International Journal of Scientific Engineering and Research (IJSER)
ISSN (Online): 2347-3878
SJIF (2024): 6.623

GenDa Architecture: A Conceptual Data Processing Framework for Accelerating Residential Mortgage Origination

Madhan Gopal Perumal

The University of Texas at Dallas, Richardson, TX 75080 USA
Corresponding Author Email: *mxp165530[at]utdallas.edu*

Abstract: ***In the realm of residential mortgage origination, enterprises face complexities in automating their loan approval process. Enterprises encounter challenges when designing an effective data architecture to accelerate the loan approval process. Manual processing of loan application has long served as the keystone for mortgage approval; therefore, the impetus for a transformative mortgage origination process is more profound than ever before. This paper introduces the GenDa architecture, a methodology aimed at enhancing residential mortgage origination data processing. In this paper we outline at a high level the integration of Generative AI with Lambda architecture to create the Genda architecture, a paradigm designed to reshape and augment the design of data platforms for mortgage loan applications. This paper delves into practical implementation of GenDa architecture, the implementation concepts detailed herein may serve as a foundational guide for loan data processing, to automate the residential mortgage origination process. Additionally, this study aims to provide a roadmap for enterprises aiming to navigate the transition towards a more robust, innovative, and efficient residential home loan origination process.***

Keywords: Lambda Architecture, Modular Data Architecture, Automating loan origination, Generative AI. Cloud Data Management. Residential Mortgage

1. Introduction

Can mortgage lenders apply a methodology or framework to process data to automate the residential mortgage loan origination process? Mortgage lenders operate in highly competitive environments. The Great Financial Crisis (GFC) of 2008, transformed the landscape of residential mortgage applications. Contrary to the popular belief that unsustainable levels of credit to low-income and subprime borrowers contributed to the GFC, middle-income, high-income, and prime borrowers also increased their share of delinquencies contributing to the GFC crisis(Adelino et al.,2016). Thus, the GFC crisis may not solely be attributed to delinquencies by low-income and sub-prime borrowers. Therefore, credit evaluation risks for residential mortgage applications exist across all income groups. To mitigate these risks, it is a necessary to process data effectively and optimize the overall residential mortgage origination process, irrespective of the income levels.

As the GFC unraveled, mortgage fraud investigations increased. Mortgage loan fraud is defined as “the material misstatement, misrepresentation, or omission by an applicant or other interested party, relied upon by an underwriter or lender to fund, purchase or insure a loan”(Federal Bureau of Investigations [FBI], 2010). A mortgage loan originates from an applicant or borrower applying for it. After the application, mortgage loan origination typically involves multiple stages of data processing, including credit and financial evaluation, property appraisal, title search, underwriting, and final approval. Figure [1] shows the stages involved in the origination of mortgage loans. Each of these stages may contribute to fraud. For example, poor underwriting standards and practices associated with subprime lending are major contributors to fraud(Nguyen and Pontell, 2010). Mortgage fraud led to additional regulations in the mortgage industry after the GFC. Thus, significant time is required to process mortgage application data because of the complexities involved at each stage of application. As a result, in the mortgage origination landscape, institutions are burdened with increasing costs in application processing, which impedes their ability to adapt to growing customer demands and regulatory requirements.

Recently, innovations in financial technology(FinTech) have been promising, and they may offer some assistance in reducing friction in processing mortgage origination data. FinTech lenders process mortgage applications 20% faster than other lenders (Fuster et al.,2019). FinTech lending grew annually by 30% from $34bn total originations in 2010 to $161bn in 2016 (Fuster et al.,2019). Additionally, FinTech default rates are approximately 25% lower than those of traditional lenders (Fuster et al.,2019). Literature suggests that FinTech lenders may rely on artificial intelligence(AI) for mortgage processing. Technical functionalities associated with AI have been in place for decades, but it is only recently that advancements in generative AI have made the use of this technology ubiquitous in mortgage origination. If a framework is formulated to accelerate data processing in mortgage origination, by utilizing generative AI (LLMs) to streamline the loan origination process, then it may open up opportunities to generate less risky loans and assist in the early detection of fraud in mortgage originations. Efficient and effective processing of mortgage origination data, from application to loan closing, is crucial for a lender’s success. Even with FinTech innovations, the average time to fund a mortgage loan may range from 30 to 60 days.

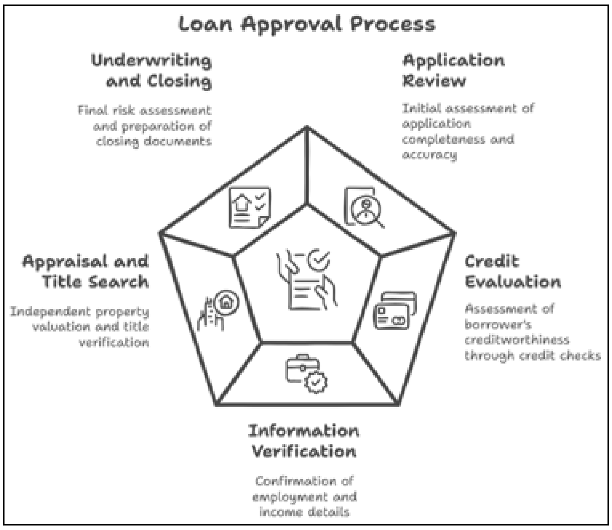

Loan Approval Process
Underwriting and Closing
Final risk assessment and preparation of closing documents
Application Review
Initial assessment of application completeness and accuracy
Appraisal and Title Search
Independent property valuation and title verification
Credit Evaluation
Assessment of borrower's creditworthiness through credit checks
Information Verification
Confirmation of employment and income details

Figure 1: Mortgage Loan Process

Approximately 60% of the loans are closed later than the target closing date as set by the lender at the beginning of the process (Arin et al.,2021). Delays can lead to customer dissatisfaction, increased operational costs, and cause potential loss of business to agile competitors. At a mortgage enterprise, reducing the time taken to close loans without compromising the accuracy and compliance of the process is essential for maintaining a competitive edge, and enhancing customer experience.

Several factors contribute to delays in loan processing:

1) Manual Processing: Verifying and processing various documents manually is time consuming and prone to errors.
2) Task Dependencies: Traditional processes often require tasks to be completed in sequence, which causes bottlenecks.
3) Third-Party Dependencies: Ordering and receiving title searches, appraisals, and inspections from third-party vendors can introduce significant delays.
4) Evaluation Complexity: For example, the underwriting process, which involves assessing the applicant's creditworthiness and risk, is often manual and labor-intensive.

To accelerate and optimize this mortgage origination process, this study proposes the GenDa architecture (an integration of Lambda Architecture and Generative AI). By integrating these, we aim to:

1) Automate document processing: Using generative AI models to automatically request and process necessary documents, such as title searches, appraisals, and inspections.
2) Enhance real-time decision-making: Implementing a robust real-time processing layer to handle immediate data updates and reduce bottlenecks.
3) Enable auto-approval: Train and fine-tune generative AI models capable of automatically approving loans when all the required documents and criteria are met. This allows us to streamline the underwriting process.
4) Reduce closing time: Significantly reduce the time from loan application to closing, aiming to reduce the average processing time of 30-60 days to a shorter timeframe.

Using GenDa, we aim not only to speed up the loan origination process, but also to improve accuracy, reduce operational costs, and enhance overall customer satisfaction. For illustration, this paper details the implementation of these solutions using Azure cloud services and outlines the tangible benefits of this approach

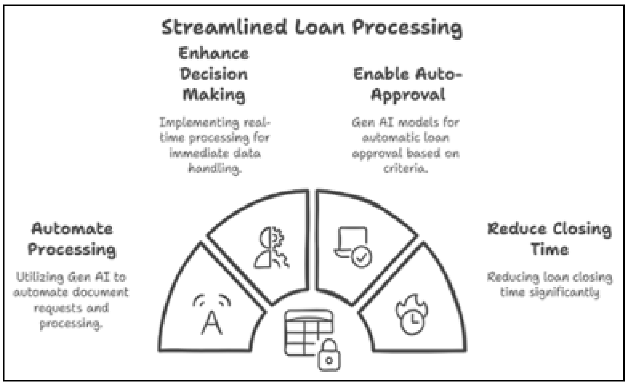

Streamlined Loan Processing
Enhance Decision Making
Implementing real-time processing for immediate data handling.
Enable Auto-Approval
Gen AI models for automatic loan approval based on criteria.
Automate Processing
Utilizing Gen AI to automate document requests and processing.
Reduce Closing Time
Reducing loan closing time significantly

Figure 2: GenDa Processing

2. Recent Literature Findings

Recent studies on the application of AI in the mortgage industry have demonstrated advancements in automating document processing, risk assessment, etc. There are papers that discuss how AI is applied in data processing, Kotecha et al.(2024) explore AI driven approaches for unstructured document analysis, Hilal et al.(2022) review anomaly detection techniques and recent advances. Recent literature discusses the use of large language models (LLMs) to extract information from documents and automate tedious tasks, such as data entry or verification. However, there is a lack of literature focusing on designing data architectures to integrate AI, especially generative AI.

2.1 Perspectives on data design & architecture in mortgage data processing

Mortgage origination data may be fragmented, as they are sourced from different systems, such as CRMs, credit bureaus, and third party verification systems. A framework for APIs and microservices data extraction, alongside pipelines to ingest and transform the data, is required for processing the mortgage origination data. Recent studies have focused on real-time data pipelines, serverless data architectures, and building data lakes to consolidate data while overcoming challenges related to the performance of data architectures. In a recent review, Kranas et al.(2022), discussed the simplification and acceleration of data pipelines in digital finance. Werner and Tai (2024) presented an architecture for serverless big data processing. Recent studies have discussed cost-effective data processing pipelines and efficient utilization of cloud-native infrastructures. Additionally, there are studies with growing interest in AIOps, use of knowledge graphs for linking data points, and semantic data layers. However, there is a lack of literature exploring a process that combines batch and stream processing of data repositories to assist mortgage loan processing.

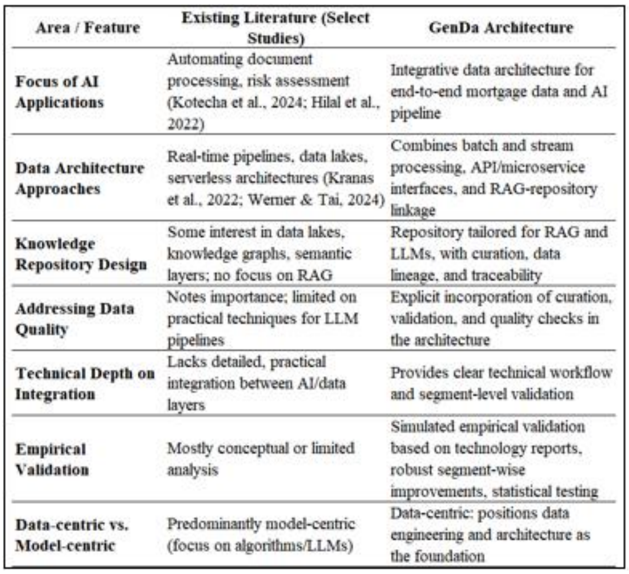
2.2 Perspectives on LLM in mortgage data processing

The mortgage industry is regulated, and loan application data are highly sensitive. Ensuring compliance with mortgage regulations requires safe processing of data, which presents risks in utilizing LLMs. LLMs can think, analyze, and generate impressive content but encounter challenges such as hallucination and outdated knowledge. Retrieval-Augmented Generation(RAG) has emerged to overcome the challenges, and enhance the accuracy and credibility of the LLMs, particularly for knowledge-intensive tasks (Gao et al.,2023). An RAG-based LLM may produce context-rich responses. Moreover, RAG improves factual accuracy in generative responses(Lewis et al.,2020). However, even in RAG, LLMs require concise input owing to input-token limitations (Pesl et al., 2024). There is a lack of literature addressing the design of knowledge repositories supporting AI, particularly LLMs. Additionally, there is a lack of depth in addressing the data lineage, traceability, and security of data in architectural design for repositories feeding LLMs.

2.3 Limitations

Th existing literature lacks depth in data collection design and practical technical considerations, such as what data is needed or how curation has to be applied to data fed to LLMs. Despite advancements, much current research fails to recognize the role of data architecture in building knowledge repositories that support the RAG based LLM, especially in mortgage data processing. Poor data quality may become a major bottleneck in the implementation of a successful RAG based LLM. Therefore, an efficient data architecture design is essential to minimize the knowledge gaps between the data and LLM applications. Beyond model-centric research on LLM, there is a need to focus on data-centric architectures. The GenDa design is centered around data, enabling a transparent view of LLM processing in the architecture. The proposed architecture bridges the gap between data engineering and LLMs.

Table 1: Comparative Summary: Recent Literature and Genda Architecture



Area / Feature	Existing Literature (Select Studies)	GenDa Architecture
Focus of AI Applications	Automating document processing, risk assessment (Kotecha et al., 2024; Hilal et al., 2022)	Integrative data architecture for end-to-end mortgage data and AI pipeline
Data Architecture Approaches	Real-time pipelines, data lakes, serverless architectures (Kranas et al., 2022; Werner & Tai, 2024)	Combines batch and stream processing, API/microservice interfaces, and RAG-repository linkage
Knowledge Repository Design	Some interest in data lakes, knowledge graphs, semantic layers; no focus on RAG	Repository tailored for RAG and LLMs, with curation, data lineage, and traceability
Addressing Data Quality	Notes importance; limited on practical techniques for LLM pipelines	Explicit incorporation of curation, validation, and quality checks in the architecture
Technical Depth on Integration	Lacks detailed, practical integration between AI/data layers	Provides clear technical workflow and segment-level validation
Empirical Validation	Mostly conceptual or limited analysis	Simulated empirical validation based on technology reports, robust segment-wise improvements, statistical testing
Data-centric vs. Model-centric	Predominantly model-centric (focus on algorithms/LLMs)	Data-centric: positions data engineering and architecture as the foundation

3. Background

Inspired by the lambda architecture, which is well suited for efficient data processing, this study introduces the GenDa architecture. GenDa integrates a Retrieval Augmented Generation (RAG) based Large Language Model(LLM) into the lambda architecture. The Lambda Architecture is composed of three layers (Cuzzocrea et al., 2018), namely, (a) Batch Layer , (b) Speed Layer, and (c) Service Layer. The GenDa Architecture is structured to process the data in four layers. It is designed to handle massive amounts of data and automate credit analysis processes involved in mortgage origination.

3.1 Data Considerations for optimizing Mortgage origination

To optimize mortgage loan origination using the GenDa architecture, we must identify the correct data sources and collect the data. To choose the data sources, we can align the business objectives into business contexts and define the data processing strategy within each context. This approach ensures that the data sources align with the business objectives. Moreover, this approach provides a clear understanding of where and how each piece of data will be used in strategic processing. The following are the contexts that we recommend in this paper.

Business Context: Property Information

Data: Property information is based on appraisals and valuations. The market value of a property is calculated solely as of a specific date, and it only provides an opinion of value relevant at that particular point in time (Quentin,2009-09). Accurate market values are critical for determining the value of collateral-backing the mortgages. Accurate appraisals and valuations help assess the loan-to-value ratio and other key metrics of loan origination.

Processing strategy: Having historical data on a property, helps in risk assessment and pricing strategies. Batch processing is suitable.

Business Context: Legal & Regulatory Information

Data: Regulations are crucial for mitigating future financial crisis (Aikman et al.,2019). Macroprudential policies may affect institutions' assets or liabilities. Macroprudential policies regulate metrics such as LTV, DTI, and PTI. Ensuring legal compliance is crucial for mortgage firms to conduct their business operations. Compliance with regulations protects both the lender and the borrower.

Processing strategy: Regulations may not be changed daily. Therefore, this data may be suitable for batch processing.

Business Context: Tax, Title & Insurance information

Data: Title searches and insurance data are required. Title searches verify the legal ownership of a property. Title insurance protects against future claims on property. To purchase a property, nearly all lenders require that the property buyer purchase the lender's title insurance policy (Hemphill, 2019). Property taxes affect borrower's DTI ratio. DTI is a key metric for determining loan approvals. Long-term loan performance depends on whether the property is located in an area where borrowers are required to have flood insurance (Kousky et al.,2020). Lenders need proof of adequate home insurance coverage to protect their collateral.

Home insurance ensures that the property's value is safeguarded, thus protecting the lender's investment.

Processing strategy: Systematic processing of Tax, Title, and Insurance information mitigates unexpected costs and ensures compliance with the regulations. Batch processing is suitable.

Business Context: Economic & Market data

Data: Market Analysis is conducted using Economic Indicators. Economic indicators such as interest rates, employment rates, and housing market trends impact mortgage origination. For example, a 1 percentage point increase in the rate on a 30-year fixed-rate mortgage reduces the first mortgage demand by 2 to 3 percent (DeFusco & Paciorek, 2017). Federal benchmarks and the activities of government-sponsored-enterprises(GSEs),directly influence mortgage rates. Therefore, economic indicators help in strategic planning and forecasting.
Processing strategy: Market and Economic analyses provide insights into market conditions, helping adjust loan terms and interest rates accordingly. Batch processing is suitable.

Business Context: Customer Information

Data: Loan applications are the primary source of customer information. This includes personal details, income, desired loan amount, and property information. Identity evaluation confirms an applicant's identity, prevents fraud and ensures security and legal compliance.

Processing strategy: Real-time processing applications allow for immediate feedback and quicker loan processing times.

Business Context: Credit Evaluation

Data: The credit evaluation assesses borrowers' creditworthiness based on their credit history and score. This is crucial to risk management.

Processing strategy: A real-time assessment of the borrower's financial reliability, mitigates risk. Speed processing is suitable.

Business Context: Income and Employment Evaluation

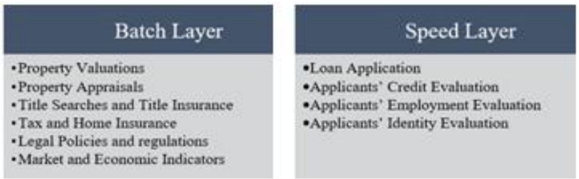
Data: Verifying employment status and income is essential to ensuring the borrower's ability to repay the loan.

Processing strategy: Real-time employment evaluation helps verify the stability and sufficiency of the borrower's income, thus reducing default risks. Speed processing is suitable.

Business Context: Identity Evaluation

Data: Identity validation is crucial for ensuring security and legal compliance. This is crucial for mortgage firms to eliminate fraud and conduct business operations.

Processing strategy: Streaming the borrower's identity will help protect against identity theft and fraud



Batch Layer

- Property Valuations
- Property Appraisals
- Title Searches and Title Insurance
- Tax and Home Insurance
- Legal Policies and regulations
- Market and Economic Indicators

Speed Layer

- Loan Application
- Applicants' Credit Evaluation
- Applicants' Employment Evaluation
- Applicants' Identity Evaluation

Figure 3: Batch and Speed Layer Entities

4. Method

In the GenDa architecture the four layers of data processing include

- Batch Layer: To process large volumes of historical data.
- Speed Layer: To handle real-time data.
- Serve Layer: To combine and integrate the speed layer data with the batch layer data.
- Digest Layer: Provides an innovative platform to automate loan application evaluation using a generative AI model and handle analytics.

4.1 Batch Layer

Once the data sources are finalized, we must process the data. To process large volume of historical data, the batch layer may be used. This layer aggregates and computes data. Figure [3], shows the data context for the batch and stream layers. Data aggregation and precomputations are crucial for more efficient and effective decision-making. We can reduce the costs involved owing to precomputations and aggregations using batch layer processing.

Figure [4] shows the GenDa batch process. The data contexts for our batch processing are

- Property Valuations
- Property Appraisals
- Title Searches and Title Insurance
- Tax and Home Insurance
- Legal policies and regulations
- Market and Economic Indicators

Property valuations: Property valuation is an estimate of the amount of money a home is worth. Property valuation is essential for determining the value of a home at a given point in time. Property valuation data can be sourced from several third parties, including real estate databases (Zillow, redfin etc.), real estate agents, realtor websites, public records of county assessors, and insurance companies. Additionally, a mathematical valuation model may be created using data points such as the number of bedrooms, bathrooms, and square foot. Automated valuation methods(AVM) are used to evaluate properties. However, banks are more likely to rely on real estate agent valuations to meet government-mandated loan-to-value(Reite, 2023). By batch processing property valuations and maintaining a history of property valuations, lenders may be able to mitigate risks.

Property Appraisals: Property appraisals are typically required by lenders prior to mortgage approval. Appraisals provide expert opinions on a property's market value. Appraisals include detailed information about the condition of the property, and the suggested improvements. Appraisals directly impact the contract prices of mortgage loans.

Appraisers aware of contract prices, were more than twice as likely to reach an appraised value at least equal to the contract price; on average, their valuations were 4.2%-to-8.3% higher than appraisers unaware of contract prices (Ericksen et al., 2019). Property appraisal data can be sourced from licensed appraisers' report, appraisal management companies, inspection services, and insurance claims. By batch processing property appraisals and maintaining a history, we can expedite the mortgage approval process. Property appraisals combined with valuations provide a holistic view of a property's worth and help avoid over-lending on a property that may not hold its value.

Title Searches and Title Insurance: A title search is used to verify the ownership of a property; it helps identify legal problems. Title insurance helps protect the lender and/or borrower from potential ownership disputes. Deeds generally do not guarantee indefensible title to recorded titles; thus, title insurance will defend against litigation(Hemphill, 2019). Title data can be sourced from title search companies' databases, public records, legal databases, etc., and batch processed. By batch processing the title information and maintaining a history, we can facilitate the faster processing of mortgage applications. Moreover, we can avoid the costs involved in audits, and the potential losses due to title issues

Tax and Home Insurance: Taxes and Home insurance are part of the total housing expenses. They help to estimate the housing payments. Higher property taxes and insurance increase monthly obligations, potentially affecting DTI ratio. By comparing historical tax data across different properties in the same area, lenders can assess the economic stability of the neighborhood. Additionally, a history of insurance claims in a neighborhood may indicate potential risks associated with a property, such as a high-risk area for natural disasters, For example, where flood insurance is required, the loan prepayment rate increases with property damage(Kousky et al.,2019). Property tax and insurance data are required to make informed decisions regarding the borrower's eligibility. Tax and insurance information processed using the GenDa architecture may allow for the automated calculation of monthly obligations and may assist in risk assessment.

Legal policies and regulations: Regulatory and supervisory frameworks exist in almost all countries and industries. A financial crisis can also be mitigated by improving the banks' assets, to reduce the size and potential losses of poor loans (Morgan et al.,2019). Macroprudential policies prevent the lenders from lending excessively during boom and help reduce loses in economic downturns. For example, LTV policies may significantly lower mortgage loan creation by 5.9%, after one year of implementation (Morgan et al.,2019). Therefore, regulations may be necessary for financial stability and to ensure compliance with fair lending laws, such as the Equal Credit Opportunity Act (ECOA) and General Data Protection Regulation(GDPR). The batch processing of legal and regulatory policies may help maintain a legal database. Mortgage lenders may incorporate the rules required in loan processing and automate the loan approval process. Moreover, we can avoid the costs involved in audits, and the potential losses arising from non-compliance with regulations.

Market and Economic Indicators: Market and Economic indicators provide a context for the broader economic environment , including interest rates, and housing market trends. An increase in the mortgage rate from 5 to 6 percent(100 basis point), leads to a decline in the first mortgage demand of 2 to 3 percent(DeFusco & Paciorek 2017). Thus, the economic indicators help assess risk in mortgage originations at the time of application. This data can be sourced from government databases and reports, financial data providers, public regulator databases(such as FDIC, Fannie Mae etc.), and real estate databases. By batch processing the market indicators and maintaining a history we can facilitate faster processing of mortgage applications by adjusting lending strategies to the current economic conditions. We can gain insights into market trends and support strategic decision-making and risk management.

All data that would require batch processing must be curated for consumption. To logically organize and curate the data, the GenDa architecture follows a medallion data design for the batch layer.

Medallion data architecture in Batch Layer

A medallion data architecture, also referred to as a multi-hop architecture, is a data design pattern used to organize data. The data would flow from Bronze to Silver to Gold and will have progressive improvements in the structure and quality as it flows from bronze to gold. In GenDa, batch processing would be as follows.

Bronze:

Raw Data Ingestion: Raw data collected from various sources would be stored without transformations. Load raw data using data engineering tool into a data lake(Azure Data Lake Storage shown in this paper).

1) Define data pipelines to pull data from various batch data sources identified.
2) Schedule regular data ingestions to keep data up to date.

Silver:

Data Cleaning and Standardization: Raw data is processed to remove duplicates, handle missing values, and standardize formats. Use any data engineering tool for data processing.

1) Perform necessary transformations such as joins, and calculations.
2) Use delta lake to store processed data.
3) Write the transformed data into Delta tables.

Gold:

Data Enrichment and Aggregation: Enhance data with additional insights, aggregate for analytics, and prepare for advanced querying.

1) Curate and enrich the data with historical trends and comparisons. Create tables and views for querying and extracting insights.
2) Enable use of predictive analytics and Dashboarding for extraction of insights.

Finally, the data in the batch layer must be updated and refreshed in a timely manner, to implement a suitable Change Data Capture (CDC) processing. The batch layer is crucial for successful functioning of the GenDa architecture. Therefore, the batch layer must be implemented with the required transformations and aggregations of data. After the implementation, the batch layer must be efficiently monitored for accurate processing and computations.

4.2 Speed Layer

The Speed Layer in the Architecture is used for real-time data processing and is critical for approving mortgage applications with low latency. Figure [4], shows the stream processed data. The main business contexts for our stream processing include.

- Loan Application
- Applicants' Credit Evaluation
- Applicants' Income Evaluation
- Applicants' Identity Evaluation.

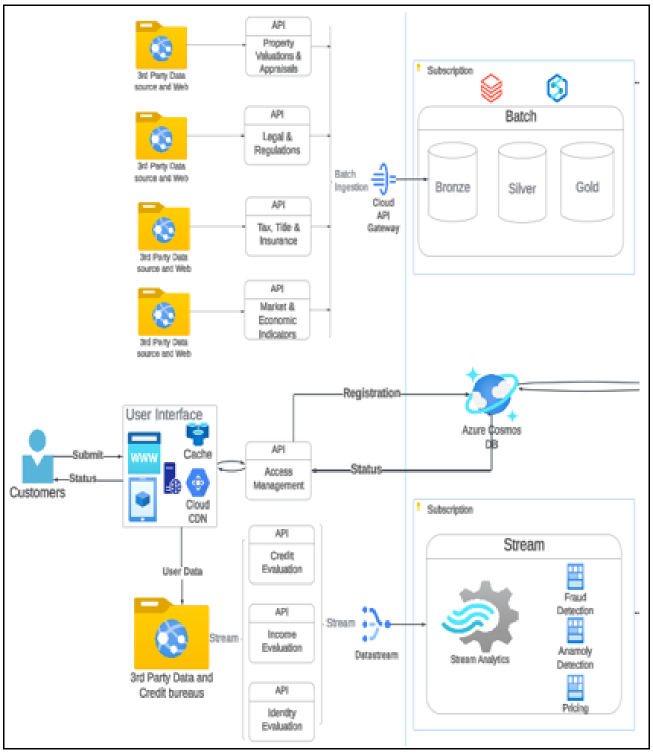


Figure 4: Batch & Speed Layer Processing

Loan Application: A loan application is the first step in mortgage origination. Loan application data may be streamed from online application forms, origination systems, or CRM systems. Potential borrowers may submit their information for loan consideration, including personal details, loan amounts, property addresses, and financial information. Loan application data may be stored in the Document DB, for quick querying. By speeding up the processing of loan applications, mortgage lenders can reduce delays arising from data accuracy and completeness issues identified at later stages of the loan lifecycle.

Applicant Credit Evaluation: Credit evaluation involves assessing applicants' creditworthiness based on their credit score, history, debt-to-income ratio, and other financial metrics. This is critical in determining the risk associated with lending to a particular borrower. When potential borrowers submit loan applications, the data required for credit worthiness can be streamed from credit bureaus and other third party or internal credit scoring models. Speed processing of the credit data would help lenders determine the borrower's eligibility quickly and eliminate the risks posed by new debts arising between their mortgage application and loan closure.

Applicant Income/Employment Evaluation: Verifying an applicant's employment status, income, and job stability is vital for assessing the ability to repay the loan. When potential borrowers submit loan applications, the data required for employment evaluation can be streamed from the employment verification services, payroll databases, and employer verification records. Speed processing of credit data would help lenders, mitigate lending risk.

Applicant Identity Evaluation: Identity evaluation confirms an applicant's identity and prevents identity theft. When potential borrowers submit loan application, real-time verification of identity can be performed from identity verification service databases, document verification systems, and government records. Speed processing of identity verification would protect mortgage originators, against identity fraud and ensure compliance with regulatory requirements.

In the context of this paper, Stream Analytics is proposed for processing and analyzing data streams from credit, employment, and identity evaluations. The speed layer architecture outlined in this paper allows stream analytics jobs to process the input streams and enables stream analytics queries along with advanced machine learning for the following purposes.

Fraud Detection:

1) Immediate Detection: Real-time detection of fraudulent activities, such as falsified employment information and inconsistencies in credit reports, helps prevent potential losses.
2) Pattern Recognition: Advanced analytics and machine learning help recognize fraud patterns such as multiple identities linked to a single individual.

Anomaly Detection:

1) Real-Time Alerts: Identify anomalies in application data, credit evaluation, employment record, and identity verification as they occur.
2) Proactive Measures: Enable proactive measures to address potential issues before they escalate.

Loan Pricing:

1) Dynamic Pricing: Adjust loan pricing(APR) dynamically based on real-time data inputs, to optimize risk and return.
2) Competitive Terms: Offer competitive loan terms tailored to individual applicants in real-time, improving customer satisfaction and market positioning

4.3 Serve Layer

The serve layer is critical for providing access to both real-time (speed layer) and batch-processed data. Figure [5] shows the serve layer for the proposed GenDa architecture. Through the serve layer, we ensure that loan data are available for consumption. Customer information from loan applications, property information, economic indicators, and other details are integrated to derive the required data points to be used by the LLM. For example, we may derive the loan-to-value(LTV) in the serving layer, which depends on property appraisals from the batch layer and loan amount from the Speed Layer. LTV is a key data point for loan processing, and loan-to-value ratios are effective measures to curb the expansion of mortgage debt (Morgan et al., 2019).

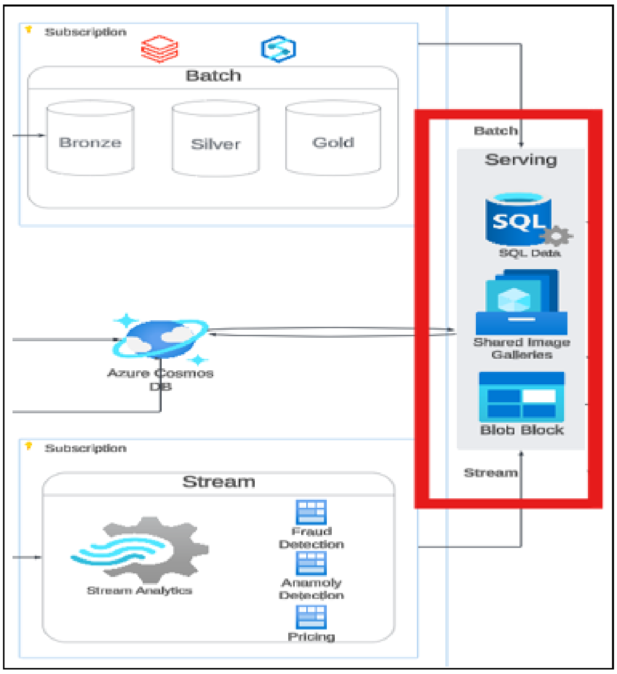


Figure 5: Serve Layer

Data Aggregation: Batch and Speed Layers
Aggregate the data from the batch layer (historical data) and speed layer (real-time data) to create comprehensive integrated data for Analytics and AI applications. Here are the steps

a) Create Data Pipelines

Define data movement activities to copy data from the batch and speed layers to the serve layer.

b) Transform Data

Clean, normalize, and integrate data from both layers using data flows. Data flow activities can be used to create a unified schema

c) Store Data

An SQL database can be used to store structured refined data from the batch layer and stream analytics queries. Populate the application data from the Document DB into SQL tables for processing. Create a data lake and define blob storages to organize unstructured reports(Json, XML etc.) such as credit bureau reports, and government identity reports.

4.4 Digest Layer: Generative AI Integration and Semantics

This layer supports data analytics, data-driven decision making, and auto-approval of a loan application using a Large Language Model (LLM). The RAG paradigm best serves the purpose of automating loan processing using LLM. There are multiple RAG paradigms such as Naive RAG, Advanced RAG, and Modular RAG (Gao et al.,2023). Any RAG paradigm can be incorporated into GenDa architecture. Credit reports, appraisal reports, and other documents should be broken into chunks. The chunks are vectorized and indexed using a vector store, this paper illustrates the use of Azure AI Search for vector store. Through vectorization we may be able to reduce the LLM's hallucinations. This study recommends using a vector similarity search to retrieve relevant contexts and provide it to the LLM to make informed responses for loan applications. The design outlined here will enable lenders to automate loan approval and underwriting for efficient processing. Figure [6], shows the digest layer.

a) Building The Semantic Layer

A semantic layer is created to abstract the underlying data complexity and provide a user-friendly interface for data analytics. This study recommends using the following components.

Analysis Services: For analysis and reporting. Create an analysis services instance and define a semantic model that integrates data from SQL Database, Blob Storage, and File Storage. Create data models, including measures, dimensions, and hierarchies, to facilitate data analytics. Build datasets and reports based on the semantic model.

Meta-Data Catalog: Create a data catalog for discovery, metadata, and governance. In addition, data lineage can be defined to enhance data discoverability and governance.
The data stored in serve layer is leveraged and utilized in the LLM with the Retrieval-Augmented Generation (RAG) paradigm. Integrating the LLM requires the following:

- Provisioning the LLM infrastructure
- Creating an AI workflow for processing data

b) Provisioning the Infrastructure:

Vector DB: Create a vector DB service and define the index schema, including content fields for loan applications, credit reports, appraisal reports, and other relevant documents.

Set up UI using web-app: Create a web app service using a web framework such as Flask (Python) or Node.js deploy the front-end code for chatting and querying from the RAG paradigm. This web application may enable loan officers to chat with the LLM and query any loan application status.

Set up Authentication and Security: Implement authentication and authorization to secure the integrations and APIs. Virtual networks and role-based access- controls (RBAC) can be implemented to secure access and communication to the various tools deployed.

c) Creating AI Workflow

Create an AI workflow to systematically provide all contexts of a loan application to the LLM for evaluation and response. The workflow may be Agentic or Non-Agentic. In this study, a non-agentic generative AI workflow is discussed for automating loan decisioning.

Set up a Chunking and Vectorizing strategy: An efficient RAG operation requires the documents to be chunked and indexed. The chunking strategy impacts the contexts retrieved for loan decisioning. Choose a chunking strategy that provides maximum efficiency; for example, an LLM based format specific approaches to chunking, will outperform naive chunking methods(Pesl et al., 2024). After chunking the documents, text embedding models can be used to convert the chunks into vectors.

Define Contexts retrieval mechanism for LLM Response: Use vector database semantic and keyword search capabilities or utilize vector search libraries to retrieve relevant information related to a loan application from the vector store.

Set up Loan Decisioning (Non-Agentic): Create business logic using functions to automatically approve/reject loan applications based on generated responses from the deployed generative AI model. Set criteria for auto-approval (e.g., credit score threshold, employment stability, property valuation). Update the loan application status in the SQL Database in Serving layer.

Additionally, functions should be in place to ensure compliance with fair lending laws such as ECOA, GDPR etc. Control functions include robust auditing mechanisms, regular bias testing, and adherence to explainability requirements. This would ensure that LLM-driven decisions do not result in disparate impact or unintentional discrimination.

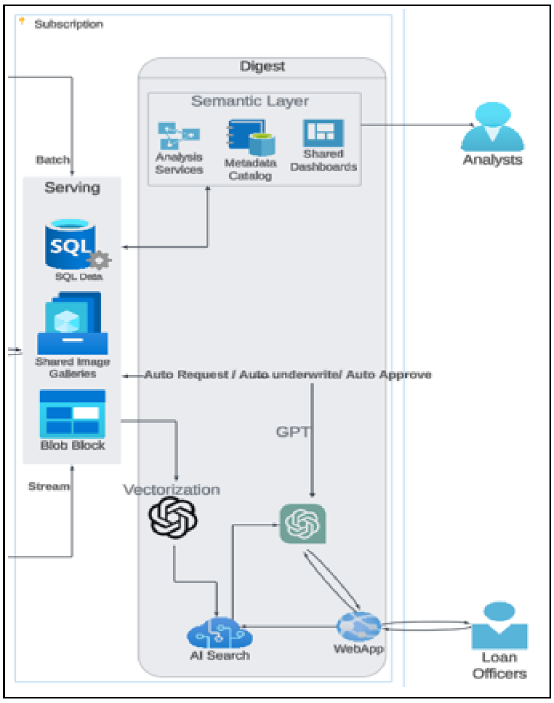


Figure 6: Digest Layer: Gen AI and Analytics

5. Emperical Validation

To assess the effectiveness of the proposed GenDa architecture, an empirical validation is conducted by comparing key operational metrics between legacy (baseline) processes and the new GenDa architecture. Stepwise process times for both scenarios are simulated based on data points from industry reports and practical lender references. The baseline performance metrics were derived from recent industry reports, white papers, and academic studies that highlight typical workflow bottlenecks and durations in current mortgage approval processes:

Data Entry & Collection: Manual entry is a significant bottleneck, with some lenders reporting that loan officers spend up to 40% of their time on data entry and form processing (CrossCountry Mortgage, n.d.; Mortgage Data Capture, n.d.). Manual entry is estimated at 30–60 minutes per application for this study.

Document Verification: Employment/income verification and manual document review can take hours to days. Manually verifying takes a few days to weeks (Argyle, 2022). Current workflows for gathering, validating, and verifying financials to close a loan often take days (Brahma et al., 2021). Therefore, document verification is estimated at 2–24 hours for this study, assuming no incomplete submissions.

Credit Check: Mortgage loan closing typically spans 45–60 days(Brahma et al., 2021), with the credit check segment in current workflows estimated at 10–30 minutes as part of embedded processes (Industry norms).

Underwriting: Conventional underwriting averages 1–2 days after complete submission (The Mortgage Reports, 2024).

Error Rate: Approximately 10% of applications require error correction in a conventional workflow (Appstek Corp, 2022).

The proposed GenDa data architecture leverages automation, AI-driven document processing, and rule-based underwriting to enhance efficiency. While no peer-reviewed publications provide per-step benchmarks, targets are constructed on documented case studies and technology reports:

Data Entry & Collection: Reduced to 1–2 minutes per application for this study. Automated ingestion and Optical Character Recognition (OCR) significantly reduce data extraction time (Docsumo, 2023). Automated document processing can reduce data extraction and entry to an average of 45 seconds (Multimodal, 2025)

Document Verification: Lowered to 15–120 minutes for this study. Automating the verification process can complete verifications in minutes to hours (Argyle, 2022). AI-driven extraction and agentic intelligent document processing enhance processing efficiency (AWS, 2025).

Credit Check: Reduced to approximately 1-5 minutes for this study. By using API-based, real-time data integration and AI, automated credit checks can provide near-instant feedback on credit applications (Akira AI, 2023).

Underwriting: Reduced to 1–4 hours for this study. Through algorithmic and AI-based rules, loan processing is reduced by up to 40% (Cflow,2025).

Error Rate: Reduced to 1–2% due to improved data accuracy and digital quality controls (Appstek Corp, 2022; True AI, 2023).

To empirically validate the effectiveness of the proposed data architecture, a synthetic dataset consisting of 5,000 mortgage application records was generated. These records were designed to mirror real-world heterogeneity in applicant demographics and process complexity. Each application was assigned attributes such as applicant age, income, credit score, employment status, loan purpose, documentation completeness, data quality, and complexity. Stepwise process times were constructed from the outlined industry-validated estimates, to simulate both the legacy(baseline) and proposed GenDa workflows. Addition to the applicant profile and loan information, indicators were added to segment the data. The applications were segmented along these features:

Documentation Completeness: Complete vs Incomplete
Data Quality: Clean vs Dirty
Application Complexity: Simple vs Complex
Credit Score Group: Subprime (<620), Near-prime (620–680), Prime (681+).

In both the legacy (baseline) and the GenDa(proposed) process step-specific processing times reflect industry-validated estimates. To model realistic bottlenecks and delays frequently reported in industry benchmarks, for both scenarios, processing times were increased by 1.5 times, and error rate increased by 1%, when an application was problematic. Problematic is when it was both complex, had dirty data, or had incomplete documentation. The third scenario was designed with 50% increased proportion of dirty and incomplete application.

Three scenarios were:
Baseline: Legacy process timings, and error rates.
Proposed: Reduced timings and error rates.
Proposed Stress: Increased proportion of problematic (dirty/incomplete) application with GenDa process.

Statistical significance of differences in total processing time between scenarios (Baseline vs. Proposed, Proposed vs. Proposed Stress) was evaluated using independent two-sample t-tests. The Kolmogorov-Smirnov test was employed against standardized data to assess conformity to the normal distribution, because violations of normality assumptions can undermine t-test results and necessitate the use of non-parametric alternatives. Additionally, homogeneity of variance across scenarios was assessed using Levene's test and the Fligner-Killeen test. Variance heterogeneity can significantly impact the Type I error rate of standard t-tests, which would require the use of non-parametric alternatives when violated. The Mann-Whitney U test was undertaken as the non-parametric equivalent to the independent t-tests. Segment-wise performance improvement was also analyzed across documentation completeness, data quality, complexity, and credit score strata.

To check the robustness of findings from the t-test or the Mann-Whitney U test, cross-validation was performed using 5-fold validation; the Kruskal-Wallis test was undertaken for an omnibus testing across all three scenarios simultaneously. Cross-validation assesses the stability of treatment effects across different subsets of the data. This approach helps distinguish between genuine system improvements and spurious findings that might arise from particular characteristics of the dataset. The Kruskal-Wallis test is a non-parametric test to confirm Mann-Whitney test. This multifaceted validation approach allows for demonstration of both the average and conditional effectiveness of the proposed data architecture. As real-world operational data may violate the assumptions of simple statistical tests, these multiple complementary approaches on simulated data may provide confidence that conclusions are robust.

6. Results & Discussion

This paper outlines a layer-by-layer implementation of the GenDa architecture. Figure [7] shows the full implementation architecture for our use case. To recap, we outlined the design of GenDa architecture with Batch, Stream, Serve, and Digest layers. The first is the batch layer, which helps derive significant business value. We collected the data and stored it in a raw data repository, referred to as the bronze layer. To curate the data loaded into bronze, we used silver and gold layers. As we move the data in a staged approach, we transform it into the required destination format. There are two approaches for this ingestion and transformation: namely ETL and ELT. ELT has more architecture flexibility, so the recommendation is to use ELT for GenDa. Second, a speed layer is implemented to handle data streams from loan applications, credit evaluations, employment evaluations, and identity validations. The speed layer allows lenders to effectively utilize real-time processing capabilities. The speed layer may help analyze fraudulent applications real-time, thereby it improves risk management. Additionally, with speed layer, we can optimize loan pricing and boost overall operational efficiency. Next, the serve layer integrates data from batch and speed layers. Finally, the digest layer is implemented to facilitate automatic loan decisioning and advanced data analytics. Additionally, this paper outlines how we can effectively implement an RAG pattern to automate loan approvals using LLM. Indexing and vectorizing loan documents, allows the LLM to overcome hallucinations and generate informed responses for loan applications.

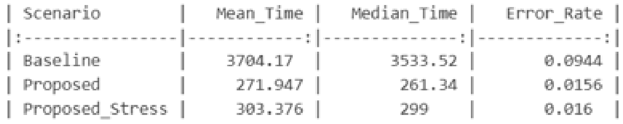
6.1 Empirical Results

The simulated validation demonstrates significant improvement in mortgage processing. The GenDa architecture reduces average end-to-end loan processing time compared to the legacy workflow. The mean total processing time for the baseline (legacy) workflow was approximately 3,704 seconds. Whereas the proposed data architecture reduced this figure to just 272 seconds (about 4 minutes), indicating over a 90% reduction in mean processing time. The proportion of applications requiring manual correction or exception handling dropped from 9.44% in the baseline scenario to 1.56% in the GenDa scenario, reflecting improved data quality. Independent t-tests conducted within the simulation indicate that reductions in both processing time and error rates are significant.

- Baseline vs Proposed: p-value ≈ 0.0000
- Proposed vs Proposed Stress: p-value ≈ 0.0000

Under the simulated stress scenario, introducing higher rates of data issues and incomplete documentation resulted in only a modest increase in mean processing time (from 272 to 303 seconds), and a slight further increase in error rate (from 1.56% to 1.6%), see Table 2.

Table 2: Scenarios: Total Processing Times

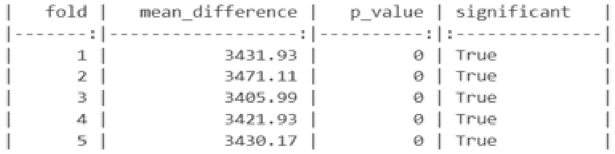


Scenario	Mean_Time	Median_Time	Error_Rate
Baseline	3704.17	3533.52	0.0944
Proposed	271.947	261.34	0.0156
Proposed_Stress	303.376	299	0.016

The t-tests indicated statistically significant improvements in processing time distributions relative to both the baseline and the stress-test scenarios. However, the normality test (Kolmogorov-Smirnov test) showed departures from normality. For the baseline scenario, the KS test had $p < 0.001$ (p-value = 3.0399e-20) and statistic = 0.0675, showing a relatively large deviation from the normal distribution. For the proposed scenario, the KS test had p-value= 1.7762e-10 and statistic= 0.0481. For the stress test scenario, the KS test had p-value = 1.3381e-05 and statistic = 0.0345. All three scenarios show departures from normality, but with differences in severity. The variance homogeneity tests showed evidence of unequal variances across scenarios, Levene's test: F = 7306.63, p ≈ 0; Fligner-Killeen: χ^2 = 8492.79, p ≈ 0. Therefore, both non-normality and variance inequality undermine standard t-test results. Thus, the Mann-Whitney U test becomes essential. The Mann-Whitney U test confirmed statistical significance. For Baseline vs Proposed, p-value≈0.0000 and the U statistic was maximum(5,000 × 5,000). For Proposed vs Stress, p-value 1.4895e-61, and the U statistic was less than half the maximum. The Mann-Whitney U test confirmed that the proposed (GenDa) processing times are significantly lower than the legacy (Baseline) processing times. Additionally, the segmented analysis reveals consistent benefits across all operational categories. The p-values by categories were all ≈0.000, showing statistical significance. In the documentation completeness category, the mean processing times reduced by 4151.8 minutes for incomplete applications and reduced by 3280.9 minutes for complete applications. In the application complexity category, the mean processing times reduced by 4211.2 minutes for complex applications and by 3193.7 minutes for simple applications. In the data quality category, the mean processing times reduced by 4214.2 minutes for dirty applications and by 3315.4 minutes for clean applications. See Table 4 for all categories.

Robustness Check: The 5-fold cross-validation demonstrates statistical robustness. The mean differences across the folds range from 3,405 to 3,471 minutes, and all p-values were essentially zero (see Table 3). Thus, all folds achieve statistical significance. Therefore, it confirms there are no spurious findings, as results are not dependent on particular subsets of data. The Kruskal-Wallis test had a statistic: 10,121.14 and p ≈ 0, evidence that at least one scenario differs significantly from the others, which confirms the findings of the pairwise evaluation of the Mann-Whitney U test.

Table 3: Cross Validation



fold	mean_difference	p_value	significant
1	3431.93	0	True
2	3471.11	0	True
3	3405.99	0	True
4	3421.93	0	True
5	3430.17	0	True

6.2 Limitations

While the empirical validation results support the effectiveness of the proposed data architecture, there are limitations. The analysis is based on a generated dataset designed to reflect mortgage lending. In this study, our empirical validation relied on synthetic data generated to model the mortgage origination workflow, using parameters sourced from industry benchmarks and published reports. First, the generated data may not capture the full spectrum of variability, correlations, and unanticipated issues present in real-world operational environments. Second, the process times used in the simulations are derived from industry estimates and standardized workflow assumptions. Actual process durations and escalation frequencies may vary across institutions, geographies, and over time, potentially affecting the magnitude of simulated validation. Therefore, the reported improvements should be interpreted as upper-bound estimates contingent upon favorable integration. The stress test scenario increases the proportion of problematic applications to approximate adverse conditions; however, further research is recommended to gauge the real workload fluctuations and system integrations. While the proposed architecture streamlines legal and regulatory compliance and automates credit decision workflows, it is not without challenges. The primary limitation is the complexity involved in ensuring that LLMs are free from biases that may contravene fair lending laws such as ECOA and GDPR. Furthermore, model-driven decisions may lack transparency, making explainability a persistent challenge, especially when customers seek clear reasons for loan denials. Another concern arises when increased competition within digital lending markets leads to an expansion in credit supply. In such scenarios, biases or misrepresented risk factors in LLM-driven credit models may result in short-term drops in default rates, giving an illusion of improved borrower quality. However, this can mask the accumulation of longer-term risks. While the RAG paradigm is integrated in the GenDa architecture to minimize hallucination risks in creditworthiness evaluations, it may not entirely eliminate these risks. As market conditions evolve, ongoing updates to the data corpus and retrieval parameters are necessary to ensure credit risk models adapt responsibly and fairly. The effectiveness of the system relies on continued evaluation, re-training, and independent audits to mitigate risks. As identified by Arin et al. (2021), some persistent challenges, such as borrower-side documentation errors, frequent regulatory policy changes, and communication gaps, may continue to exist. The GenDa system is designed to reduce, but not entirely eliminate, delay factors related to process automation and information flow. Future work may include field trials, adaptation to other financial products, and assessment of downstream impacts to extend the generalizability of findings beyond the specific mortgage lending context. Additionally, future work may include the integration of fairness and explicability metrics that can communicate decision rationales to both regulators and customers.

7. Conclusion

This study establishes and validates the GenDa architecture, integrating the best practices of lambda data architecture and cutting-edge Generative AI (GenAI) to transform mortgage loan origination. Our approach not only accelerates key process steps through automation but also demonstrates, via simulation, substantial gains in operational efficiency. Through an empirical simulation using realistic industry-derived benchmarks, the GenDa framework reduced the average end-to-end mortgage processing time. The simulation demonstrated that manual data entry, verification, and underwriting time can be reduced with the proposed architecture. The simulation also showed that problematic applications see the most excellent absolute improvement. With automated ingestion and intelligent document processing, GenDa can provide transformational benefits. The consistency across segments and statistical robustness offers confidence for the operational deployment of GenDa. Despite the potential benefits, some challenges may exist. This paper's primary contribution is the domain-specific integration of modern AI and data orchestration architectures within mortgage origination workflows. While the simulation demonstrates potential benefits, this work remains an incremental advance until substantiated with live operational performance data. Additionally, the current Gen AI models may reflect the biases that have existed historically in the mortgage market. However, opportunities exist for proactive, responsible AI model development designed to remove systemic barriers to mortgage credit access (Perry et al., 2023).

The GenDa architecture may help not only in accelerating the processing of loan applications but also in focusing more on deriving business value from data. In this study, Azure was chosen as the infrastructure to reduce the complexity of deployment; however, this architecture can be deployed on any cloud platform using similar tools and services. Enterprises can extend this framework to other cloud environments to achieve similar benefits. The framework presented here is scalable and opens possibilities for additional research. The integration of advanced AI and retrieval-augmented workflows (RAG), reliable ingestion and curation pipelines (ELT), and multi-layered data orchestration ensures that mortgage lenders adopting GenDa can realize transformational improvements. By leveraging this architecture, enterprises can achieve significant improvements in processing efficiency, data accuracy, and customer satisfaction. The following are the improvements that may be achieved.

1) Efficient Data Processing
2) Automated Loan Decisions
3) Robust Data Infrastructure

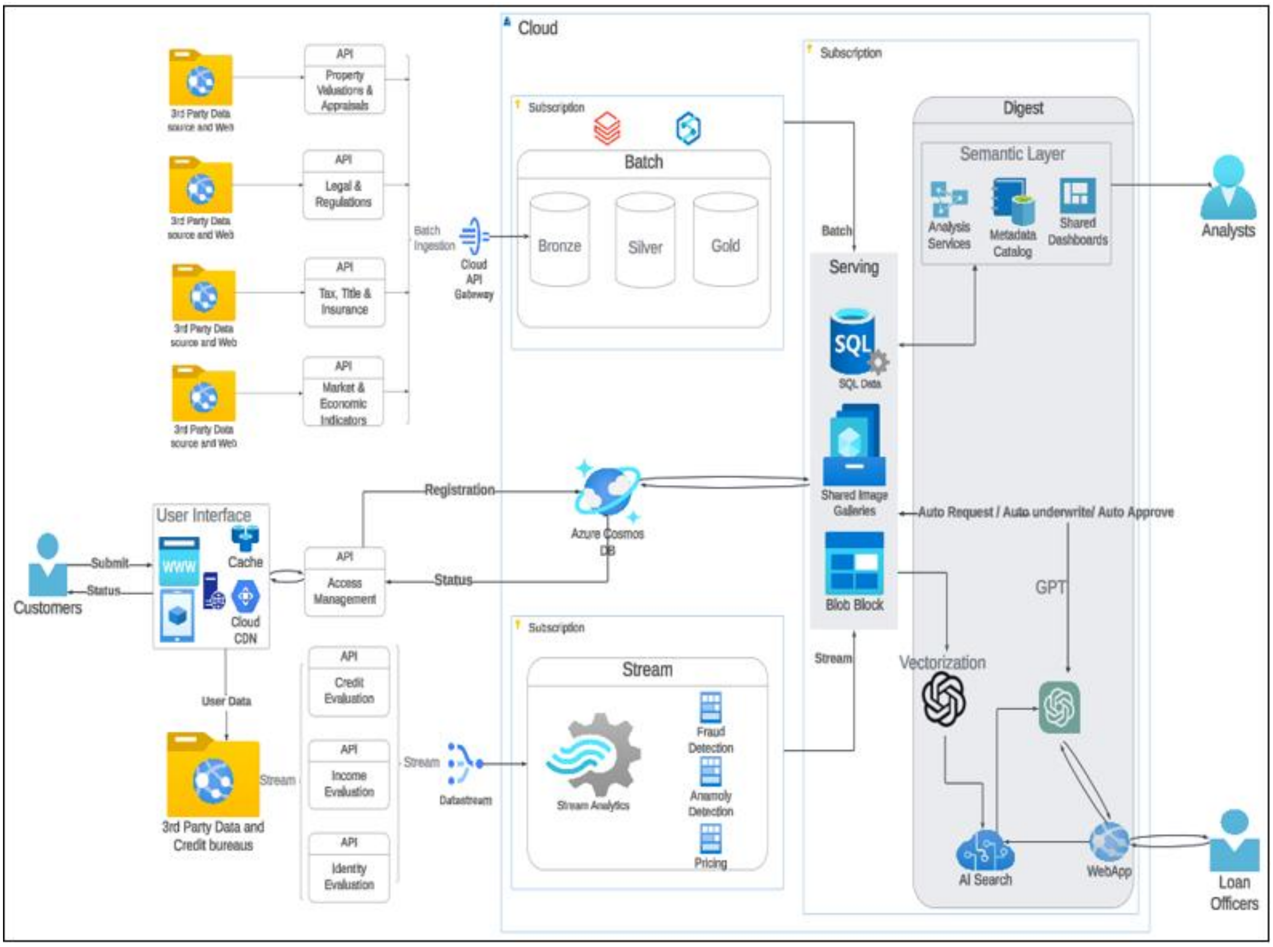
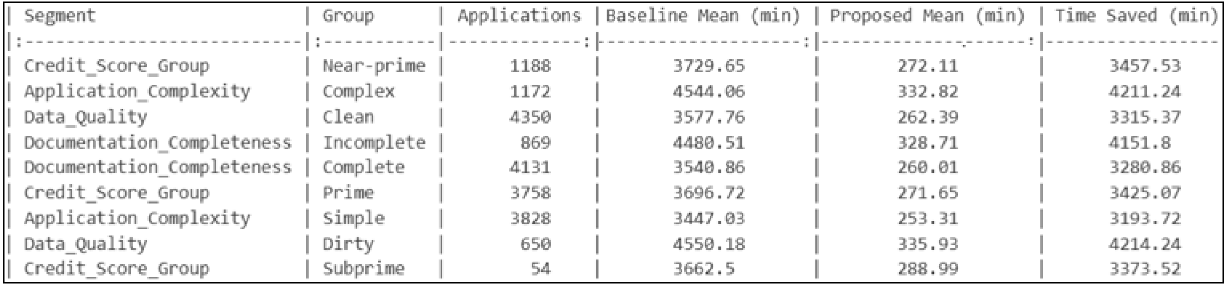


Figure 7: GenDa Architecture

Table 4: Segmented Processing Times



Segment	Group	Applications	Baseline Mean (min)	Proposed Mean (min)	Time Saved (min)
Credit_Score_Group	Near-prime	1188	3729.65	272.11	3457.53
Application_Complexity	Complex	1172	4544.06	332.82	4211.24
Data_Quality	Clean	4350	3577.76	262.39	3315.37
Documentation_Completeness	Incomplete	869	4480.51	328.71	4151.8
Documentation_Completeness	Complete	4131	3540.86	260.01	3280.86
Credit_Score_Group	Prime	3758	3696.72	271.65	3425.07
Application_Complexity	Simple	3828	3447.03	253.31	3193.72
Data_Quality	Dirty	650	4550.18	335.93	4214.24
Credit_Score_Group	Subprime	54	3662.5	288.99	3373.52

References

[1] Agarwal, Sumit, Gene Amromin, Itzhak Ben-David, Souphala Chomsisengphet, and Dou-glas D Evano (2014), Predatory lending and the subprime crisis, Journal of Financial Economics 113, 29-52.

[2] Nguyen, T. H., & Pontell, H. N. (2010). Mortgage origination fraud and the global economic crisis: A criminological analysis. Criminology & Public Policy, 9(3), 591–612. https://doi.org/10.1111/j.1745-9133.2010.00653.x

[3] Fuster, A., Plosser, M., Schnabl, P., & Vickery, J. (2019). The Role of Technology in Mortgage Lending. The Review of Financial Studies, 32(5), 1854–1899. https://doi.org/10.1093/rfs/hhz018

[4] Rubayyi Alghamdi, Martine Bellaiche (2023), An ensemble deep learning- based IDS for IoT using Lambda architecture. Cybersecurity 6,5. https://doi.org/10.1186/s42400-022-00133-w

[5] Ji, Z., Lee, N., Frieske, R., Yu, T., Su, D., Xu, Y., ... & Fung, P. (2023). Survey of hallucination in natural language generation. ACM Computing Surveys, 55(12), 1-38. https://arxiv.org/html/2202.03629v6#S14

[6] AWS. (2025). Jagadish Kumar, Eesha Kumar, & Thiyagarajan Arumugam, Build a big data Lambda architecture for batch and real-time analytics using Amazon Redshift https://aws.amazon.com/blogs/big-data/build-a-big-data-lambda-architecture-for-batch-and-real-time-analytics-using-amazon-redshift/

[7] Brahma, A., Goldberg, D. M., Zaman, N., & Aloiso, M. (2021) Automated mortgage origination delay detection from textual conversations, Decision Support Systems, 140, 113433.https://doi.org/10.1016/j.dss.2020.113433

[8] Federal Bureau of Investigation. 2007. 2006 Mortgage Fraud Report. https://www.fbi.gov/stats-services/publications/mortgage-fraud-2010

[9] Microsoft (2024). Piethein Strengholt, Tobias Zimmergren, Mia Sartschev, Liz Casey, Andrea Courtright. Data Domains. https://learn.microsoft.com/en-us/azure/cloud-adoption-framework/scenarios/cloud-scale-analytics/architectures/data-domains

[10] Miloslavskaya, N., & Tolstoy, A. (2016). Big data, fast data and data lake concepts. Procedia Computer Science, 88, 300-305. https://doi.org/10.1016/j.procs.2016.07.439

[11] Quentin, J. (2009). The subprime crisis--implications for property valuation? The revival of mortgage lending value. The Appraisal Journal, 77(4), 312-.

[12] Kumar, A., Mishra, A., Kumar, S. (2024). Data Lake, Lake House, and Delta Lake. In: Architecting a Modern Data Warehouse for Large Enterprises. Apress, Berkeley, CA. [6].https://doi.org/10.1007/979-8-8688-0029- 0_3

[13] Arasu, A., Babu, S., & Widom, J. (2006). The CQL continuous query language: semantic foundations and query execution. The VLDB Journal, 15, 121-142 [7]

[14] L'Esteve, R.C. (2023). Adopting a Cloud Platform. In: The Cloud Leader's Handbook. Apress, Berkeley, CA. https://doi.org/10.1007/978-1-4842-9526-7_5

[15] Hemphill, T. A. (2019). The title insurance industry: infusing innovation and competition. Business Economics (Cleveland, Ohio), 54(3), 177–181. https://doi.org/10.1057/s11369-019-00135-6

[16] Consider your options: Homebuyers should select a title company to protect their investment. (2014). Business Wire. https://www.proquest.com/wire-feeds/consider-your-options-homebuyers-should-select/docview/1556522150/se-2

[17] Foko Sindjoung, M.L., Fotseu Fotseu, E.B., Velempini, M., Fotsing Talla, B., Bomgni (PI), A.B. (2024). The Impact of Data Ingestion Layer in an Improved Lambda Architecture. In: Arai, K. (eds) Intelligent Systems and Applications. IntelliSys 2023. Lecture Notes in Networks and Systems, vol 824. Springer, Cham. https://doi.org/10.1007/978-3-031-47715-7_22

[18] Pal, G., Li, G., & Atkinson, K. (2018). Multi-agent big-data lambda architecture model for e-commerce analytics. Data, 3(4), 58.

[19] Pierre Brice, Wei Jiang, Guohua Wan, (2010) A Cluster-Based Context-Tree Model for Multivariate Data Streams with Applications to Anomaly Detection. INFORMS Journal on Computing 23(3):364-376 https://doi.org/10.1287/ijoc.1100.0407

[20] Microsoft (2025). Implement medallion lakehouse architecture in Microsoft Fabric.https://learn.microsoft.com/en-us/fabric/onelake/onelake-medallion-lakehouse-architecture

[21] Morgan, P. J., Regis, P. J., & Salike, N. (2019). LTV policy as a macroprudential tool and its effects on residential mortgage loans. Journal of Financial Intermediation, 37, 89–103. https://doi.org/10.1016/j.jfi.2018.10.001

[22] Aikman, D., Bridges, J., Kashyap, A., & Siegert, C. (2019). Would Macroprudential Regulation Have Prevented the Last Crisis? The Journal of Economic Perspectives, 33(1), 107–130. https://doi.org/10.1257/jep.33.1.107

[23] Reite, E. J. (2023). Mortgage lending valuation bias under housing price changes and loan-to-value regulations. Finance Research Letters, 58, 104677-. https://doi.org/10.1016/j.frl.2023.104677

[24] Kousky, C., Palim, M., & Pan, Y. (2020). Flood Damage and Mortgage Credit Risk: A Case Study of Hurricane Harvey. Journal of Housing Research, 29(sup1), S86–S120. https://doi.org/10.1080/10527001.2020.1840131

[25] Eriksen, M. D., Fout, H. B., Palim, M., & Rosenblatt, E. (2020). Contract Price Confirmation Bias: Evidence from Repeat Appraisals. The Journal of Real Estate Finance and Economics, 60(1–2), 77–98. https://doi.org/10.1007/s11146-019-09716-w

[26] Perry, V. G., Martin, K., & Schnare, A. (2023). Algorithms for All: Can AI in the Mortgage Market Expand Access to Homeownership? AI (Basel), 4(4), 888–903. https://doi.org/10.3390/ai4040045

[27] Cuzzocrea, A., Moussa, R., Vercelli, G., Chin, F. Y. L., Khan, L., Chen, C. L. P., Zhang, L.-J., & Lee, K. (2018). An Innovative Lambda-Architecture-Based Data Warehouse Maintenance Framework for Effective and Efficient Near-Real-Time OLAP over Big Data. In Big Data – BigData 2018 (Vol. 10968, pp. 149–165). Springer International Publishing. https://doi.org/10.1007/978-3-319-94301-5_1

[28] DeFusco, A. A., & Paciorek, A. (2017). The Interest Rate Elasticity of Mortgage Demand: Evidence from Bunching at the Conforming Loan Limit. American Economic Journal. Economic Policy, 9(1), 210–240. https://doi.org/10.1257/pol.20140108

[29] Armbrust, M., Ghodsi, A., Xin, R., & Zaharia, M. (2021, January). Lakehouse: a new generation of open platforms that unify data warehousing and advanced analytics. In Proceedings of CIDR (Vol. 8, p. 28)

[30] Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., Dai, Y., Sun, J., Wang, M., & Wang, H. (2023). Retrieval-Augmented Generation for Large Language Models: A Survey. https://doi.org/10.48550/arxiv.2312.10997

[31] Pesl, R. D., Mathew, J. G., Mecella, M., & Aiello, M. (2024). Advanced System Integration: Analyzing OpenAPI Chunking for Retrieval-Augmented Generation. https://doi.org/10.48550/arxiv.2411.19804

[32] Adelino, M., Schoar, A., & Severino, F. (2016). Loan Originations and Defaults in the Mortgage Crisis: The Role of the Middle Class. The Review of Financial Studies, 29(7), 1635–1670. https://doi.org/10.1093/rfs/hhw018

[33] Alvarez-Rodríguez, J. M., Zuñiga, R. M., Pelayo, V. M., & Llorens, J. (2019, July). Challenges and opportunities in the integration of the Systems Engineering process and the AI/ML model lifecycle. In *INCOSE International Symposium* (Vol. 29, No. 1, pp. 560-575).

[34] Mahadevkar, S.V., Patil, S., Kotecha, K. *et al.* Exploring AI-driven approaches for unstructured document analysis and future horizons. *J Big Data* 11, 92 (2024). https://doi.org/10.1186/s40537-024-00948-z

[35] Hilal, W., Gadsden, S. A., & Yawney, J. (2022). Financial fraud: a review of anomaly detection techniques and recent advances. *Expert systems With applications*, *193*, 116429.

[36] Kranas, P., Burgos, D., Jimenez-Peris, R., Mahíllo, J. (2022). Simplifying and Accelerating Data Pipelines in Digital Finance and Insurance Applications. In: Soldatos, J., Kyriazis, D. (eds) Big Data and Artificial Intelligence in Digital Finance. Springer, Cham. https://doi.org/10.1007/978-3-030-94590-9_2

[37] Appstek Corp. (2022). Why Modern Mortgage Lenders Are Racing To Automate Document Processing. https://appstekcorp.com/blog/why-modern-mortgage-lenders-are-racing-to-automate-document-processing/

[38] AWS. (2025). Autonomous Mortgage Processing Using Amazon Bedrock Data Automation and Amazon Bedrock Agents. https://aws.amazon.com/blogs/machine-learning/autonomous-mortgage-processing-using-amazon-bedrock-data-automation-and-amazon-bedrock-agents/

[39] CrossCountry Mortgage. (n.d.). The Underwriting Process. https://crosscountrymortgage.com/mortgage/resources/underwriting-process/

[40] Docsumo. (2023). Transforming Mortgage Underwriting with OCR. https://www.docsumo.com/blogs/ocr/mortgage-underwriting

[41] Fannie Mae. (2023). The Digital Mortgage Journey. https://www.fanniemae.com/media/49231/display

[42] True AI. (2023). AI in the Mortgage Industry: Beyond the Hype. https://true.ai/ai-in-the-mortgage-industry-beyond-the-hype/

[43] The Mortgage Reports. (2024). How long does underwriting take? https://themortgagereports.com/72583/how-long-does-underwriting-take

[44] Akira AI. (2023). Automated Credit Checks. https://www.akira.ai/blog/automated-credit-checks

[45] Mortgage Data Capture. (n.d.). What It Is & How It Works. | IBML. https://www.ibml.com/blog/mortgage-data-capture-what-it-is-and-how-it-works/

[46] Multimodal. (2025). Top Challenges in the Mortgage Industry Solved With AI. https://www.multimodal.dev/post/challenges-in-the-mortgage-industry

[47] Argyle. (2022). How Verification of Income & Employment for Mortgages Works. https://argyle.com/blog/how-verification-of-employment-voe-for-mortgages-works/

[48] Cflow. (2025). AI-Powered Workflow Automation for Mortgage and Loan Processing. https://www.cflowapps.com/ai-powered-workflow-for-mortgage-and-loan-processing/