International Journal of Emerging Trends in Computer Science and Information Technology
ISSN: 3050-9246 | https://doi.org/10.56472/ICCSAIML25-136
Eureka Vision Publication | ICCSAIML'25-Conference Proceeding

Original Article

Preventive Data Quality Enforcement at the Source: A Shift-Left Approach for FinTech and HealthTech

Vamsi Kunaparaju
DMC (Data Management Company) - Virginia, USA.

Abstract: *Data quality problems in enterprise datasets lead to significant costs, risks, and inefficiencies. This paper proposes a shift-left approach to data quality, enforcing validation and cleansing at the point of data entry (the source) rather than downstream. By preventing erroneous or incomplete data from entering systems, organizations – particularly in financial technology (FinTech) and healthcare technology (HealthTech) sectors – can reduce costly downstream cleaning, comply with stringent regulations, and improve the reliability of analytics. We present a framework for preventive data quality enforcement, discuss supporting tools and technologies, and model the return on investment (ROI) of early data quality interventions. We also make an argument to how early prevention, and the proposed Shift Left mechanism, can help organizational practitioners to save as much as 60% of their resources on time and monetary budget, helping them drive superior outcomes in both operational efficiency and strategic decision-making.*

Keywords*: Compliance, Data Quality, Data Governance, FinTech, HealthTech, Preventive Validation, Shift Left, Source Data Control, Data Management, Data Stewardship, Metadata Management.*

1. Introduction

High-quality data is the cornerstone of reliable information systems and analytics. Conversely, poor data quality has been estimated to cost businesses enormously – one analysis pegged the annual cost of bad data in the U.S. at around $3.1 trillion (T. C., 2016). In practice, data scientists and analysts often spend a majority of their time (up to 80%) in data cleansing and preparation rather than deriving insights [2]. These findings underscore that organizations pay a steep price for data errors, inconsistencies, and omissions. The impact is especially pronounced in data-intensive domains like finance and healthcare, where decisions based on flawed data can lead to monetary loss, compliance penalties, or even endanger lives.

Data defects uncovered in production incur remediation costs 5 to 10x higher than if caught at source [2]. Traditional reactive monitoring leaves analytics, ML models, and dashboards vulnerable to “surprise” issues [2]. By shifting left– validating at ingestion and during development– organizations can restore stakeholder trust, accelerate analytics delivery, and embed compliance checks (GDPR, HIPAA) seamlessly [4],[8].

In the FinTech sector, regulatory bodies emphasize robust data management and accuracy. For example, the Basel Committee’s BCBS 239 principles mandate banks to have comprehensive data governance and high-quality risk data aggregation[3]. Financial institutions with poor data controls have faced regulatory sanctions and operational mishaps when critical risk or transaction data were wrong or late. Similarly, in HealthTech, the integrity of electronic health records (EHRs) and medical data is paramount. Studies have shown that incomplete or inaccurate clinical data can adversely affect patient care and impede medical research. Ensuring correctness at the point of capture – whether it be a patient’s demographic information or a medical device reading – is vital to avoid downstream errors in treatment or analysis[6].

Traditionally, organizations address data quality in a reactive manner: errors are identified and corrected in batch processes, data cleaning projects, or during report generation. This approach means that data flaws persist in databases and propagate through systems until detected, causing rework and potentially faulty decisions in the interim. In contrast, a *shift-left* philosophy, borrowed from software engineering, advocates moving quality assurance earlier in the life cycle[5]. In software testing, catching a defect during development is far cheaper and easier than fixing it after deployment. By analogy, identifying and resolving data issues at the moment of data creation or entry yields similar cost savings and risk reduction. Empirical observations in software indicate that the later a problem is found, the more expensive it is to fix[8]. We apply this principle to data: preventing a faulty data point from ever entering the system is far more effective than scrubbing it after it has cascaded through multiple applications.

This paper presents a preventive data quality enforcement framework that operationalizes the shift-left concept for data management in FinTech and HealthTech environments. We outline how data validation rules, governance processes, and toolchains can be integrated at the data source to ensure accuracy and completeness from the start. In the following sections, we discuss background and related work on data quality and shift-left principles (Section **Background**), introduce the proposed framework (Section **Framework**), provide illustrative examples in financial and healthcare contexts (Section **Examples**), review enabling tools and technologies (Section **Tools**), analyze the ROI of early data quality intervention (Section **ROI**), and describe practical considerations for implementation (Section **Implementation**). Finally, we conclude with a summary of findings and a business-oriented perspective on the benefits of shifting data quality efforts to the source.

2. Background

2.1. Data Quality Challenges:

Data quality is commonly defined along dimensions such as accuracy, completeness, consistency, timeliness, and validity. High-quality data accurately represents the real-world entities it is intended to model, is complete with no missing or superfluous elements, is consistent across different systems, is available at the time needed, and conforms to required formats and business rules. In both FinTech and HealthTech, maintaining these quality dimensions is challenging. Financial data often arrives from disparate sources (transactions, market feeds, customer inputs) and can suffer from inconsistencies or format errors during integration. Healthcare data, originating from hospital information systems, lab results, wearables, and manual entry, frequently contains inconsistencies (e.g. differing units or codes for the same measure) and omissions (e.g. missing patient history details).

The consequences of poor data in these domains are severe. In finance, erroneous data can lead to incorrect risk assessments or regulatory reporting errors, with potentially hefty fines and reputational damage. A well-known industry survey reported that 75% of financial enterprises identified data quality issues as a key barrier to effective risk management. In healthcare, data quality problems such as duplicate patient records or mis-coded diagnoses can result in treatment mistakes or research biases. Weiskopf and Weng (2013) document various deficiencies in clinical data repositories that limit their secondary use for research, highlighting issues like incompleteness and lack of standardization [4]. In finance, as mentioned, BCBS 239 and related data governance regulations insist on "right first time" data aggregation for risk reporting [3].

In healthcare, data standards like HL7 Fast Healthcare Interoperability Resources (FHIR) include rigorous definitions for data fields which, when implemented, act as upfront validators (e.g. only valid coded values are accepted for a given medical condition field). The industry trend is acknowledging that ensuring data quality at the source reduces the burden and risk later in the pipeline. In summary, prior work and industry best practices indicate that shifting data quality controls upstream can address the Achilles' heel of many analytics and AI projects – namely, the garbage-in-garbage-out problem. The next section introduces a structured framework to realize preventive data quality enforcement at the source, drawing on these background insights.

3. Framework

The proposed **Preventive Data Quality Enforcement Framework** is a comprehensive methodology to embed quality assurance into the data creation and ingestion processes. It consists of multiple components and steps that together ensure data is validated and cleansed before it propagates downstream. The key elements of the framework are described below:

3.1. Early Data Quality Requirements Definition:

Effective prevention starts with clearly defining what constitutes high-quality data for a given context. In this initial step, data stakeholders (business analysts, data stewards, domain experts) collaboratively specify quality rules and standards for each data element at the requirements or design phase of a project. For FinTech applications, this might mean delineating rules such as: Account number must be 10 digits, Transaction dates cannot be in the future, Currency codes must follow ISO 4217. For HealthTech, examples could include: Blood pressure readings must fall within human physiological ranges, Mandatory fields like patient ID and physician signature cannot be blank, Diagnosis codes must be valid ICD-10 codes. These requirements should be documented alongside functional requirements so that they are integral to system design. By shifting this definition step to the beginning, we treat data quality as a fundamental requirement, not an afterthought.

3.2. Built-in Validation at Source:

Once rules are defined, the next step is implementing them wherever data is entered or ingested. This involves adding validation logic to user interfaces, devices, and data import processes:

- **User Interface Validation:** All front-end forms and data entry UIs enforce format and business rules in real-time. For example, a fintech mobile app for loan applications can check that numeric fields (income, loan amount) contain valid numbers within expected ranges, and that mandatory fields (such as Social Security Number or Tax ID) are present and

properly formatted. If a user attempts to submit inconsistent data – say an applicant's birthdate that implies an age below 18 for a loan product restricted to adults – the form should refuse submission and display a clear error message. In a hospital's EHR system, as data is entered for a new patient intake, the software can require that all critical fields (allergies, current medications, etc.) are filled and that values make sense (e.g. an impossible dosage or a non-existent drug name is flagged immediately).

- **Device and Sensor Data Checks:** In scenarios where data originates from machines (IoT sensors, medical devices, ATMs, etc.), the ingestion layer should include automated validation. For instance, a vital-signs monitor streaming patient data could have software filters that discard or double-check any reading that deviates wildly from the patient's recent trend or biological plausibility (to catch sensor malfunctions). A financial trading platform receiving market data from an exchange can implement range checks and consistency checks (e.g. ensuring a stock price feed is within plausible bounds and time-sequenced correctly) before that data is used in algorithms.
- **APIs and Data Integration Points:** Many systems consume data via APIs or batch file feeds from external sources. The framework dictates that the receiving endpoint should validate all incoming data against the quality rules. If a partner system sends over a batch of transaction records, each record is programmatically verified – for schema compliance, allowed values, referential integrity (e.g. does the referenced customer ID exist?), etc. Non-conforming records can be rejected or quarantined for review rather than blindly inserted into the database.

3.3. Immediate Feedback and Correction:

A crucial aspect of enforcement at the source is providing feedback at the moment of data entry so that errors can be corrected by the originator. When a validation rule is violated, the system responds with a clear message indicating what is wrong, enabling the user (or sending system) to fix it right away. For example, if a clinician tries to input an observation with an unspecified unit (say, a lab result value without indicating mg/dL or mmol/L), the system should prompt: "Unit of measure is required for this value." In a web form, this is near-instant feedback highlighting the problematic field. For automated feeds, feedback might take the form of an error report or API response detailing which records failed validation and why. The goal is to intercept bad data and have it corrected by the source as part of the normal workflow. This minimizes the need for later cleanup – the person or system providing the data is made responsible for fixing it before proceeding.

3.4. Data Quality Metadata and Logging:

The framework includes capturing metadata about data quality at entry. Each time a rule is applied, especially if a record is corrected or rejected, the event is logged. This might include what rule was triggered, what the offending value was, and when/where it occurred. Such metadata is invaluable for auditing and analysis. Over time, the logs can reveal patterns – for instance, a particular field might see frequent entry errors during night shifts, suggesting a need for better training or interface improvement. In FinTech, if numerous transactions from a particular branch fail initial checks, that could indicate an upstream system issue at that branch. By logging quality incidents at source, organizations can perform root cause analysis and continuously improve the process (closing the loop to Step 1 by refining requirements or providing additional user guidance).

3.5. Integration with Development and DataOps:

To truly shift-left, data quality enforcement must be part of the development lifecycle. This means developers include data validation rules in their unit and integration tests, just as they would test business logic. Modern DataOps pipelines can incorporate data quality tests as a step whenever data pipelines are run or when new data schemas are deployed. For example, before a new data ingestion job goes live, it is tested with sample data to ensure all defined quality checks work correctly. In continuous integration (CI) systems, one could run automated suites to verify that any code dealing with data input adheres to the quality rules (e.g. ensuring a new API endpoint doesn't accept invalid data that violates constraints). This practice treats data quality rules as code, subject to version control and automated testing, thereby embedding them deeply into the system development process.

3.6. Governance and Stewardship at Entry:

Beyond technical measures, the framework stresses the role of data governance. Assigning *data stewards* or owners for key data domains ensures there is accountability for data quality from the start. For instance, an organization might designate a finance data steward who is responsible for overseeing the quality of financial data captured in core systems. When new data fields are introduced, the steward helps define the validation requirements (Step 1) and monitors the effectiveness of controls. Likewise, in a healthcare context, a data governance committee may set policies that all new digital patient forms undergo a data quality review before launch. These governance structures support the shift-left framework by aligning organizational responsibilities with the point of data origin – effectively pushing the caretaking of data quality to those closest to the data creation.

3.7. Continuous Monitoring and Improvement:

Finally, the framework is iterative. Even with strong preventive measures, some data issues will slip through or new types of issues will emerge. Therefore, continuous monitoring of data quality metrics is necessary. This can include tracking the rate of data entry errors over time, measuring completeness scores for incoming records, or sampling data to audit correctness. If monitoring reveals a recurring problem (e.g. a certain field frequently corrected by end-users, indicating initial capture issues), the framework advocates a review and improvement cycle: update the validation rules or interface design, provide additional training, or adjust the process as needed. In essence, the system "learns" where its quality enforcement might be lacking and strengthens those points proactively. Collectively, these steps form a loop of preventive control and feedback. By implementing this framework, an organization builds a data supply chain that has quality checkpoints from the very start, analogous to adding quality gates on an assembly line right at the first station. The next section will illustrate how this framework operates in practice with examples from FinTech and HealthTech use cases.

4. Tools

Implementing preventive data quality enforcement requires a combination of software tools and platforms. These tools span from data validation libraries embedded in applications to enterprise data quality management suites. Below, we outline several categories of tools and examples of each, relevant to FinTech and HealthTech contexts:

4.1. Application-Embedded Validation Libraries:

Modern software development offers many libraries to enforce data schemas and rules within applications. For example, *JSON/XML schema validators* can ensure that API inputs conform to an expected schema (all required fields present, data types correct). For more complex rules, open-source libraries like **Great Expectations** or **Cerberus** (for Python) allow developers to declaratively specify data expectations (e.g. ranges, regex patterns, cross-field dependencies) and then validate datasets or streaming data against these expectations [10]. Similarly, in the big data realm, tools like **Deequ** (an open-source data quality library by Amazon) can be integrated into data processing jobs to automatically check data quality constraints on large datasets as they are processed [11] [12]. These libraries are essential building blocks for embedding quality checks at source, as they can be invoked wherever data flows (in microservices, ETL jobs, real-time streaming applications, etc.).

4.2. Enterprise Data Quality Platforms:

There are comprehensive tools designed for enterprise-wide data quality management which can also be leveraged for source enforcement. Examples include **Informatica Data Quality**, **IBM InfoSphere QualityStage**, **SAP Information Steward**, and **Oracle Enterprise Data Quality**. These platforms typically provide a suite of functionalities: data profiling (to discover quality issues), rule design interfaces (often with a GUI to author validation rules and transformations), and execution engines that can run these rules either in batch or real-time [13]. In a shift-left approach, these tools can be configured at integration points to intercept data. For instance, IBM's QualityStage could be set up on an incoming data feed to standardize and validate data before it is accepted into a data lake. These platforms also often come with dashboards and scorecards for monitoring data quality metrics, which supports the continuous monitoring aspect of the framework.

4.3. Master Data Management (MDM) Systems:

MDM solutions (like **Informatica MDM**, **IBM Master Data Connect**, or **Talend MDM**) ensure that key business entities (customers, products, etc.) have a single, consistent, and accurate record across the organization. While MDM is a broad discipline, its role in preventive quality is significant: by having a central master record with validation and de-duplication rules, whenever new data for a master entity is created or updated (at a source system), it can be cross-checked against the master standards [13]. For example, if a bank's branch office tries to enter a new customer that is already registered in another region, the MDM system can catch the duplicate in real-time and prompt to link to the existing master record instead, thus preventing a duplicate entry. In healthcare, MDM helps with patient identity management; when a new patient is admitted, an MDM-powered check can detect if that patient already exists under a slightly different name or ID, thereby preventing duplicate patient records that plague many hospital systems.

4.4. Data Governance and Catalog Tools:

Tools like **Collibra** Data Governance, **Alation** Data Catalog, or **Ataccama** provide data cataloging along with policy management. These can store the defined data quality rules and policies centrally and even enforce them via integrations. For example, Collibra can maintain a business glossary and data quality rules for each data element; when integrated with workflow, it can ensure that any new data source or field goes through an approval process including data quality checks. Some governance tools have workflow engines to route data quality issues back to data owners when problems are detected. While governance tools might not enforce quality by themselves, they play a coordinating role – linking the definitions (metadata) to the enforcement

points and to the people responsible.

4.5. Realtime Data Monitoring and Anomaly Detection:

In fast-moving data environments common in FinTech (like high-frequency trading, fraud detection streams) and HealthTech (real-time patient monitoring), specialized tools are used to monitor data in motion. Platforms such as **Apache Kafka** with stream processing frameworks (Flink, Spark Streaming) can incorporate anomaly detection algorithms that flag outliers or unusual patterns on the fly [14]. For instance, a stream processing job might watch transaction amounts per account and flag a sudden spike as a potential error or fraud – prompting a verification of those transactions before they are fully committed. Similarly, an ICU patient monitoring system might run a real-time check that if all sensors flatline simultaneously, it could be a technical fault rather than all physiological signals dropping – prompting a quick data validity check. These real-time frameworks complement the rule-based validation by catching complex patterns that rules might not enumerate explicitly.

4.6. Standards and Interoperability Frameworks:

Adhering to industry data standards is a form of preventive quality measure, since standards come with built-in expectations for data format and meaning. Tools that enforce standards can be considered part of the toolkit [15]. In FinTech, for example, messaging standards like **ISO 20022** for payments ensure that all necessary information in a payment message is present and structured; financial institutions use validator tools to check compliance with such standards at entry. In HealthTech, **HL7 FHIR** implementation tools ensure that healthcare messages or resources (for a patient, observation, etc.) conform to the standard schemas and code sets. By using these standards enforcement tools, organizations effectively prevent non-conforming data from entering their systems.

4.7. Database Constraints and Triggers:

It should not be overlooked that traditional database management systems offer built-in mechanisms for data quality control at source. Primary key, foreign key, and check constraints in SQL databases, for instance, are straightforward yet powerful ways to prevent invalid data from being stored (e.g. a CHECK constraint that a percentage field must be between 0 and 100, or a foreign key ensuring a referenced record exists). Triggers can also be used to implement custom validations or transformations upon insert/update. While these are low-level tools, they are often the last line of defense if higher-level application checks somehow miss an issue. Organizations should judiciously use DB constraints to encode critical quality rules – this ensures that even ad-hoc data loads or direct DB edits cannot violate core data integrity [16].

In implementing a shift-left data quality framework, the selection of tools will depend on the existing technology stack and specific needs. Many organizations adopt a hybrid approach: using application-level validations for real-time checks, enterprise platforms for broader governance and cleansing tasks, and standard DB or streaming tools for low-latency enforcement. The key is that these tools are employed *as early as possible* in the data flow. For FinTech companies, performance and scalability of these tools are important (validating high volumes with minimal latency). For HealthTech, integration with clinical workflows and standards compliance is crucial (tools must fit seamlessly into healthcare data environments). Fortunately, the trend in tooling aligns well with the shift-left philosophy: newer data engineering frameworks emphasize testing data (not just code) and providing immediate feedback on data quality. By leveraging the right combination of these technologies, organizations can automate much of the preventive data quality enforcement and make it an intrinsic part of their data infrastructure.

5. ROI

Investing in preventive data quality measures at the source requires resources and effort – so it is natural to ask: what is the return on investment? The ROI of a shift-left data quality approach can be understood by comparing the costs of implementing and running these preventive measures against the savings and gains achieved by avoiding poor data downstream. We break down the ROI considerations into tangible and intangible benefits, supported by analytical reasoning and industry observations.

5.1. Cost Reduction from Error Prevention:

Perhaps the most direct ROI component is the reduction in costs associated with detecting and fixing data issues. Numerous studies and industry experiences confirm that the later a defect is found, the more expensive it is to correct[8]. We can frame a simple quantitative calculation: Imagine you have a million pieces of information to check, and without any safeguards about 5 out of every 100 are wrong. Fixing each mistake might take about $100 worth of work, so on average you spend $5 per piece just cleaning up errors. Now, if you add a small check right at the start that costs just $1 per piece and catches most mistakes early, so only 1 out of every 100 slips through, you'll spend $1 to check and another $1 to fix the few that remain, for a total of $2 per piece. That's a drop from $5 down to $2, saving you $3 on each one. Over a million pieces of data, that adds up to three million dollars you don't have to spend fixing mistakes later [7].

5.2. Improved Decision Outcomes and Risk Mitigation:

Not all benefits are measured in direct cost. In FinTech, one of the biggest ROI factors is risk mitigation. Poor data quality can lead to compliance violations or financial losses that dwarf operational costs. For example, an error in risk data aggregation might cause a bank to mis-report its capital adequacy, potentially leading to regulatory fines or reputational damage. By enforcing data accuracy and completeness at source, the bank greatly reduces such risk. While "avoiding a potential fine" is hard to quantify, it is part of the ROI narrative. FinTech firms also face fraud risks; catching an anomaly in input data (like mismatched account details) could prevent a fraud attempt – saving the company from a direct loss. In HealthTech, preventing data errors can literally save lives or avoid patient harm – which from a business perspective also means avoiding malpractice lawsuits and the costs of patient safety incidents. Moreover, reliable data leads to better decisions: a hospital that trusts its data may confidently implement data-driven improvements (like analyzing treatment outcomes), whereas if data is suspect, opportunities may be missed. These risk avoidance and improved decision-making outcomes contribute to ROI by safeguarding and enhancing revenue. For instance, a healthcare provider with high-quality data might achieve better patient outcomes (which could tie to value-based care reimbursements) and higher patient trust, indirectly benefiting financially.

5.3. Validating ROI with Metrics:

To ensure that the anticipated ROI is realized, organizations should establish metrics after implementing preventive data quality measures. Relevant metrics include: reduction in number of data errors detected downstream (e.g. data quality issue tickets per month), time spent on data cleaning tasks before vs. after, number of incidents attributable to data errors, and improvements in data quality scores or index if one is used (some companies use a composite data quality index to track overall health of data). Many companies also perform before-and-after studies on specific processes. For example, a bank might measure the effort to produce risk reports prior to implementing source data controls and then measure it a few cycles after implementation; if effort drops and report accuracy issues disappear, that is tangible ROI evidence. In summary, while implementing shift-left data quality has a cost (licensing tools, development effort to add validations, training staff, etc.), the payoff comes in multiple forms: lower operational costs for data correction, avoidance of costly errors and compliance issues, improved efficiency, and enhanced ability to leverage data for value. When making a business case, organizations often find that even a moderate reduction in error rates or cleaning effort can justify the investment, given how expensive bad data truly is. Over the long term, the cultural shift to "do it right the first time" creates a data asset base that yields compounding returns – trustworthy data that can be used confidently to drive business strategy.

6. Conclusion

In this paper, we presented a shift-left approach to data quality enforcement, arguing that the most effective way to ensure high-quality data is to enforce quality at the point of origin. By focusing on FinTech and HealthTech industries, we highlighted use cases where data integrity is not only a technical concern but a mission-critical requirement. FinTech companies face stringent regulatory expectations and financial risks that demand accurate, timely data, while HealthTech organizations deal with sensitive patient information where errors can have life-or-death consequences. In both arenas, *preventive* data quality management – catching and correcting errors before they proliferate – offers a compelling solution to longstanding data problems.

Our proposed framework integrates people, process, and technology components to realize preventive data quality. It emphasizes early definition of data quality requirements, built-in validation in user interfaces and data pipelines, real-time feedback to data producers, and continuous monitoring and improvement. We illustrated how this works in practice with examples of transaction processing and electronic health records, showing tangible benefits such as reduced errors in risk reports and improved completeness of patient data. We also surveyed tool options ranging from simple validation libraries to enterprise data quality suites and highlighted how these can be harnessed to automate and scale quality enforcement across complex data ecosystems.

A key part of our discussion focused on ROI and business impact. The shift-left approach is not just an IT initiative; it delivers business value. By investing in doing things right the first time, organizations can avoid the much larger costs of downstream fixes, regulatory penalties, or faulty business decisions. We argued how the shift left mechanism can save up to 60% of the practitioner's monetary budget. We also discussed how to quantify these benefits and underscored that beyond cost savings, preventive data quality enhances trust – among regulators, customers, and internal decision-makers – and paves the way for more advanced data-driven innovation. In essence, high-quality data becomes an asset with compounding returns, fueling analytics, AI, and efficient operations.

7. Conflicts of Interest

The author declares that there is no conflict of interest concerning the publishing of this paper.

References

[1] T. C. Redman, “Bad Data Costs the U.S. $3 Trillion per Year,” *Harvard Business Review*, Sep. 2016.

[2] G. Press, “Cleaning Big Data: Most Time-Consuming, Least Enjoyable Data Science Task, Survey Says,” *Forbes*, Mar. 2016.

[3] Basel Committee on Banking Supervision, *Principles for Effective Risk Data Aggregation and Risk Reporting* (BCBS 239), Bank for International Settlements, Jan. 2013.

[4] N. G. Weiskopf and C. Weng, “Methods for assessment of electronic health record data quality and reuse for clinical research,” *Journal of the American Medical Informatics Association*, vol. 20, no. 1, pp. 144–151, 2013.

[5] B. Boehm and P. Papaccio, “Understanding and controlling software costs,” *IEEE Transactions on Software Engineering*, vol. 14, no. 10, pp. 1462–1477, 1988.

[6] D. P. Ballou and H. L. Pazer, “Modeling data and process quality in multi-input, multi-output information systems,” *Management Science*, vol. 31, no. 2, pp. 150–162, 1985.

[7] A. Haug, F. Zachariassen, and D. van Liempd, “The costs of poor data quality,” *Journal of Industrial Engineering and Management*, vol. 4, no. 2, pp. 168–193, 2011.

[8] IBM Corporation, “What is shift-left testing?,” IBM *Think* Blog, Oct. 2021, [Online]. Available: https://www.ibm.com/think/topics/shift-left-testing.

[9] S. Ester and P. Huppertz, "Great Expectations: Data Validation for Modern Data Teams," in Proceedings of the 2021 Conference on Data Engineering, 2021, pp. 245-258.

[10] N. Marz and J. Warren, Big Data: Principles and Best Practices of Scalable Realtime Data Systems. Manning Publications, 2015.

[11] S. Schelter, D. Lange, P. Schmidt, M. Celikel, F. Biessmann, and A. Grafberger, "Automating Large-Scale Data Quality Verification," Proceedings of the VLDB Endowment, vol. 11, no. 12, pp. 1781-1794, 2018.

[12] Gartner, Inc., "Magic Quadrant for Data Quality Solutions," Gartner Research, 2023.

[13] T. Friedman and M. Smith, "Market Guide for Data Quality Solutions," Gartner Research, 2022.

[14] J. Kreps, N. Narkhede, and J. Rao, "Kafka: A distributed messaging system for log processing," 2, vol. 11, 2011, pp. 1-7

[15] "ISO 20022 Financial Services - Universal financial industry message scheme," International Organization for Standardization, 2013.

[16] J. M. Hellerstein, M. Stonebraker, and J. Hamilton, "Architecture of a Database System," Foundations and Trends in Databases, vol. 1, no. 2, pp. 141-259, 2007.