witboost

A European Bank automates end-to-end data product delivery with Witboost

Large European Bank
Success Case



witboost.com

Executive summary

A major European Bank established a self-service, standardized, and governed approach to data product delivery with Witboost. Integrated with the bank's technology and data ecosystem, Witboost enabled guided product definition and policy checks to automated deployment, execution, and a data contract-driven lifecycle.

Witboost coordinated the lifecycle and governance controls. Integrated platform tools provisioned resources, executed workloads, and performed data validations. This gave producers a repeatable delivery process without replacing the bank's data platform or catalog.

Key Results

The declarative data product automation developed with Witboost achieved the following results:

30%
Lower Delivery & Operational Costs

45%
Higher Data Quality at Scale

55%
Faster Time to Value

99%
Data Policy Coverage

93%
Improved Data Product Intelligibility

2X
Faster User Onboarding

Foundation for Enterprise Scale Adoption

The European Bank established a clear foundation for scale with its new operating model, enabling adoption at scale after the successful initial rollout:

- Proven approach for standardizing data product development across multiple domains
- Established reusable patterns that accelerate onboarding, delivery, and operational support
- Created a foundation for AI-assisted and agentic data product automation capabilities

Key learnings

01

Metadata is the foundation of automation

High-quality and machine-readable metadata is a key enabler of standardization and data governance.

02

Embedded Data Governance is a must

Compliance-by-design improves consistency.

03

Data Contracts connect producer commitments to operational checks

Declaration alone is not enough without deployment and runtime validation.

04

Standardization enables and accelerates scale, onboarding, and handovers

Reusable templates and blueprints significantly reduce implementation effort.

05

Self-service delivery velocity

Guided user experiences and automated workflows.

06

Automation and governance complement each other

Automation without governance creates risk while governance without automation limits scalability.

Key Witboost Capabilities Used

The most used Witboost capabilities and features by the European Bank as evaluated by the bank's Witboost users:

01 Blueprints & Templates Management

Witboost standardizes data product specification by providing centrally governed, open-standard–based blueprints and templates that guide users through a consistent, structured, and machine-readable definition process. All of this enables scalable and high-quality data product development.

Centralized Template (Blueprint) Management

Witboost provides a central catalog of reusable templates (blueprints) for different types of data products. Each template defines:

- Required metadata structure (fields, schema, ownership, SLA)
- Data contract format
- Allowed configurations and dependencies

Template as a Governed Artifact

Templates are treated as controlled, versioned code artifacts. Updates to templates (e.g., governance rules, metadata fields) are:

- Managed centrally
- Version-controlled
- Propagated consistently across new data products

Declarative Specifications via Templates

Templates define a declarative model for data products. Users fill in structured inputs instead of writing code. These inputs are validated against:

- Template rules
- Organizational standards

DevOps Integration (Key Enabler for Automation)

Each data product is linked to a dedicated DevOps repository. The generated descriptor and template outputs are:

- Automatically committed to version control
- Managed using standard Git workflows

02 Enabling Metadata Descriptor

Witboost enables metadata descriptors as a machine-readable, version-controlled, and blueprint-driven artifact that standardizes data product specifications as Source of Truth and directly drives automated pipeline generation and governance enforcement.

Descriptor as the Core Artifact

Witboost generates a metadata descriptor as the central representation of a data product. The descriptor acts as the single source of truth, capturing:

- Data contract
- Metadata attributes
- Technical configurations
- Pipeline logic references

Declarative Specification Model

Data products are defined declaratively via metadata, not imperative code. Users specify what the data product should do, not how it is implemented. The descriptor abstracts infrastructure complexity.

Blueprint-Driven Descriptor Creation

Descriptor structure is derived from predefined blueprints/templates. Blueprints enforce:

- Mandatory metadata fields
- Standard data contract formats
- Predefined configuration patterns

Machine-Readable & Structured Format

Descriptor is generated in a machine-readable format (e.g., JSON/YAML-like). Supports:

- Automation by execution engines
- Integration with pipelines and tools
- Validation using policy frameworks exposed as MCP tool
- Exposed as MCP tool

03 Enabling Computational Governance

Witboost enables computational governance by providing a fully integrated policy-as-code framework with centralized policy management, template-level enforcement, real-time validation, and DevOps-integrated execution. These ensure that data governance is automated, consistent, and enforced across the entire data product lifecycle.

Policy Definition & Management (Policy-as-Code Engine)

Witboost provides a framework to define governance rules as code-based policies. Policies can be expressed declaratively and applied across: metadata attributes, data contracts, and pipeline configurations.

- Central definition of policies (e.g., mandatory fields, schema rules)
- Reusable policy libraries
- Support for design-time and run-time policy definitions

Centralized Policy Governance & Lifecycle Management

Policies are managed as version-controlled, governed artifacts. Witboost allows versioning of policies, controlled updates and rollouts, and backward compatibility checks

- Policy lifecycle management (draft → grace → active → deprecated)
- Change tracking and audit history
- Standardized governance rollout across environments

Design-Time Validation Engine

Witboost provides a validation engine that evaluates policies during data product definition.

- Real-time validation of user inputs against policies
- Shift left and self-service validation

Runtime Policy Enforcement Integration

Witboost integrates governance policies with execution frameworks, demonstrated by integration with internal technologies and execution engines.

- Automatic propagation of policies into runtime pipelines.
- Enforcement of data quality thresholds, schema validation, SLA monitoring, and data retention rules

Real-Time Feedback & User Guidance

Witboost provides interactive governance feedback in the UI.

- Immediate alerts on policy violations
- Guided correction of non-compliant configurations
- Context-aware recommendations
- Notifications via corporate channels

DevOps Integration for Governance Enforcement

Policies and descriptors are integrated with DevOps workflows.

- Policy validation embedded into CI/CD pipelines
- Automated checks during Code commits, Pull requests, Deployment stages

04 Technology Adaptor Capability

Enabling Standardized Data Product Automation Across Diverse Technology Ecosystems

Witboost's Technology Adaptors enable data product specifications to remain technology-agnostic while supporting automated deployment and execution across different data platform technologies.

Technology Abstraction Layer.

- Decouples data product specifications from underlying implementation technologies.
- Allows producers to define data products using standardized blueprints and metadata rather than platform-specific configurations.
- Supports a consistent user experience regardless of the target execution environment.

Blueprint-to-Technology Mapping

- Technology adaptors translate standardized data product descriptors into platform-specific deployment artifacts.
- Enables a single blueprint to be reused across different technology stacks while maintaining a common specification model.
- Reduces the need for teams to understand platform-specific implementation details.

Automated Provisioning & Deployment

- Generates technology-specific configurations, infrastructure definitions, and deployment artifacts automatically.
- Integrates with DevOps pipelines to automate deployment and lifecycle management.
- Ensures consistent implementation of enterprise standards across platforms.

Standardization with Flexibility

- Promotes a common metadata and governance model across all data products.
- Allows platform teams to implement technology-specific capabilities through adaptors without changing the user-facing specification process.
- Ensures consistency while preserving flexibility to support different use cases and target environments.

Introduction – Situation in Context

Creating, managing, finding, understanding, moving, and consuming data in multinational banking organizations is time-consuming and manual. The stakes for a mistake could not be higher, so data standards, workflows, quality, and the data governance policies put in place need to be as rigorous as possible.

Even with established standards, workflows, and policies, any such enterprise-level organizations with ARRs in the hundreds of millions needs to trust its people to respect them, and be as thorough as possible before pushing their data use cases into production.

Trust and people are never the issue. After more than 10 years of experience with customers we've narrowed it down to a single problem: overhead when an organization scales its data operating model.

The manual, repetitive work scales with the organization while becoming increasingly inconsistent. The data governance checks become a bottleneck as well, as they scale with the organization, while the headcount remains the same. This overcharged capacity and the resulting bottlenecks greatly afflict data quality and compliance, introducing the risk of customer dissatisfaction, massive fines, and reputation loss.

For a large Western-European bank with over 10M retail and business customers this situation became reality when their CDO observed their Data Product Development processes slowly start slipping in terms of delivery time, errors, and higher costs.

The same overhead affected the agreement between producers and consumers. Defining a schema was only one part of the work; teams also needed to make business meaning, quality expectations, and responsibilities explicit, then keep those commitments aligned with the data as products evolved.

Challenges

The situational report pinpointed the following issues:

Variable Quality

Manual execution of Data Product Development practices led to inconsistencies and increased error rates

Long Time-to-value

Long time-to-market from ideation to production due to manual and fragmented practices

Limited Scalability & Increased Risk

Lack of automated control resulted in quality & compliance risks (governance was manual, non-standardized, and after-the-fact)

Higher Costs

Limited standardization led to necessity of "reinventing the wheel" for each Data Product (high effort, low throughput)

Solution

The solution required a systematic approach that could scale with the organisation. Further analysis revealed a golden path towards solving the bank's challenges: Moving from manual practice and operations towards declarative data product automation.

Data contracts provided a way to connect these concerns. A shared, machine-readable agreement could feed delivery automation and quality checks, rather than leave each team to translate documented expectations into its own implementation.

In order to impact all affected areas, the team identified 3 pillars to which it could apply this new approach.

These 3 pillars needed the following capabilities:

01 Self-Service Data Product Operations

A self-service portal that enabled data product management in a consistent and transparent manner throughout the entire lifecycle (planning, implementing, operating, retiring).

02 Standardized Data Product Operations

Declarative blueprints and templates based on open standards and metadata-driven automation to standardize data product and enable fast development, including ODCS Bitol for data contracts .

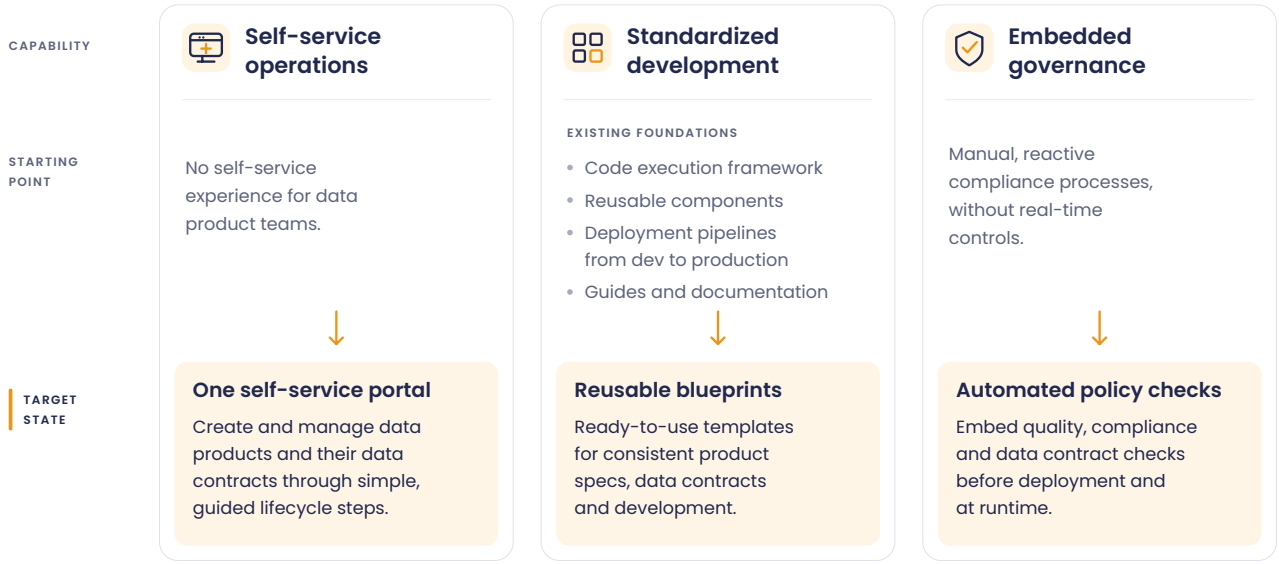
03 Embedded Computational Governance

Data products compliant by design, with automated guardrails on quality and compliance standards, which enable real-time insights.

Data contracts connect all three pillars: self-service for definition, templates for repeatable structure, and configured governance checks before deployment and at runtime.

The path to automated data product delivery

Once identified, the European bank moved to assess the current status of these 3 pillars and the level they wished to evolve each pillar to.



Once the goals were established, the team had to evaluate existing platforms on the market that could provide these capabilities.

After selecting Witboost, the two teams worked together on an implementation that moved the 3 pillars from the starting point to the target state, and even beyond.

Implementation objectives

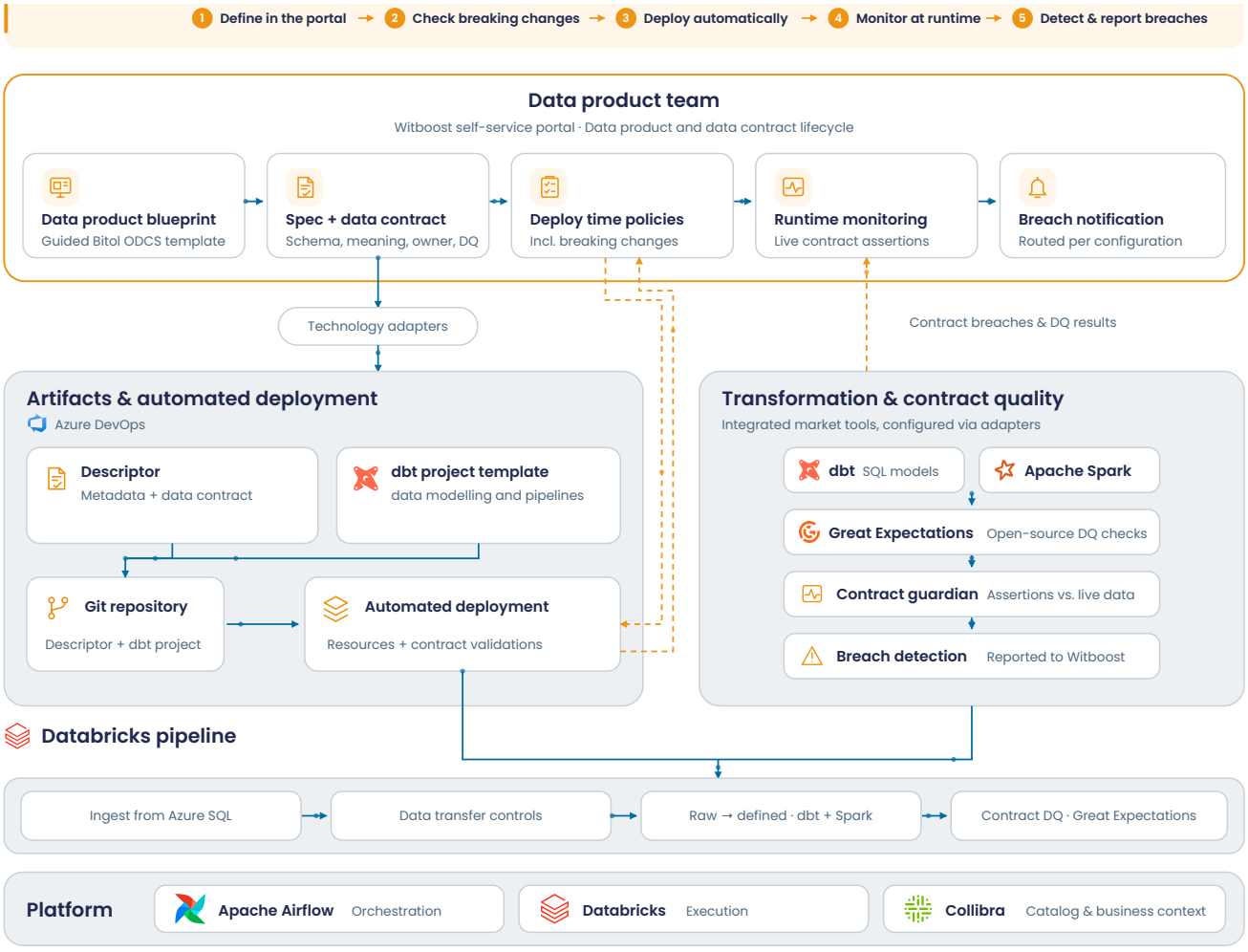
- ✓ Enable self-service capability for data producers
- ✓ Standardize data product development with declarative blueprints
- ✓ Embed computational governance using policy-as-code
- ✓ Realize the bank's Data Product Automation vision

Technical Methodology

To put this all into practice, the teams had to decide exactly how the architecture and processes for both data product blueprints and computational governance looked like.

What follows is a technical breakdown of the methodology, including the parts that were handled by Witboost.

Architecture Overview



Blue arrows: artifact and execution flow.
Dashed orange arrows: policy and data contract checks.

View the full-size architecture diagram

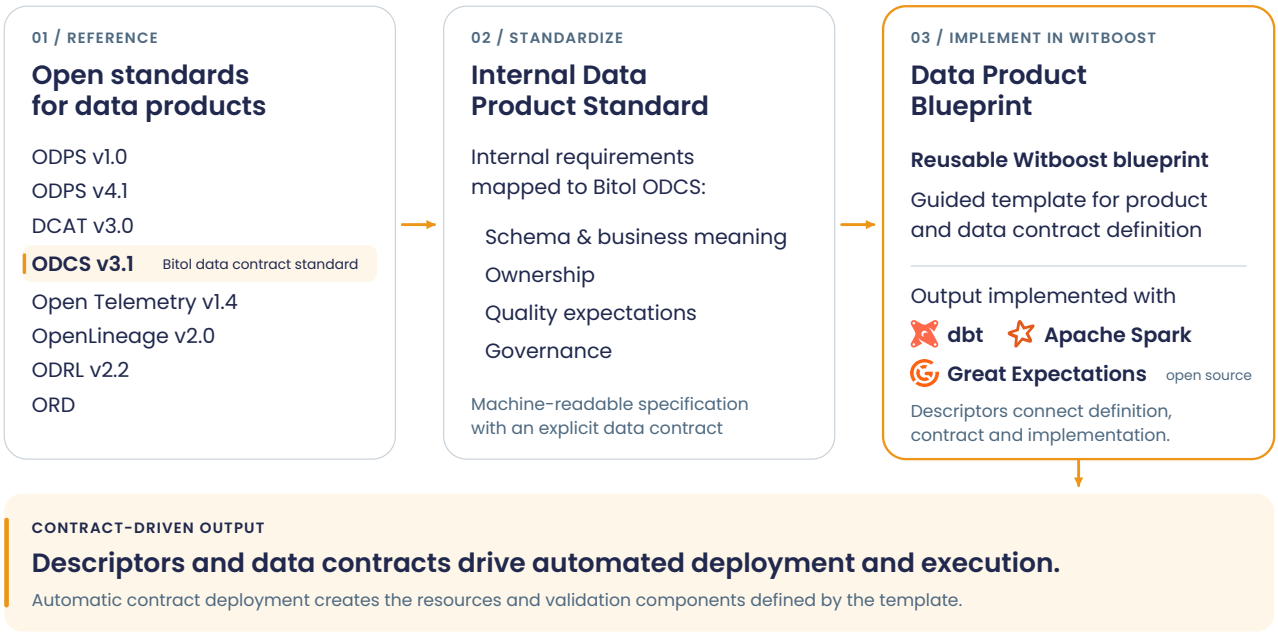
Architecture roles and responsibilities

The chosen architecture composed of multiple standards, platforms, and tools:

Layer	Responsibility and relevant technologies
Lifecycle and governance	Witboost : guided definition, product descriptors and data contracts, deployment orchestration, policy evaluation, and operational visibility.
Catalog and business context	Collibra : tha bank's existing catalog and governance context, complementing product and contract definitions.
Transformation and processing	dbt for SQL transformations and Apache Spark for distributed processing.
Orchestration and execution	Apache Airflow and Databricks: the execution technologies identified in the original case.
Provisioning and delivery	Technology adapters, version-controlled repositories, reusable deployment components, and CI/CD workflows: translating specifications into deployed.
Contract data quality validation	Open-source Great Expectations : executing configured data quality checks within the contract workflow.

> A data contract is an agreement between a producer and its consumers about the data exposed and the expectations attached to it.

Data Product Blueprint Realization Process



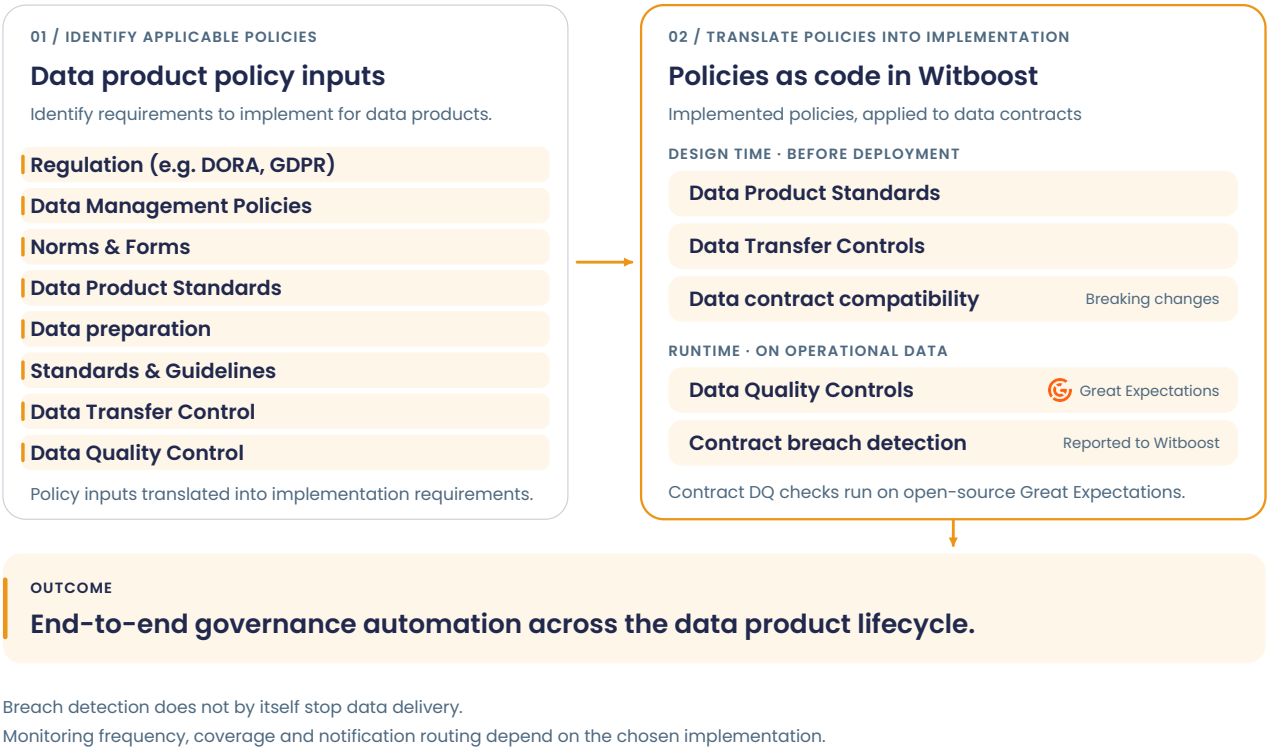
Reference standards do not imply every standard was implemented.
Witboost orchestrates the lifecycle; integrated tools provision, execute and validate.

The blueprint turned the bank's chosen development pattern into a guided workflow. Producers supplied metadata, technical configuration, and contract requirements; the resulting machine-readable descriptor supplied the inputs for deployment automation.

In Witboost, automatic contract deployment uses technology adapters and integrated delivery workflows to turn a specification into the resources and validation components defined by the chosen template. Deployment means creating the contract's implementation, not merely publishing its definition.

View the full-size blueprint diagram

Computational Governance Realization Process



Breach detection does not by itself stop data delivery.
Monitoring frequency, coverage and notification routing depend on the chosen implementation.

The data contract approach separates two checks.

Before deployment

Before deployment, policies evaluate the declared specification.

At runtime

At runtime, implemented assertions test operational data against contract requirements.

Within the context of data contracts and data quality, the team leveraged Great Expectations, the open-source data validation framework. Validation failures can be mapped to contract requirements and reported through computational governance as breaches.

Contract evolution adds another control point. Witboost supported configurable compatibility policies that compare a proposed contract with its previous version before deployment. Removing a field or changing its type incompatibly can be treated as a breaking change and prevented from progressing through the configured release workflow. A change in business meaning required owner review even when the schema stays the same.

Once a data product was running, a contract guardian evaluated implemented assertions against actual data and report breaches into Witboost. Monitoring frequency, coverage, and notification routing depended on the chosen implementation.

> Note: breach detection does not by itself mean that data delivery is automatically stopped.

Key Achievements

Using Witboost together with the team's internally developed and other vendor products, the European bank could fully automate their data product lifecycle with a metadata-driven approach.

The lifecycle automation achieved the following benefits:

01 Self-service data product definition – using blueprints and templates, the guided workflow within Witboost captured metadata, data contracts, data quality requirements, and technical configurations from the very start

02 Standardized data product descriptor and downstream automation – thanks to Witboost's machine-readable descriptor generation

03 Executable pipelines configurations and deployment artifacts – using an internally developed descriptor interpreter and translation into metadata specifications that trigger the workflow

04 Repeatable and compliant automated data product deployment – applying standardized repository structures, reusable components, testing frameworks, and CI/CD pipelines

05 Compliance, data quality, and operational standards were applied consistently – thanks to design-time and runtime computational policies enforced throughout the lifecycle

06 Ongoing visibility and governance along the operational lifecycle – with automated execution, policy monitoring, data quality validation, and alerting

> Witboost demonstrated the feasibility of a metadata-driven, self-service, and governed approach to Data Product development. This approach brought data contract definition into product delivery and gave data consumers a declared reference for assessing operational quality.

witboost

Bring automation to your data product lifecycle.

Want to see how Witboost can help you achieve your goals?

Explore Witboost →

Read the success case online
witboost.com/success-cases/european-retail-and-business-bank

