Illia I.

DATA ENGINEER

Education
Computer Science and Software Engineering

Language proficiency
English – B2
Polish – C1

Domains
Banking
FinTech

Data Engineer with 5+ years of experience.
Data Engineer with over 5 years of experience in architecting vendor-independent open-source platforms on bare-metal Kubernetes and managing cloud pipelines in Azure. Expert in building scalable ETL/ELT pipelines using PySpark, self-hosted Apache Airflow, and dbt, with deep hands-on expertise in open-source Lakehouse stacks (Apache Iceberg, Trino). Advanced proficiency in SQL/T-SQL, time-series modeling (TimescaleDB), and DevOps practices (Docker, Terraform, CI/CD), ensuring resilient infrastructure and secure governance for sensitive banking data. Skilled in collaborating with financial stakeholders to transform complex transaction data into actionable regulatory and risk analytics.

Programming languages
Python, SQL.

Data Engineering
Apache Spark, PySpark, Apache Iceberg, Apache Airflow, dbt, Databricks, OpenMetadata.

Cloud
Azure (ADLS Gen2, Data Factory, SQL Database, VMs, Monitor, ACR, Key Vault, Entra ID, AKS, etc.)

Databases
MS SQL Server, Trino, PostgreSQL, TimescaleDB.

BI and Data Visualization
Power BI, Superset.

Data Science
Pandas, NumPy.

Data Modeling
Dimensional Modeling (Star Schema, Snowflake

Schema), Medallion Architecture, Entity-Relationship (ER) Modeling, Time-Series Modeling, Normalization / Denormalization.

DevOps
Docker, Docker Compose, Kubernetes, Grafana, Terraform, Prometheus, GitLab CI/CD.

Version control systems
Git, GitLab, Azure DevOps.

Projects

ENTERPRISE ON-PREMISE BANKING DATA PLATFORM

An enterprise-grade, self-hosted banking data platform built from scratch on bare-metal infrastructure without managed cloud services. The solution centralizes high-frequency core banking transactions, ledger reconciliations, and sensitive financial logs into a vendor-independent Lakehouse. By leveraging an open-source stack on Kubernetes, the platform ensures strict regulatory compliance, ACID transactions, federated analytical querying, and end-to-end data governance for executive risk reporting.

Project roles
Data Engineer

Period
04.2023 - till now

Responsibilities

- Designed a vendor-independent open-source data platform, managing custom container image building for self-hosted processing engines on Kubernetes;
- Engineered high-throughput PySpark pipelines on self-managed Spark clusters to ingest and transform core banking REST APIs and transactional streams;
- Designed scalable Apache Iceberg table architectures with hidden partitioning, schema evolution, and Nessie catalog integration for ACID financial ledgers;
- Executed Iceberg table maintenance workflows, automating snapshot expiration, data file compaction, and time-travel queries for regulatory audit recovery;
- Developed a federated SQL query layer using Trino to unify analytical access across Apache

Iceberg storage, PostgreSQL databases, and legacy sources;

- Optimized Trino resource groups and query plans to accelerate analytical processing for cross-departmental financial risk assessment;
- Developed SQL-based transformation layers using dbt, building incremental models, custom macros, generic tests, and automated schema docs;
- Deployed stateful and stateless data platform services on Kubernetes using custom Helm charts, configuring StatefulSets and persistent volume claims for reliable storage;
- Architected PostgreSQL and TimescaleDB schemas, leveraging hypertables, continuous aggregates, and compression policies to store time-series transaction telemetry;
- Managed self-hosted Apache Airflow orchestration, sensors, idempotent DAGs, and strict SLA alerting for banking pipelines;
- Implemented data governance, role-based access control, and automated lineage harvesting using OpenMetadata to secure sensitive banking records;
- Wrote advanced SQL transformations and reconciliation routines to validate financial ledger balances and prevent schema drift;
- Automated infrastructure provisioning and bootstrapping using custom Terraform modules and self-managed state backends;
- Developed modular Python automation packages and internal CLI tooling with robust unit testing for platform maintenance;
- Designed interactive Apache Superset dashboards to visualize liquidity KPIs and real-time transaction anomalies for executive risk committees;
- Monitored Spark job execution metrics, pipeline latency, and Kubernetes cluster health using Prometheus and Grafana;

- Engineered automated GitLab CI/CD pipelines for container image builds, dbt testing, and zero-downtime environment promotion;
- Authored comprehensive technical documentation, Source-to-Target mappings, and disaster recovery run-books for on-premise components;

Environment
Python, SQL, Apache Spark, PySpark, Apache Iceberg, Trino, Apache Airflow, dbt, Pandas, NumPy, PostgreSQL, TimescaleDB, OpenMetadata, Apache Superset, Kubernetes, Docker, Docker Compose, Terraform, Prometheus, Grafana, GitLab, GitLab CI/CD.

ENTERPRISE FINANCIAL RISK & TRANSACTION ANALYTICS

A high-performance data platform designed for the ingestion and analysis of global financial transactions. The system centralizes data from disparate banking cores to identify fraud patterns, monitor liquidity, and automate regulatory risk reporting, replacing manual auditing with a high-throughput automated ecosystem.

Project roles
Data Engineer

Period
04.2021 - 03.2023

Responsibilities
- Engineered scalable Azure Data Factory orchestration pipelines to ingest high-velocity transactional streams from MS SQL Server and external FinTech REST APIs;
- Developed distributed PySpark transformation modules on Databricks using DataFrame API to standardize heterogeneous ISO 20022 financial messaging formats;
- Tuned heavy PySpark jobs in Databricks by analyzing execution plans, resolving data skew, and implementing broadcast joins and strategic repartitioning;
- Built automated data quality and reconciliation frameworks using SQL and PySpark to ensure high accuracy for sensitive ledger balances;
- Designed a Medallion Architecture serving layer

within Databricks, building optimized Star Schema models for fast analytical querying;
- Refactored resource-intensive legacy SQL Server stored procedures into optimized set-based T-SQL logic to accelerate analytical reporting;
- Designed and deployed containerized data processing agents using Docker and Kubernetes, managing image building and resource allocation via ACR;
- Provisioned supporting infrastructure and network security groups for distributed processing VMs using modular Terraform code;
- Developed interactive Power BI dashboards with complex DAX measures to visualize institutional risk KPIs and liquidity for C-level executives;
- Configured Azure Monitor, Log Analytics, and Prometheus to track data pipeline SLAs, alerting on transaction volume anomalies;
- Implemented secure data access controls and identity management using Microsoft Entra ID and Key Vault to protect sensitive financial records;
- Executed historical data migrations from legacy on-premise databases into Azure Data Lake Storage Gen2 using incremental loading techniques;
- Utilized Pandas and NumPy within modular Python scripts to prototype complex fraud detection algorithms and data validation rules;
- Managed database indexing strategies, partitioning, and continuous query optimization across high-load MS SQL Server analytical databases;
- Created custom technical run-books and Source-to-Target mapping documentation to ensure vendor-independent system maintainability;
- Troubleshot distributed processing bottlenecks, resolving memory spills and network timeouts across hybrid cloud gateways;

- Automated routine infrastructure scaling and database maintenance tasks using custom Python scripting and Bash automation tools.

Environment
Python, SQL, Apache Spark, PySpark, Databricks, Azure (ADLS Gen2, Data Factory, VMs, Monitor, ACR, Key Vault, Entra ID, AKS, etc.), MS SQL Server, Power BI, Docker, Docker Compose, Kubernetes, Terraform, Prometheus, Pandas, NumPy, Azure DevOps.

Professional skills

SKILLS		EXPERIENCE IN YEARS	LAST USED
Programming languages	Python	5	2026
	SQL	5	2026
Data Engineering	Databricks	2	2023
	Apache Spark	5	2026
	PySpark	5	2026
	Apache Iceberg	3	2026
	Apache Airflow	3	2026
	dbt	3	2026
	OpenMetadata	3	2026
Cloud	Azure	2	2023

SKILLS		EXPERIENCE IN YEARS	LAST USED
Databases	MS SQL Server	2	2023
	Trino	3	2026
	PostgreSQL	3	2026
	TimescaleDB	3	2026
BI and Data Visualization	Power BI	2	2023
	Superset	3	2026
Data Science	Pandas	5	2026
	NumPy	5	2026
DevOps	Docker	5	2026
	Docker Compose	5	2026
	Kubernetes	5	2026
	Grafana	3	2026
	Terraform	5	2026
	Prometheus	5	2026
	GitLab CI/CD	3	2026
Version control systems	Git	5	2026
	GitLab	3	2026
	Azure DevOps	2	2023