An End-to-End Data Quality Management Framework for Banking Systems with Generative AI Integration

Debabrata Pruseth
AI Architect & Applied AI Researcher
Singapore

Author Note

This article is a research-style companion version of the author's blog post "End-to-End Data Quality Management Framework (DQMF) in Banking with GenAI Integration"

Abstract

Data quality management is a foundational capability for banking systems because financial institutions rely on trusted data for customer onboarding, transaction processing, regulatory reporting, risk aggregation, anti-money laundering monitoring, credit decisioning, operational resilience, and enterprise decision-making. Poor-quality data can weaken customer trust, increase regulatory exposure, distort risk reporting, and reduce the reliability of downstream analytics and artificial intelligence systems. In modern banking environments, data is created, stored, transformed, used, archived, and deleted across a complex network of core banking systems, customer platforms, transaction engines, risk systems, compliance tools, finance systems, data warehouses, data lakes, and end-user computing assets. This complexity creates persistent challenges around accuracy, completeness, consistency, timeliness, validity, lineage, ownership, and traceability.

This paper presents an end-to-end Data Quality Management Framework for banking systems with Generative AI integration. The framework builds on the author's original blog article, which organizes data quality management across six lifecycle phases: data creation, storage, processing, usage, archival, and deletion. The blog proposes practical Generative AI applications across these stages, including intelligent data entry assistance, metadata generation, automated data classification, lineage documentation, anomaly interpretation, report drafting, policy analysis, privacy scanning, retention planning, archive summarization, erasure request orchestration, and deletion audit summarization.

The proposed research-style framework extends that lifecycle model into a formal banking architecture and governance approach. It positions Generative AI as an augmentation layer, not as an autonomous data controller. Generative AI can support data stewards, business owners, technology teams, risk managers, compliance teams, and operations users by

summarizing issues, interpreting policies, generating candidate rules, explaining lineage, and drafting evidence-based narratives. However, because banking data is sensitive and regulated, GenAI integration must be governed through human review, retrieval-augmented generation, role-based access controls, audit logging, data masking, model risk management, and clear accountability.

The paper concludes that banks should treat data quality management as both a regulatory capability and an AI-readiness capability. Trusted data is not only required for accurate reporting and compliance; it is also essential for safe and scalable Generative AI adoption.

---

1. Introduction

Banking is one of the most data-dependent sectors in the global economy. Every customer account, payment, trade, loan, risk exposure, regulatory submission, fraud alert, and financial report depends on data. If the data is incomplete, inaccurate, duplicated, stale, inconsistent, or poorly governed, the consequences can be serious. A wrong customer identifier can weaken Know Your Customer controls. Missing transaction attributes can reduce the effectiveness of anti-money laundering monitoring. Incorrect collateral data can distort credit risk calculations. Poor lineage can prevent a bank from explaining how a regulatory number was produced. Stale retention records can increase privacy and legal risk.

The author's blog article frames data quality as mission-critical in banking and notes that banks handle diverse categories of data, including customer information, transactions, and risk metrics. It also highlights the regulatory expectation that this data should be accurate, complete, timely, and well governed. This is consistent with the broader regulatory direction in banking, where risk data aggregation, data governance, operational resilience, privacy, and model risk management all depend on reliable enterprise data.

Traditionally, data quality management has been implemented through data profiling, validation rules, reconciliation, exception reporting, manual remediation, data stewardship, and governance forums. These practices remain essential. However, the volume and complexity of banking data have increased significantly. Banks now operate across digital channels, APIs, cloud platforms, data lakes, analytics environments, third-party ecosystems, and AI-enabled workflows. This has made data quality management more difficult, but also more important.

Generative AI introduces a new opportunity. Large language models can interpret text, summarize documents, classify information, generate code, draft reports, compare policy requirements, explain lineage, and assist users through natural language. In a data quality context, this means GenAI can help data teams identify issues faster, understand root causes, generate remediation suggestions, and translate technical findings into business language. The author's blog identifies several such use cases across the banking data lifecycle, including intelligent validation at data creation, automated classification during storage, AI-powered cleansing during processing, report generation during usage, retention planning during archival, and deletion audit summarization during data deletion.

However, Generative AI also creates risks. It may hallucinate explanations, generate incorrect remediation actions, expose sensitive data, misinterpret regulatory requirements, or produce outputs that users accept without sufficient verification. In banking, these risks cannot be ignored. Therefore, the central argument of this paper is that Generative AI should be integrated into data quality management as a controlled augmentation layer, supported by governance, access controls, human review, evidence-based retrieval, and auditability.

---

2. Problem Statement

Banks frequently have large volumes of data but limited confidence in its quality. A data element may exist in multiple systems with different definitions. A customer may have several identifiers across products. A transaction may be enriched differently in payment, finance, risk, and compliance systems. A data field used in regulatory reporting may pass through several transformations before appearing in a final report. When discrepancies occur, teams may spend significant time investigating which system is correct, which rule failed, who owns the issue, and what remediation is required.

The problem is not simply that data quality issues exist. The deeper problem is that many banking data environments lack a fully integrated lifecycle-based framework for preventing, detecting, explaining, remediating, and governing data quality issues across all stages of data use.

The author's blog provides a useful lifecycle structure by organizing the Data Quality Management Framework into six phases:

1. Data creation
2. Data storage
3. Data processing
4. Data usage
5. Data archival
6. Data deletion

For each phase, the blog identifies practical GenAI use cases and implementation steps. For example, during data creation, it proposes an intelligent data entry assistant to validate and clean input data in real time. During storage, it proposes automated data classification and GenAI-supported lineage documentation. During processing, it proposes AI-powered data cleansing and anomaly interpretation. During usage, it proposes automated report writing, policy analysis, and privacy scanning. During archival and deletion, it proposes retention planning, archive summarization, erasure orchestration, and deletion audit summarization.

The research problem can therefore be stated as follows:

How can banks design an end-to-end Data Quality Management Framework that covers the full data lifecycle while integrating Generative AI safely, effectively, and in alignment with governance and regulatory expectations?

---

3. Research Objectives

This paper has five objectives.

First, it formalizes the lifecycle-based data quality model introduced in the author's blog into a research-style banking framework.

Second, it defines how data quality controls should operate across creation, storage, processing, usage, archival, and deletion stages.

Third, it identifies practical Generative AI use cases at each lifecycle stage.

Fourth, it explains the governance and risk controls required to use GenAI safely in a banking data quality environment.

Fifth, it positions data quality management as an essential foundation for responsible AI adoption in banking.

---

4. Conceptual Foundation: Data Quality in Banking

Data quality refers to the degree to which data is fit for its intended purpose. In banking, purpose matters. A data element may be acceptable for marketing segmentation but insufficient for regulatory reporting. A customer address may be useful for communication but inadequate for legal residency determination. A transaction description may be readable by humans but too inconsistent for automated monitoring.

A banking data quality framework should consider at least the following dimensions.

4.1 Accuracy

Accuracy refers to whether data correctly represents the real-world object, transaction, customer, or event. Inaccurate customer names, incorrect balances, wrong risk ratings, or misclassified products can lead to operational errors and regulatory concerns.

4.2 Completeness

Completeness refers to whether all required fields are present. Missing KYC fields, missing transaction purpose, missing collateral values, or incomplete counterparty information can reduce the reliability of compliance and risk processes.

4.3 Consistency

Consistency refers to whether the same data is represented in the same way across systems. For example, if a customer is classified as high risk in one system and low risk in another, the bank must understand whether this is a valid business distinction or a data quality issue.

4.4 Timeliness

Timeliness refers to whether data is available and updated when required. A risk report based on stale exposures, an AML alert based on delayed transaction data, or a customer profile based on outdated KYC information can create control weaknesses.

4.5 Validity

Validity refers to whether data conforms to approved formats, value lists, and business rules. Examples include valid country codes, approved account statuses, correct date formats, and permissible product codes.

4.6 Uniqueness

Uniqueness refers to whether entities are represented without unnecessary duplication. Duplicate customer records can weaken customer due diligence, relationship-level exposure aggregation, and fraud detection.

4.7 Lineage and Traceability

In banking, data quality is incomplete without lineage. It is not enough to know that a value is wrong. The bank must know where it came from, how it was transformed, who owns it, where it is consumed, and what reports or processes are impacted.

---

5. Regulatory and Governance Context

Data quality in banking is not only an internal efficiency concern. It is a regulatory and risk management concern. Banks are expected to produce reliable data for risk management, financial reporting, regulatory reporting, customer protection, privacy compliance, and operational resilience.

A strong Data Quality Management Framework should therefore include:

- Data ownership
- Critical data element identification
- Data quality rules
- Data lineage
- Issue management
- Root-cause analysis
- Remediation ownership
- Risk acceptance
- Audit evidence
- Management reporting
- Regulatory traceability

Generative AI does not remove the need for these controls. In fact, it increases the need for them. If a bank uses GenAI to explain data quality issues, recommend remediation, generate

SQL, classify data, or summarize compliance requirements, the bank must be able to evidence how the AI output was produced, what data it used, who reviewed it, and what action was taken.

---

6. Proposed End-to-End Data Quality Management Framework

The proposed framework follows the six lifecycle phases from the author's blog and expands them into a banking-grade operating model:

1. Data creation
2. Data storage
3. Data processing
4. Data usage
5. Data archival
6. Data deletion

Across all six phases, the framework includes the following control principles:

- Define data ownership.
- Identify critical data elements.
- Establish business rules.
- Automate validation where possible.
- Capture metadata and lineage.
- Monitor exceptions.
- Assign remediation accountability.
- Maintain audit evidence.
- Use GenAI only with approved controls.
- Keep humans accountable for final decisions.

---

7. Phase 1: Data Creation

Data creation is the point at which data first enters the banking environment. This may occur through customer onboarding, account opening, loan origination, payment initiation, trade booking, relationship manager input, third-party feeds, mobile app forms, API submissions, or batch uploads.

Data quality issues introduced at this stage can propagate across the enterprise. For example, if an incorrect customer name is captured during onboarding, that error may flow into KYC systems, screening tools, account platforms, statements, regulatory reports, and analytics environments.

The author's blog identifies input validation as a key data creation activity and proposes an intelligent data entry assistant that uses an LLM to validate and clean entries in real time. The

example includes reviewing customer inputs such as name, date of birth, and address to identify errors and suggest corrections.

7.1 Key Controls at Data Creation

Data creation controls should include:

Control Area	Example
Mandatory field validation	Customer nationality cannot be blank
Format validation	Date of birth must follow approved date format
Domain validation	Country code must match approved ISO list
Duplicate detection	Customer ID or identity document already exists
Plausibility check	Date of birth cannot be in the future
Policy check	Only necessary data should be collected
Consent validation	Consent must be recorded for specific data use

7.2 GenAI Use Cases at Data Creation

Intelligent Data Entry Assistant

A GenAI assistant can help identify incomplete, inconsistent, or suspicious entries at the point of capture. For example, it can detect a date of birth such as “32/13/1980” as invalid, flag unusual characters in a name, or suggest standardization of an address.

Metadata and Minimal Capture Assistant

The author’s blog proposes a GenAI-based metadata and minimization checker that generates descriptions for new data fields and flags unnecessary data collection. The blog gives the example of questioning whether “Favorite Color” is necessary during account opening under a data minimization principle.

In banking, this is valuable because data collection must be justified. A bank should not collect personal data simply because it might be useful later. Every collected field should have a defined business, regulatory, risk, or operational purpose.

7.3 Governance Consideration

GenAI should not automatically approve or reject customer data without human or rule-based oversight. The safest design is for GenAI to provide recommendations, while deterministic validation rules and human review handle final acceptance for sensitive cases.

8. Phase 2: Data Storage

Once data is created, it must be stored securely, classified correctly, documented properly, and made discoverable through metadata and cataloging. Banking storage environments may include relational databases, mainframe systems, data warehouses, data lakes, cloud storage, document management systems, archival platforms, and analytics sandboxes.

The author's blog identifies two key storage-related activities: data classification and cataloging, and metadata and lineage management. It proposes an automated data classifier that reads schema information or sample data to classify fields such as personal data and financial data. It also proposes GenAI-assisted lineage documentation using ETL scripts, logs, and data flow diagrams.

8.1 Key Controls at Data Storage

Control Area	Example
Data classification	Personal, confidential, restricted, public
Encryption	Sensitive fields encrypted at rest
Access control	Role-based access to customer and account data
Metadata capture	Business definition, owner, steward, system of record
Lineage capture	Source-to-target flow documented
Retention tagging	Data tagged with retention period
Catalog registration	Critical datasets discoverable in approved catalog

8.2 GenAI Use Cases at Data Storage

Automated Data Classifier

GenAI can assist by reviewing column names, sample values, and glossary definitions to classify fields. For example, fields such as `Name`, `Account_No`, and `Balance` may be classified as personal data, account identifier, and financial data respectively. The author's blog proposes this exact type of classification and cataloging workflow.

Lineage Documentation Assistant

GenAI can analyze ETL scripts, transformation logs, and data flow documents to draft lineage descriptions. For example, it can explain how `Risk_Exposure` originates in a source system, moves through intermediate transformations, and appears in a final report. The

author’s blog identifies this as a practical use case for GenAI in metadata and lineage management.

8.3 Governance Consideration

GenAI-generated classifications and lineage descriptions should be treated as draft outputs. Data stewards and system owners must review and approve them before they become official metadata records.

---

9. Phase 3: Data Processing

Data processing includes cleansing, transformation, enrichment, standardization, reconciliation, aggregation, and integration. In banking, this phase is critical because data often moves across systems before being used in reports, risk engines, compliance tools, or customer services.

The author’s blog identifies two major processing activities: data cleansing and standardization, and anomaly detection and reconciliation. It proposes an AI-powered data cleaner that suggests deduplication, missing-value handling, and format fixes. It also proposes an intelligent anomaly detector that interprets data profiles or reconciliation reports and identifies likely causes.

9.1 Key Controls at Data Processing

Control Area	Example
Standardization	Address, name, date, and code format standardization
Deduplication	Duplicate customer or account records removed
Reconciliation	Source and target totals matched
Transformation validation	Mapping rules checked
Exception thresholding	Breaches flagged when tolerance exceeded
Outlier detection	Unusual transaction or balance values reviewed
Processing completeness	All expected files and records received

9.2 GenAI Use Cases at Data Processing

AI-Powered Data Cleaner

GenAI can suggest cleaning logic or code for inconsistent data. For example, it can recommend steps to standardize address formats, remove duplicates, and handle missing values. The blog proposes using GenAI to generate data cleaning actions or scripts, with execution first in a test environment and human review before pipeline integration.

Intelligent Anomaly Interpreter

GenAI can interpret profiling results. For example, if a transaction amount field contains negative values, the AI can identify this as an anomaly and suggest possible causes. The blog provides a similar example involving negative transaction amounts and asks the AI to identify anomalies and likely causes.

9.3 Governance Consideration

GenAI-generated code should not be executed directly in production. It should go through software development lifecycle controls, testing, peer review, data owner sign-off, and change management.

---

10. Phase 4: Data Usage

Data usage includes reporting, analytics, dashboards, regulatory submissions, customer communications, risk analysis, AI model training, compliance monitoring, and business decision-making. At this stage, poor data quality becomes visible to end users and decision makers.

The author's blog identifies three major usage-phase GenAI applications: automated report writing, policy and contract analysis, and sensitive data leakage scanning.

10.1 Key Controls at Data Usage

Control Area	Example
Report validation	Reconcile report numbers to source systems
Usage approval	Confirm data is permitted for intended purpose
Privacy review	Ensure no unauthorized PII exposure
Business sign-off	Data owner approves material outputs
Model input validation	AI/ML datasets checked before training
Access monitoring	Sensitive data access tracked
Disclosure control	External reports reviewed before release

10.2 GenAI Use Cases at Data Usage

Automated Report Writer

GenAI can convert metrics into narrative explanations. The author's blog proposes using GenAI to draft executive summaries, explanations, and slide content from raw data and analytics. It gives an example of drafting a quarterly risk summary from credit and market risk data.

In banking, this could support risk packs, data quality dashboards, regulatory issue updates, and management reporting. However, GenAI-generated narratives must be reviewed by accountable analysts.

Policy and Contract Analyst

The blog proposes a GenAI copilot that cross-references usage scenarios with internal policies or legal documents. For example, if a bank plans to share transaction data with a fintech, the AI can retrieve relevant policy clauses and summarize requirements such as anonymization, consent, and deletion obligations.

This is especially useful when combined with retrieval-augmented generation over approved policy repositories.

Output Privacy Scanner

The author's blog also proposes using GenAI to scan reports or datasets for personal or sensitive information before publication. This can help prevent accidental exposure of customer names, account numbers, addresses, or confidential financial information.

10.3 Governance Consideration

GenAI should not be the final authority on compliance. Legal, compliance, data protection, and business teams must review high-risk outputs.

---

11. Phase 5: Data Archival

Data archival ensures that data is retained appropriately, moved to lower-cost or controlled storage when no longer actively used, and preserved for legal, regulatory, audit, or business purposes. In banking, archival decisions must consider retention schedules, legal holds, customer privacy obligations, regulatory requirements, and operational recoverability.

The author's blog identifies retention compliance and archive documentation as key archival activities. It proposes a retention policy assistant that translates retention rules into an actionable archiving or deletion list. It also proposes an archive summarizer that generates documentation describing archived datasets.

11.1 Key Controls at Data Archival

Control Area	Example
Retention classification	Data mapped to approved retention period
Legal hold check	Data not archived or deleted if under hold
Archive completeness	All required records archived
Archive metadata	Contents, period, owner, sensitivity recorded
Access restriction	Archived sensitive data protected
Retrieval testing	Data can be restored if needed
Disposal trigger	Expiry dates monitored

11.2 GenAI Use Cases at Data Archival

Retention Policy Assistant

GenAI can interpret retention schedules and data inventory summaries to identify what should be archived or deleted. The blog gives an example of user activity logs retained for one year and asks what should happen to older logs.

Archive Summarizer

GenAI can summarize archived datasets by describing their content, date range, volume, and important fields. The author’s blog proposes storing such summaries in an archive register or metadata record.

11.3 Governance Consideration

Retention decisions should be policy-driven. GenAI may assist interpretation, but legal, records management, compliance, and data owners must define and approve retention rules.

---

12. Phase 6: Data Deletion

Data deletion is the controlled removal of data when it is no longer required, when retention has expired, or when a valid customer erasure request applies. In banking, deletion is complex because data may exist across production systems, archives, backups, analytics platforms, reports, logs, and third-party environments.

The author's blog identifies two deletion-phase activities: right-to-erasure request handling and deletion verification/audit. It proposes an erasure request orchestrator that interprets customer deletion requests, identifies relevant systems, and drafts confirmation. It also proposes a deletion audit summarizer that reviews deletion logs and produces a human-readable verification report.

12.1 Key Controls at Data Deletion

Control Area	Example
Request validation	Confirm customer identity and request scope
Legal basis check	Determine whether deletion is permitted
System mapping	Identify all systems containing the data
Deletion execution	Remove or anonymize approved records
Exception handling	Identify data that cannot be deleted immediately
Audit evidence	Store logs and approvals
Customer response	Provide compliant confirmation

12.2 GenAI Use Cases at Data Deletion

Erasure Request Orchestrator

GenAI can extract key information from customer requests, identify likely systems containing customer data using the bank's data map, and draft a deletion plan. The blog gives an example involving a customer requesting deletion of all personal data and asking the AI to identify systems where the data may reside.

Deletion Audit Summarizer

After deletion jobs run, GenAI can summarize logs into a human-readable audit report. The blog proposes using AI to summarize whether records were removed, whether exceptions remain, and what evidence should be attached to the compliance file.

12.3 Governance Consideration

Deletion is legally sensitive. GenAI should support workflow orchestration and documentation but should not independently approve deletion or override retention obligations.

---

13. Cross-Lifecycle GenAI Architecture

A safe GenAI-enabled DQMF architecture should include the following components:

1. Banking source systems
2. Data ingestion and integration layer
3. Data quality rule engine
4. Metadata and lineage repository
5. Data catalog
6. Data issue management platform
7. Policy and regulatory document repository
8. Retrieval-augmented generation layer
9. Enterprise-approved LLM
10. Guardrail and validation layer
11. Human approval workflow
12. Audit logging and evidence store

13.1 Reference Architecture

```
Banking Systems
   ↓
Data Ingestion and Processing
   ↓
Data Quality Rules and Profiling
   ↓
Exceptions, Metadata, Lineage, and Catalog
   ↓
Retrieval-Augmented Generation Layer
   ↓
GenAI Assistant
   ↓
Human Review / Steward Approval
   ↓
Remediation Workflow
   ↓
Audit Evidence and Governance Reporting
```

The key architectural principle is that GenAI should be grounded in approved enterprise sources. It should retrieve from governed metadata, lineage, data quality results, issue logs, policies, standards, and approved documentation. It should not fabricate definitions, ownership, rules, or regulatory interpretations.

---

14. Operating Model

The framework requires a clear operating model.

Role	Responsibility
Data Owner	Accountable for data quality in a business domain
Data Steward	Manages operational data quality issues
Data Custodian	Maintains technical systems and controls
Control Owner	Defines and monitors specific DQ controls
Compliance Team	Reviews regulatory and policy implications
Data Protection Officer	Oversees privacy and deletion obligations
AI Governance Team	Reviews GenAI use cases and controls
Model Risk Team	Reviews AI model risk where applicable
Internal Audit	Provides independent assurance

A key principle is that data quality cannot be owned only by technology. Technology teams can build controls and platforms, but business teams must define meaning, quality expectations, severity, and acceptable risk.

---

15. Risk Management for GenAI Integration

15.1 Hallucination Risk

GenAI may produce plausible but incorrect explanations. In data quality management, this could result in a wrong root cause, incorrect remediation, or misleading compliance interpretation.

Mitigation: Use retrieval-augmented generation, display source references, require human approval, and prohibit unsupported answers.

15.2 Data Leakage Risk

Banking data may include personal, financial, confidential, or regulated information.

Mitigation: Use enterprise-approved AI platforms, apply role-based access control, mask sensitive data, and prevent confidential data from being sent to public models.

15.3 Incorrect Code Generation Risk

GenAI may generate SQL or Python code that changes data incorrectly.

Mitigation: Restrict GenAI-generated code to test environments, require peer review, run automated tests, and use formal change management.

15.4 Compliance Misinterpretation Risk

GenAI may misread policy or legal requirements.

Mitigation: Use approved policy repositories, require compliance review, and maintain evidence of retrieved clauses.

15.5 Overreliance Risk

Users may accept AI recommendations without verification.

Mitigation: Provide training, require maker-checker controls, and clearly label AI outputs as recommendations.

---

16. Implementation Roadmap

Phase 1: Establish Data Quality Foundation

- Identify priority banking domains.
- Define critical data elements.
- Assign owners and stewards.
- Document business definitions.
- Establish baseline data quality metrics.

Phase 2: Implement Lifecycle Controls

- Add controls at data creation.
- Classify and catalog stored data.
- Define processing and reconciliation checks.
- Validate data usage and reporting.
- Define retention and deletion workflows.

Phase 3: Build Governance and Issue Management

- Establish data quality forums.
- Track exceptions and remediation.
- Measure issue ageing and recurrence.
- Report quality trends to management.

Phase 4: Pilot GenAI Use Cases

Begin with low-risk, read-only use cases:

- Data quality issue summarization.
- Metadata description drafting.
- Lineage explanation.
- Report narrative drafting.
- Policy retrieval assistance.

Phase 5: Scale with Guardrails

- Expand GenAI to more domains.
- Add workflow integration.
- Add audit logging.
- Monitor AI accuracy and hallucination.
- Integrate with AI governance and model risk processes.

---

17. Evaluation Metrics

17.1 Data Quality Metrics

Metric	Purpose
Completeness rate	Measures missing required data
Accuracy exception rate	Measures incorrect values
Duplicate rate	Measures uniqueness issues
Timeliness breach rate	Measures stale data
Reconciliation break count	Measures cross-system mismatch
CDE rule coverage	Measures control coverage

17.2 Operational Metrics

Metric	Purpose
Mean time to detect	How quickly issues are identified
Mean time to assign	How quickly ownership is established

Metric	Purpose
Mean time to remediate	How quickly issues are resolved
Recurrence rate	Whether root causes are fixed
Manual effort saved	Efficiency improvement

17.3 GenAI Metrics

Metric	Purpose
Response accuracy	Correctness of AI explanation
Citation coverage	Whether AI outputs are evidence-backed
Hallucination rate	Unsupported claims generated
Human acceptance rate	Percentage of suggestions accepted
Review correction rate	Frequency of human edits
Time saved	Efficiency gain in analysis or reporting

---

18. Discussion

The proposed framework extends the author's original blog from a practical lifecycle guide into a formal banking data quality architecture. The blog's strength is that it does not treat data quality as a single activity. Instead, it recognizes that data quality must be managed throughout the lifecycle: when data is created, stored, processed, used, archived, and deleted.

This lifecycle framing is especially valuable for banking because many data failures occur upstream but become visible downstream. A missing field during onboarding may become a compliance issue months later. A weak classification process during storage may become a privacy risk during reporting. A poorly documented transformation may become a regulatory lineage issue during audit. A weak deletion workflow may become a data protection breach.

Generative AI can add value because it improves interpretation and productivity. It can help humans understand issues faster, generate draft documentation, interpret policies, summarize logs, and explain technical flows in plain language. However, the technology must remain within a controlled operating model. The highest-value design is not "AI replaces data governance." The better design is "AI assists governed data quality workflows."

This distinction is crucial. In banking, accountability cannot be delegated to an AI model. Data owners, control owners, compliance teams, and technology custodians remain

accountable for decisions. GenAI should support them with faster analysis and better documentation.

---

19. Limitations

This framework is conceptual and implementation-oriented. It is not based on empirical measurement from a live banking deployment. Actual adoption will depend on the bank's architecture maturity, data governance capability, metadata quality, cloud strategy, regulatory jurisdiction, privacy obligations, and AI risk appetite.

The framework also assumes that the bank has access to approved metadata repositories, lineage tools, policy repositories, issue management systems, and secure GenAI platforms. Institutions with fragmented systems may need foundational modernization before implementing the full model.

Finally, GenAI technology continues to evolve. The governance model must therefore be periodically reviewed to reflect changes in regulation, model capability, cyber risk, privacy expectations, and operational resilience requirements.

---

20. Conclusion

This paper presented a research-style framework for end-to-end Data Quality Management in banking systems with Generative AI integration. Building on the author's original blog, the framework organizes data quality across six lifecycle phases: data creation, storage, processing, usage, archival, and deletion. Across each phase, the paper identified practical controls, GenAI use cases, governance considerations, and risk mitigations.

The central conclusion is that GenAI can significantly enhance data quality management, but only when embedded into a disciplined governance framework. It can help validate inputs, generate metadata, classify data, document lineage, interpret anomalies, draft reports, analyze policies, detect privacy leakage, support retention planning, summarize archives, orchestrate erasure requests, and produce deletion audit summaries. These capabilities can reduce manual effort and improve data stewardship productivity.

However, GenAI should not be treated as an autonomous authority. In banking, data quality decisions must remain explainable, auditable, policy-aligned, and accountable. The safest and most effective model is a human-in-the-loop approach where GenAI provides evidence-based assistance through approved enterprise data sources.

Trusted data is now both a regulatory requirement and an AI-readiness requirement. Banks that strengthen data quality management across the full lifecycle will be better positioned to improve compliance, reduce operational risk, accelerate analytics, and adopt Generative AI responsibly.

References

Debabrata Pruseth. (2025). *End-to-End Data Quality Management Framework (DQMF) in Banking with GenAI Integration*. Blog article.

Basel Committee on Banking Supervision. (2013). *Principles for effective risk data aggregation and risk reporting*. Bank for International Settlements.

DAMA International. (2017). *DAMA-DMBOK: Data Management Body of Knowledge*.

Suggested Citation

Pruseth, D. (2025). *An End-to-End Data Quality Management Framework for Banking Systems with Generative AI Integration*.