records in the same program until project authorization is updated, without requiring a role schema change [15]. Every retrieval event generates a structured audit record: query hash, user identity, role at query time, timestamp, target namespace, and document identifiers returned — written to the same immutable audit store as ingestion records, satisfying Part 11 §11.10(e) and establishing the Attributable and Contemporaneous ALCOA+ attributes for every retrieved document.

D. Generation Layer Controls

System prompt hardening removes the adversarial surface that prompt injection exploits by loading the model's instruction context from a version-controlled, peer-reviewed template stored in the pipeline's configuration repository — not assembled dynamically from user input or retrieved documents [9]. Output content filtering applies classification-boundary pattern matching to every generated response before delivery: outputs matching known patterns for proprietary compound identifiers, clinical enrollment parameters, or formulation specifications trigger a redaction step that logs the event before delivering the filtered response. Token-level audit logging captures a structured record for every generation event: the hash of the system prompt template used, retrieval context identifiers incorporated into the generation context, the generation model version and endpoint identifier, and a cryptographic hash of the output text. This record, chained to the preceding retrieval audit record through the shared query hash, provides an unbroken audit trail from retrieval execution through generation output — satisfying Part 11 §11.10(e) across the complete pipeline.

TABLE I. Architectural Control-to-GxP Requirement Mapping

Layer	Control	Part 11 Provision	ALCOA+ Attribute
Ingestion	Classification tagging	§11.10(e) audit trail	Attributable, Accurate
Ingestion	RBAC-gated embedding pipelines	§11.10(d) access limits	Attributable
Ingestion	Immutable append-only audit log	§11.10(e) audit trail	Contemporaneous, Original
Ingestion	Version-controlled index mgmt	§11.10(a) validation	Consistent, Enduring
Vector Store	Namespace partitioning	§11.10(d) access limits	Attributable
Vector Store	Scoped cryptographic tokens	§11.10(d) access limits	Attributable
Vector Store	Index integrity verification	§11.10(a) validation	Accurate, Consistent

Vector Store	Namespace quarantine on drift	§11.10(a) validation	Consistent
Retrieval Gate	Dynamic scope verification	§11.10(d) access limits	Attributable
Retrieval Gate	ABAC enforcement	§11.10(d) access limits	Attributable
Retrieval Gate	Retrieval audit record	§11.10(e) audit trail	Attributable, Contemporaneous
Generation	System prompt hardening	§11.10(a) validation	Consistent
Generation	Output content filtering	§11.10(c) record protection	Accurate
Generation	Token-level audit logging	§11.10(e) audit trail	Attributable, Contemporaneous, Original

5. IMPLEMENTATION PATTERNS AND VALIDATION APPROACH

A. IaC-Based Deployment for Validated-State Reproducibility

Every pipeline component — ingestion service configurations, vector store namespace definitions, retrieval gate ABAC rules, generation endpoint bindings, output filter rule sets, and audit log storage configurations — is expressed in Terraform modules maintained in a Git repository under mandatory peer review and change approval gates. No configuration reaches a non-development environment through any path other than the approved CI/CD pipeline; out-of-band console changes are blocked by deny-override Azure Policy assignments at the subscription level. Staging environments are provisioned from identical IaC code as production — the only difference is parameter files specifying environment-specific values such as endpoint URLs and capacity allocations. Operational qualification tests execute automatically against the staging environment after each approved deployment, before production promotion is permitted. GAMP 5 second edition's building-block qualification principle [10] directly reduces validation overhead at scale: a qualified IaC module for a research program namespace, once approved, does not require re-qualification when instantiated for an additional program — provided the module code is unchanged.

B. Policy-as-Code Compliance Monitoring

The deployment pipeline evaluates compliance state continuously. Policy rules check that audit logging services are active and writing to the append-only store, that namespace isolation boundaries are intact, that retrieval gate rules match the configuration-repository baseline, and that generation model versions have not changed outside the change approval workflow. A policy violation in any of these checks prevents the affected environment from receiving query traffic until the drift is resolved and the environment is re-qualified — operationalizing validated-state maintenance for a continuously running AI service.

C. Shared Responsibility Obligations

When managed cloud services host inference endpoints or vector store backends, the regulated organization's compliance obligations do not transfer to the provider. Quality agreements must specify: the provider's obligation to notify the organization of service configuration changes within a defined window (30 days for planned changes; immediate notification for security-driven changes); audit log retention periods satisfying the organization's record retention schedule; and the provider's obligation to produce evidence records for regulatory inspection on request. Supplier assessment must verify ISO 27001 certification, SOC 2 Type II reports, and the provider's access control architecture [15].

D. Illustrative Application

To demonstrate framework operation, consider a medicinal chemistry team member querying a pharmaceutical RAG system for synthesis-route precedent on a proprietary compound class — a query that intersects Program A compound records and Program B clinical pharmacokinetic data. Under the proposed framework: (1) the ingestion layer has tagged Program A compound records PROPRIETARY-COMPOUND and Program B clinical records CLINICAL-TRIAL-PHASE-II; (2) the user's cryptographic access token is scoped to Program A only; (3) the retrieval gate evaluates the token scope against the query namespace and denies Program B retrieval, logging the denial event with user identity, timestamp, and denied namespace; (4) the generation layer executes against Program A documents only, with the system prompt loaded from the approved template, and logs the output hash chained to the retrieval audit record. The resulting audit trail — ingest record, retrieval grant, denial record, and generation log — satisfies Part 11 §11.10(e) across the full interaction and provides an attributable, contemporaneous record traceable from query to output.

TABLE II. Validation Evidence Matrix — IQ/OQ/PQ Activities Mapped to RAG Pipeline Components

Component	IQ Activity	OQ Activity	PQ Activity
Ingestion Pipeline	Verify Terraform module deployment; confirm RBAC config matches approved spec	Ingest test document; confirm classification label propagated; verify audit log entry	Run full-volume ingest; confirm audit log completeness under production load
Vector Store	Confirm namespace partitioning active; verify access token scope config	Test cross-namespace query rejection; verify integrity check detects baseline mismatch	Concurrent multi-tenant queries; confirm namespace isolation maintained
Retrieval Gate	Verify ABAC rules loaded from approved config; confirm audit log writing active	Test retrieval denial for unauthorized namespace; confirm audit record captured	Confirm gate performance under peak query volume; audit log completeness
Generation Endpoint	Verify system prompt loaded from approved template version	Test prompt injection resistance; verify output filter triggers on classification pattern	Adversarial prompt battery; verify all generation events logged
Audit Logging	Verify append-only storage configured; cryptographic sealing active	Confirm immutability: attempt overwrite; verify hash chain validation	Confirm log retention for production retention schedule; test evidence retrieval

6. DISCUSSION

Perimeter-first security is a rational design choice when regulatory obligations do not attach to internal processing steps — that is, for most enterprise applications. The argument this paper advances is narrow and specific: in GxP-regulated pharmaceutical environments, the internal processing steps of a RAG pipeline are themselves regulated. Part 11 §11.10(e) does not exempt the retrieval-to-generation transition from audit trail requirements because it is pipeline-internal; ALCOA+ does not excuse generation-stage outputs from Attributable and Contemporaneous requirements because they were produced probabilistically rather than retrieved deterministically. Layer-specific controls are the engineering response to these specific obligations.

The cost of this approach is real. Retrieval-gate audit logging and generation-layer output filtering introduce measurable processing overhead per query; token-level audit logging generates substantial log volumes in high-throughput deployments; index integrity verification adds a policy evaluation step before each query is permitted. These costs accumulate and must be evaluated against query volume and latency SLAs. Operating a GxP-relevant AI

system without Part 11-compliant audit trails, however, is not a latency optimization — it is a regulatory exposure that can invalidate research data generated by the system [8], a consequence whose operational and financial cost far exceeds query processing overhead.

The framework's principal limitation is analytical: it is evaluated against regulatory requirements and architectural principles, not empirically benchmarked in a production pharmaceutical RAG deployment. Production instrumentation of the proposed control set — measuring audit trail completeness, latency overhead per layer, and namespace isolation robustness under adversarial prompting — would substantially strengthen the evidentiary basis. As FDA guidance on AI-enabled systems evolves and EU AI Act high-risk classification extends to AI systems in regulated medical product development, layer-specific GxP controls will likely shift from architectural best practice to regulatory expectation.

7. CONCLUSION

Pharmaceutical organizations deploying RAG in regulated research environments face a compliance architecture problem that the existing AI security literature does not solve. GxP obligations under 21 CFR Part 11 and ALCOA+ attach to data provenance and audit trail continuity at each pipeline processing stage — making layer-specific architectural controls a regulatory necessity rather than a precautionary measure. The framework proposed here locates controls at the ingestion, vector store, retrieval gate, and generation layers, maps each control to the Part 11 provision and ALCOA+ attribute it satisfies, and grounds implementation in Infrastructure as Code patterns compatible with Computer Software Assurance-aligned qualification. The result is a RAG deployment architecture that is compliant by design — one that treats each pipeline layer as a regulated processing step rather than a trusted internal component. Pharmaceutical R&D organisations can adopt this framework as a structured basis for building GxP-defensible AI knowledge infrastructure.

REFERENCES

1. P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W. Yih, T. Rocktäschel, S. Riedel, and D. Kiela, "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," in Proc. NeurIPS, vol. 33, pp. 9459–9474, 2020. https://arxiv.org/abs/2005.11401
2. T. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, et al., "Language Models are Few-Shot Learners," in Proc. NeurIPS, vol. 33, pp. 1877–1901, 2020. https://arxiv.org/abs/2005.14165
3. K. Singhal, S. Azizi, T. Tu, S. S. Mahdavi, J. Wei, H. W. Chung, N. Scales, et al., "Large language models encode clinical knowledge," Nature, vol. 620, pp. 172–180, Aug. 2023. https://www.nature.com/articles/s41586-023-06291-2
4. M. Moor, O. Banerjee, Z. S. H. Abad, H. M. Krumholz, J. Leskovec, E. J. Topol, and P. Rajpurkar, "Foundation models for generalist medical artificial intelligence," Nature, vol. 616, pp. 259–265, Apr. 2023. https://www.nature.com/articles/s41586-023-05881-4
5. N. Carlini, F. Tramèr, E. Wallace, M. Jagielski, A. Herbert-Voss, K. Lee, A. Roberts, T. Brown, D. Song, Ú. Erlingsson, A. Oprea, and C. Raffel, "Extracting Training Data from Large Language Models," in Proc. 30th USENIX Security Symp., pp. 2633–2650, 2021. https://arxiv.org/abs/2012.07805
6. X. Liu, L. Xie, Y. Wang, H. Ding, J. Zou, J. Xiong and Z. Ying, "Privacy and Security Issues in Deep Learning: A Survey," IEEE Access, vol. 9, pp. 4566–4593, 2021. https://ieeexplore.ieee.org/document/9294026
7. U.S. Food and Drug Administration, "Part 11, Electronic Records; Electronic Signatures — Scope and Application," Guidance for Industry, U.S. FDA, Aug. 2003. https://www.fda.gov/regulatory-information/search-fda-guidance-documents/part-11-electronic-records-electronic-signatures-scope-and-application
8. U.S. Food and Drug Administration, "Data Integrity and Compliance With Drug CGMP: Questions and Answers," Guidance for Industry, U.S. FDA, Dec. 2018. https://www.fda.gov/regulatory-information/search-fda-guidance-documents/data-integrity-and-compliance-drug-cgmp-questions-and-answers
9. F. Perez and I. Ribeiro, "Ignore Previous Prompt: Attack Techniques For Language Models," NeurIPS 2022 ML Safety Workshop, Dec. 2022. https://arxiv.org/abs/2211.09527
10. ISPE, "GAMP 5: A Risk-Based Approach to Compliant GxP Computerized Systems," 2nd ed. Tampa, FL: ISPE, 2022. https://guidance-docs.ispe.org/doi/book/10.1002/9781946964571?__cf_chl_f_tk=pdKMNeNLumLCjMnjo_Cj6kqS9ec2IdZyOAFMl_F43EY-1782889402-1.0.1.1-xEALFzYu9uuHIt8koLMRGC1bmEixCB3.gYVZ3crBJVM
11. E. J. Topol, "High-performance medicine: the convergence of human and artificial intelligence," Nature Medicine, vol. 25, no. 1, pp. 44–56, Jan. 2019. https://www.nature.com/articles/s41591-018-0300-7

12. A. J. Thirunavukarasu, D. S. J. Ting, K. Elangovan, L. Gutierrez, T. F. Tan, and D. S. W. Ting, “Large language models in medicine,” Nature Medicine, vol. 29, no. 8, pp. 1930–1940, Aug. 2023. https://www.nature.com/articles/s41591-023-02448-8
13. P. Rajpurkar, E. Chen, O. Banerjee, and E. J. Topol, “AI in health and medicine,” Nature Medicine, vol. 28, no. 1, pp. 31–38, Jan. 2022. https://www.nature.com/articles/s41591-021-01614-0
14. National Institute of Standards and Technology, “Artificial Intelligence Risk Management Framework (AI RMF 1.0),” NIST AI 100-1, Jan. 2023. https://nvlpubs.nist.gov/nistpubs/ai/nist.ai.100-1.pdf
15. National Institute of Standards and Technology, “Security and Privacy Controls for Information Systems and Organizations,” NIST SP 800-53 Rev. 5, Sep. 2020. https://nvlpubs.nist.gov/nistpubs/specialpublications/NIST.SP.800-53r5.pdf
16. H. Suresh and J. Guttag, “A Framework for Understanding Sources of Harm throughout the Machine Learning Life Cycle,” in Proc. ACM FAccT, pp. 1–9, 2021. https://arxiv.org/abs/1901.10002
17. N. Rieke, J. Hancox, W. Li, F. Milletarì, H. R. Roth, S. Albarqouni, S. Bakas, et al., “The future of digital health with federated learning,” npj Digital Medicine, vol. 3, no. 1, p. 119, Sep. 2020. https://www.nature.com/articles/s41746-020-00323-1
18. G. A. Kaissis, M. R. Makowski, D. Rückert, and R. F. Braren, “Secure, privacy-preserving and federated machine learning in medical imaging,” Nature Machine Intelligence, vol. 2, no. 6, pp. 305–311, Jun. 2020. https://www.nature.com/articles/s42256-020-0186-1
19. S. Minaee, T. Mikolov, N. Nikzad, M. Chenaghlu, R. Socher, X. Amatriain, and J. Gao, “Large Language Models: A Survey,” IEEE Trans. Neural Netw. Learn. Syst., 2024. https://arxiv.org/abs/2402.06196