The Zero Repetition Bank Series

Auto-compliance and waste reduction in tech-driven enterprises

EY Technology Strategy & Transformation

September 2024

The better the question. The better the answer.
The better the world works.

EY

Shape the future with confidence

Contents

The Zero Repetition Bank

Auto-compliance and waste reduction in tech-driven enterprises

In recent decades, banks and other financial institutions have significantly transformed their distribution of products and services via digital channels and customer operations. This shift has led to an increasingly complex landscape of interconnected IT systems and supporting infrastructure.

Much of the work in the CIO domain, however, still largely depends on cumbersome manual processes for managing and enhancing the IT landscape. Tasks such as documenting system configurations, tracking changes, approving architectural modifications, and managing compliance typically involve extensive use of digital forms, checklists and approvals. This requires human actions at each step and introduces Repetition.

We have identified three main categories of repetition in infrastructure, data, and application management:

1. Risk Assessments

Involve periodic assessments and providing compliance evidence for both Non-Financial Risk and Financial Risk requirements. For instance, IT Security Assessments and Data Privacy Impact Assessments. These assessments are manual task-heavy, with DevOps teams often conducting several assessments simultaneously.

2. IT Change Management

The recurring and meticulous task of ensuring that designed configurations are up-to-date whenever new features are implemented or a technology stack undergoes lifecycle updates or enhancements. Similar configuration checks are necessary to identify the root causes of IT incidents which can slow down troubleshooting.

> Repetition - The unnecessary and repeated manual and cognitive efforts required to build, run, and change IT systems.

3. Mandatory Technology Replacement

Periodic replacement of platforms that support applications – for example, infrastructure and monitoring platforms - to keep up with the latest technical standards and prevent technical debt from piling up. Such platform migrations cause integration dependencies between teams and substantial non-value adding workload.

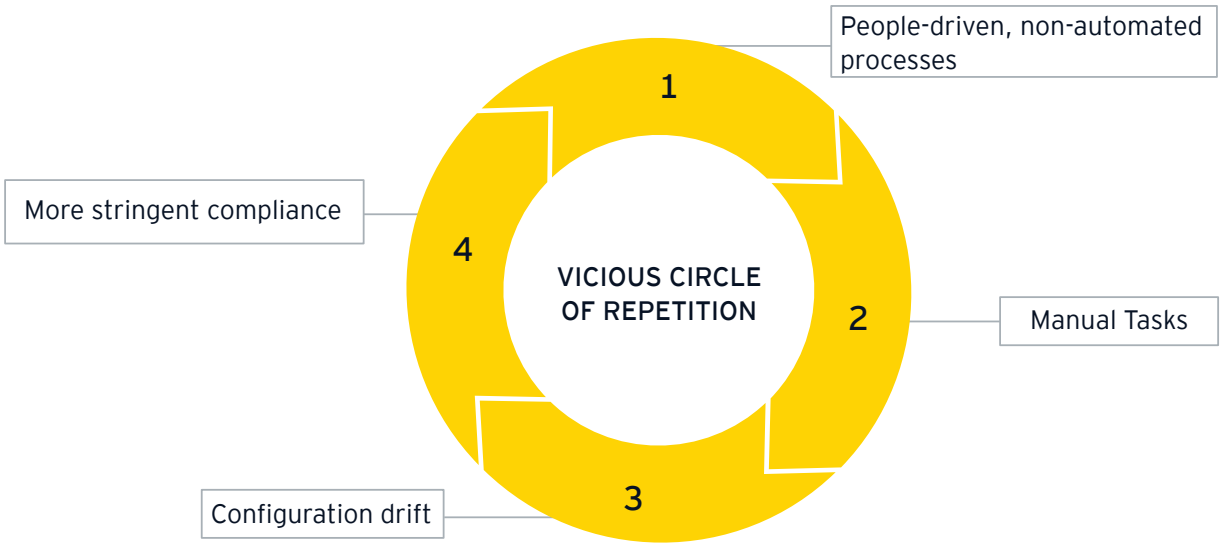


Figure 1: "Vicious Circle of Repetition"

Configuration drift

Configuration drift occurs when changes to systems or infrastructure are not properly documented or tracked over time. To prevent configuration drift, more stringent compliance processes are implemented. These demand significant amounts of manual work from both the DevOps teams and the risk function. This leads to the "Vicious Circle of Repetition": a self-perpetuating cycle where manual processes lead to inefficiencies, overburdened IT domains, reduced business agility, increased operational complexity, and heightened risks of errors. Our "Zero Repetition Bank Framework" is designed to help banks break free from this cycle.

50% of IT delivery budgets is wasted on repetition

We estimate that large financial institutions allocate 50% of their IT delivery budgets to repetitive tasks. These activities, predominantly non-value-adding, are deemed wasteful. Our assessment of a European bank that has a DevOps organization of 6,000 people identified **annual waste of over 200 million euros on repetition within infrastructure, data, and application management activities**; resources and talent not invested in change, innovation and improving the top- and bottom-line.

Integrating modern technological concepts presents a valuable opportunity to improve the top- and bottom line.

To meet the strategic expectations of the board, CIOs must overcome the barriers posed by Repetition. The combination of four key technological concepts offers a solution:

1. Automation
2. Everything-as-Code (EaC)
3. Datafication
4. Artificial Intelligence (AI)

Aligning these four concepts can be challenging without a unified strategic framework, which leads to fragmented implementations and siloed solutions, undermining the potential benefits of these advanced technologies.

The "Zero Repetition Bank Framework" assists CIOs in prioritizing projects with precision and applying Automation, EaC, Datafication, and AI consistently throughout their operations, yielding benefits such as improved operational efficiency, enhanced compliance posture, and increased business agility. This approach ultimately enables them to meet the ambitious revenue, cost, and innovation targets set by CEOs.

1 The Zero Repetition Bank Framework

1. The Zero Repetition Bank Framework

Key pointers

Four key concepts reinforce one another:

- Automation simplifies tasks
- Everything-as-Code ensures consistency and generates valuabledata
- Datafication turns operations into insightful information
- Artificial Intelligence uses the information to optimize processes

By combining these technological concepts smartly, banks create a **continuous improvement cycle** for IT operations, improving efficiency significantly.

The Zero Repetition Bank Framework aims to boost agility and resilience by implementing four cornerstone concepts. The combined adoption of these technological concepts not only enhances business agility and resilience but also streamlines operations and reduces operational costs. The framework encourages innovation and equips CIOs with the tools and capabilities necessary to pursue their strategic objectives and create enhanced value.

The Zero Repetition Bank Framework is built on extensive consultations and collaborative projects with leading banks. Our approach is rooted in real- world insights and experiences from various banks, each facing unique challenges and opportunities in their digital transformation.

Let us delve into the four key concepts and how they reinforce one another.

1. Automation

Automation uses technology to perform tasks without human intervention. By automating repetitive tasks, CIOs can save time and resources, allowing its employees to focus on more strategic activities. For example, pipelines automate steps like building, testing, and deploying software. CI/CD automates code integration and deployment to production. Tools like Jenkins, GitLab CI, and CircleCI help create and manage these pipelines, making the process repeatable and reliable.

Automation sets the foundation for Everything-as-Code (EaC) by standardizing repetitive tasks and creating a consistent environment for managing and operating code.

2. Everything-as-Code (EaC)

Everything-as-Code treats every aspect of the IT environment as code. This includes infrastructure, configurations, policies, and data management. By doing this, organizations can version, review, and automate all parts of IT operations, ensuring consistency, scalability, and repeatability.

Infrastructure-as-Code (IaC) is the practice of managing and provisioning computing infrastructure through machine-readable files, rather than through physical hardware configuration or interactive configuration tools. Tools like Terraform, Ansible, and CloudFormation are commonly used for IaC.

Policy-as-Code (PaC) involves defining and managing policies (such as security and compliance rules) through code. This ensures that policies are consistently applied and automatically updated across the infrastructure. For instance, security policies can be written as code and enforced automatically, reducing the risk of manual errors, and ensuring compliance with regulations. Open Policy Agent (OPA) is a well- known tool for PaC.

EaC builds on IaC and PaC by treating all aspects of IT operations, including configurations and data management, as code. This means that every element of the IT environment can be versioned, reviewed, and automated. By implementing Automation and EaC, organizations can codify and automate their IT environment. This produces valuable data, such as log files from automated processes, laying the groundwork for datafication.

3. Datafication

Datafication turns all aspects of IT operations into data, which can then be analyzed for insights and decision-making. This involves collecting data from various automated activities and IT systems. For example, results from automated tests, such as unit tests, integration tests, and end-to-end tests, are collected and stored. This data provides insights into the stability and performance of software.

With a robust Everything-as-Code (EaC) framework, datafication becomes even more powerful as every aspect of IT operations– configurations, policies, and the code itself–is managed through code. This leads to vast amounts of data on the IT operations of the bank. By having all this data, including the code, readily available, organizations can create more accurate AI models for IT operations. These models drive further optimization leading to continuous improvement in IT operations.

4. Artificial Intelligence (AI)

Artificial Intelligence allows computers to learn from data and make decisions without human intervention AI has made huge leaps in recent years and is set to have a massive impact on how organizations operate their IT domain.

For example: AI can analyze data from IT systems, such as server performance or network traffic, to spot unusual patterns. If it detects something abnormal, it can trigger alerts and help resolve issues quickly.

The Continuous Improvement Cycle

In conclusion, automation simplifies tasks and establishes the groundwork for Everything-as-Code, standardizing environments and processes. Everything-as-Code ensures consistency across all IT aspects and generates valuable data. This data facilitates datafication, turning operations into insightful information. Artificial Intelligence uses these insights to optimize processes and predict issues. AI-driven analytics uncover new automation opportunities, creating a continuous improvement cycle that makes IT environments more agile and efficient. By applying and combining these technologies intelligently, banks create a continuous improvement cycle for IT operations, improving efficiency significantly.

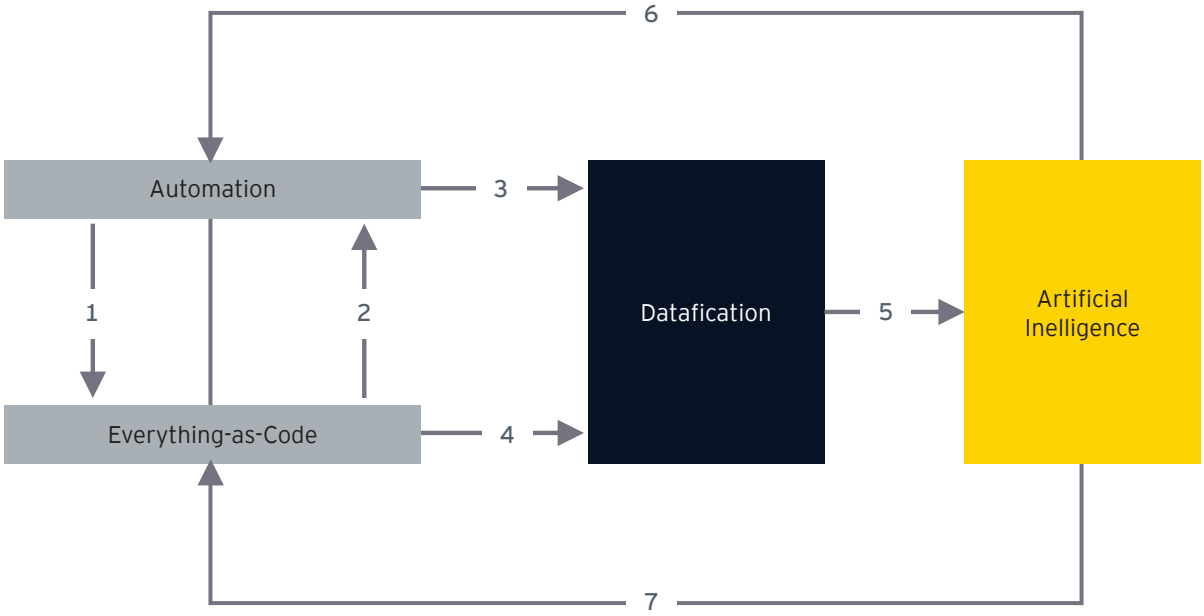
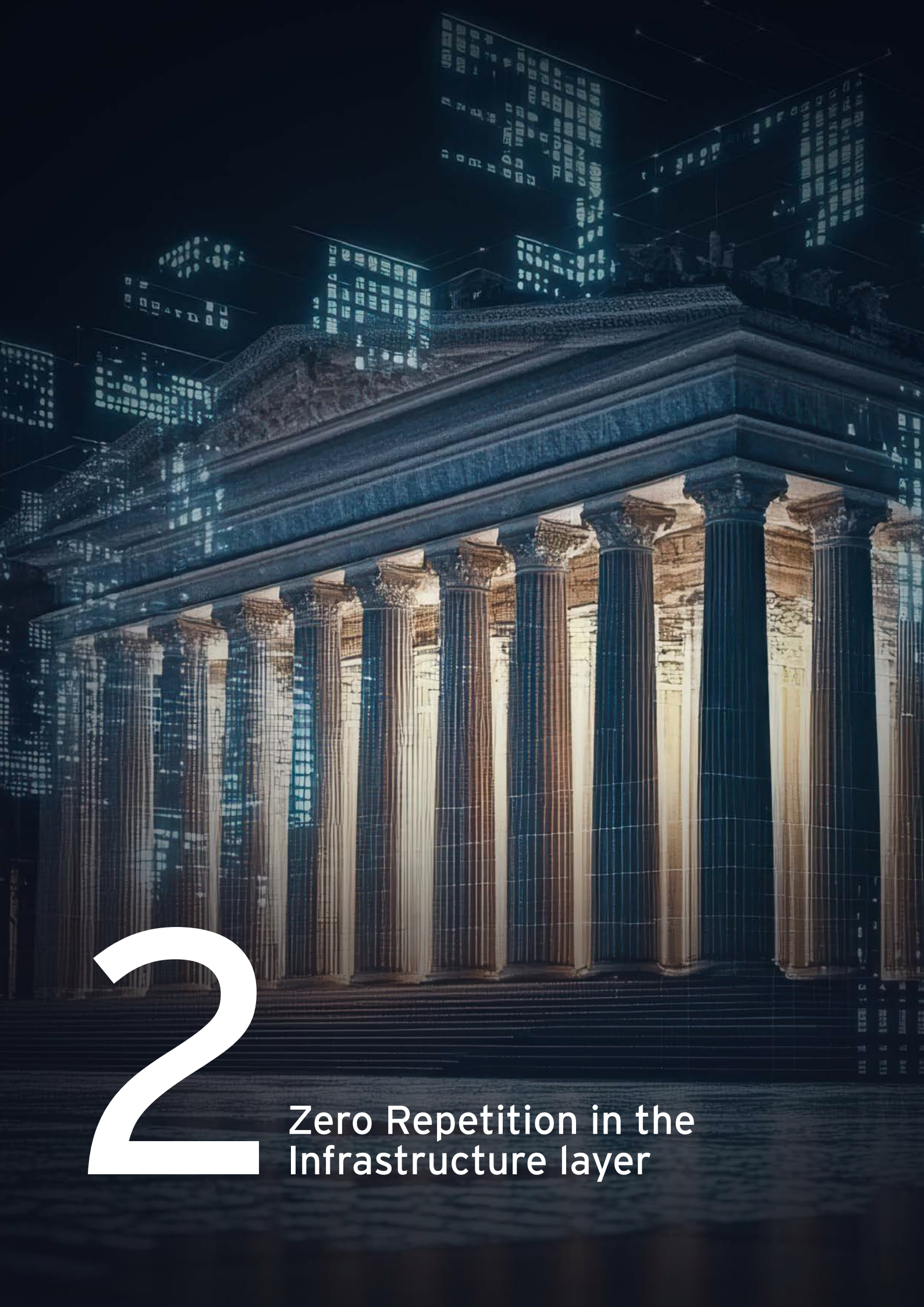


Figure 2: The Continuous Improvement Cycle



2 Zero Repetition in the Infrastructure layer

2. Zero Repetition in the Infrastructure layer

Key pointers

- Infrastructure is controlled through labor-intensive processes.
- Approval processes can take months to complete.
- Cloud technology enables a new approach by means of IaC and PaC.
- Usage of IaC and PaC streamlines the processes.

Banks can **efficiently** achieve a state of **auto-compliance** of their infrastructure both at the design and operational stages.

Infrastructure serves as the basis for the Zero Repetition Bank, providing a resilient and scalable foundation to extend the model towards the Data Flow layer and beyond. Infrastructure-as-Code (IaC) and Policy-as-Code (PaC) are instrumental in automating and optimizing resource management, ensuring both compliance and operational efficiency.

Present-day Infrastructure Operations

Until recently, most banks managed their infrastructure through traditional, manual processes.

DevOps teams would start by selecting the necessary infrastructure services to construct an infrastructure that aligns with the organization's risk appetite for their application. They then design the infrastructure, ensuring it meets the necessary performance, security, compliance, and financial requirements.

Once the design is complete, it undergoes a rigorous validation process. Policy guardians review the design extensively in what is known as the Test of Design. This review ensures that the proposed infrastructure aligns with organizational policies and risk appetite. This process can take months to complete.

Once the infrastructure is in use, DevOps teams must prove periodically that they are still compliant. Generating the evidence is labor-intensive and can drain resources. Manual reporting can result in inaccuracies and inconsistencies in compliance documentation, further complicating the compliance process.

Next-Gen Infrastructure Operations

To address the challenges of this lengthy, manual process, a new approach to working in the infrastructure layer has emerged, driven by the four key technology concepts of Automation, Everything-as-Code, Datafication, and AI.

Cloud Service Providers offer an abundance of technologies, including the latest advancements to enable the new approach. DevOps teams select from the available Infrastructure-as-Code (IaC) templates that best fit their requirements. These templates are pre-defined configurations that can be reused across different environments. This eliminates the variability and mistakes that often come with manual configurations. The IaC templates are automatically validated against all relevant policies using Policy-as-Code (PaC). Policy-as-Code (PaC) involves codifying policies into machine readable format, ensuring that every infrastructure deployment adheres to compliance standards and internal policies without the need for manual checks.

By integrating IaC and PaC in their risk frameworks and processes, organizations can achieve a state of auto-compliance, ensuring that their infrastructure is always aligned with regulatory requirements and internal policies, both at the design and operational stages.This reduces the burden of manual compliance checks and improves the overall compliance posture.

3 Zero Repetition in Data Flows

3. Zero Repetition in Data Flows

Key pointers

- Data Flows are core to orchestrate all data transactions across a bank's ecosystem.
- Currently, Data Flows are managed trough manual processes.
- Data Flows can also be automated and managed "as-code".

Automated compliance checks on Data Flows **simplify audit processes** and provide real-time assurance of adherence to policies.

In our Zero Repetition Bank Framework, Data Flows are a core component that ensures the seamless and compliant movement of data across the bank's IT ecosystem. Data Flows are designed to manage all data transactions between data providers and data consumers, maximizing efficiency and ensuring compliance with regulatory standards.

Present-day Data Flow Management

Currently Data Flow management remains heavily paper-based and people-driven, resulting in significant inefficiencies and operational risks. In the current setup, Data Flows are managed through a process similar to that of infrastructure, albeit with the additional challenge that Data Flow management consists of two sides, the data producer, and the data consumer. Data Flow validation is a manual validation process involving various stakeholders, including data officers, compliance teams and security experts. Each data flow, e.g. exchange of customer information, or transaction records, must be validated and approved. Both sides must periodically generate evidence to demonstrate that the Data Flows is compliant with the original design and organizational policies. This involves manually creating reports and proving adherence to latest standards, which is labor intensive, prone to inaccuracies, and get quickly outdated.

Next-Gen Data Flow Management

Building on the concepts of Infrastructure-as-Code, Data Flows can also be described and automated through "as-code" methodologies. This approach allows organizations to ensure that data movements are consistent, secure, and compliant with regulatory standards. The approach revolutionizing the way data is managed and transferred across systems.

When a data consumer team requires a new Data Flow, they begin by selecting a suitable template for Dataflow-as-Code, similar to IaC. Once the Dataflow-as-Code is configured, the data flow contract is automatically validated against all relevant policies using DataflowManagement-as-Code (DMaC), which is a specific PaC for Data Flows. The automation and standardization provided by Dataflow-as-Code eliminate the inefficiencies associated with manual Data Flow management. Automated compliance checks simplify audit processes and provide real-time assurance of adherence to policies, significantly enhancing compliance efficiency.

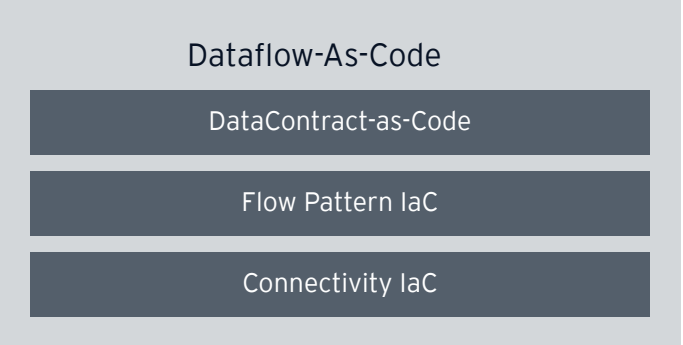


Figure 3: Dataflow-as-Code

DataContract-as-Code

DataContract-as-Code is a relatively new but essential formal agreement. Essentially, a Dataflow-as-Code specifies which data is shared and in what structure, and between what parties, ensuring clarity and consistency in data exchanges.

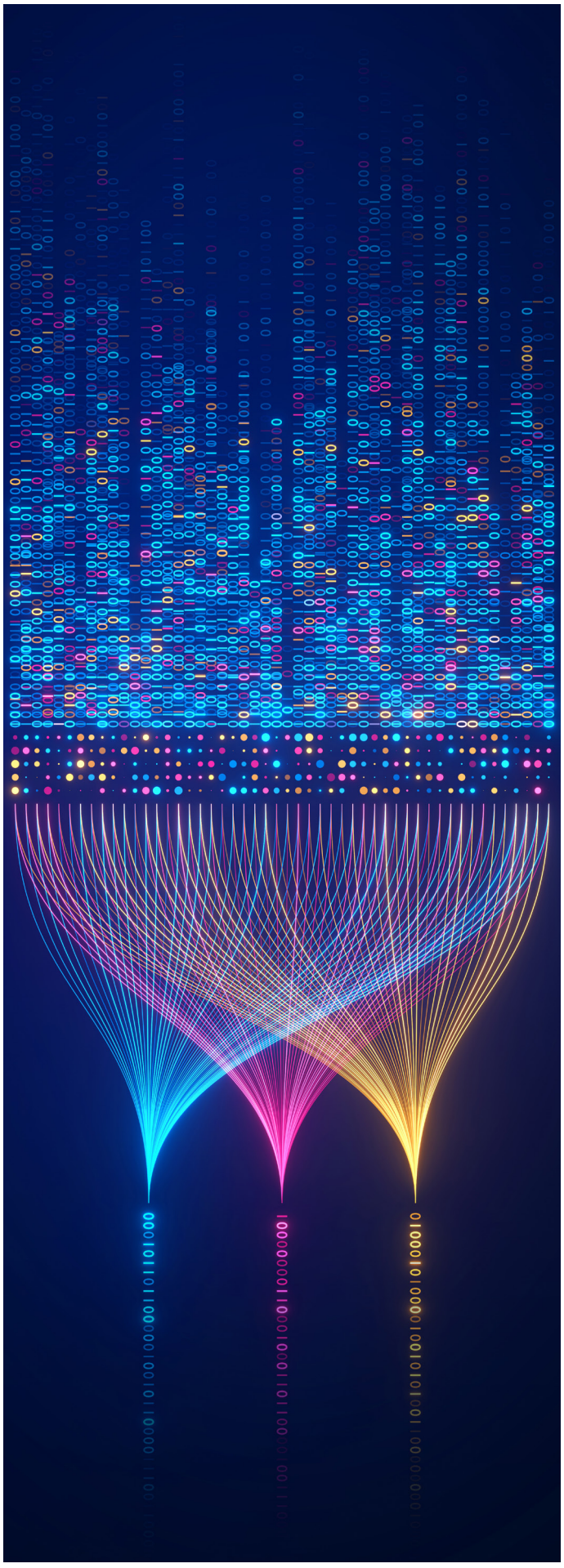
Flow Pattern IaC

Flow Patterns describe the methods by which data is moved between systems, such as file transfers, APIs, messaging systems, and streaming services. IaC provides a framework for automating these Data Flow configurations, ensuring they are consistently applied and managed. Configuring Data Flows involves setting up file transfers, defining API endpoints, configuring messaging queues, and establishing data streams.

Connectivity IaC

Connectivity encompasses the networks, protocols, and security measures that support Data Flows within an organization. IaC allows for precise and consistent configuration of these connectivity elements, ensuring that network setups are identical across different environments.

Using IaC, banks can automate the setup of virtual networks, subnets, and routing rules.



Key pointers

- Periodic risk assessments are a major repetition category.
- Covers compliance testing and evidencing against Non-Financial and Financial Risk policies and controls.
- Requires active knowledge and is heavy on manual tasks.

Our analysis of a European, with a DevOps organization of **6,000** people, indicates a conservative estimated annual spend of **30 million euros** on data and application related risk assessments.

Use Case 1

Repetitive Risk Assessments on Data Flows and applications

To remain in control of non-financial risks, e.g. operational, IT, data and security risks, and comply with regulatory requirements, e.g. DORA, banks have implemented an elaborate set of risk frameworks, control measures and policies.

DevOps teams need to periodically test and provide evidence of compliance to risk measures. DevOps teams are required to conduct different assessments, such as Data Privacy Impact Assessment (DPIA), Business/ Operations Impact Analysis (BIA/OIA), and Operational/ Information Systems Security Baseline (OSB/ISMS).

Typically, these assessments require several iterations to complete, to align views of the DevOps team, the application owner and second line risk functions.

Example: Annual cost breakdown for Data Privacy Impact Assessments

Under the EU General Data Protection Regulation (GDPR), banks are required to conduct a Data Privacy Impact Assessment (DPIA) periodically. This assessment is crucial for identifying, assessing, and mitigating or minimizing privacy risks associated with data processing activities. Each DPIA cycle necessitates that the DevOps team reviews the most recently approved DPIA for every IT asset. The team must account for any changes in the application since the last DPIA, drawing on active knowledge and conducting code-and documentation reviews.

They are tasked with identifying data usage, access rights, and procedures, as well as ensuring adherence to the latest DPIA template and process. Upon completion, the team then requests approval of the assessment outcomes from the Chief Data Officer (CDO) office. Based on our research, these assessments consume 50 to 60 hours per application, with a significant portion of time dedicated to manual tasks. The repetitive nature of this process follows from the periodic assessment requirement by the GDPR. Also, the assessments must be conducted at the application level, which leads to multiple teams simultaneously engaging in the same activity. For a bank with 1,000 applications, the annual cost attributed to DPIAs can reach 5 to 6 million euros.

Example: Total annual cost for Non-Financial Risk (NFR) and Financial Risk assessments

Beyond compliance to Non-Financial Risk requirements, banks must adhere to numerous financial (prudential) regulations such as CRD IV/CRR, LCR, NSFR, and MiFID II. Each regulation requires implementation of robust internal control frameworks. Periodically banks need to test and demonstrate to the supervisor that such controls are designed (ToD) and effective (ToE). In general, the type of tasks involved in Non-Financial Risk and Financial Risk assessments is highly comparable, as is the repetitive nature of these assessments. In our analysis of a European bank that has a DevOps organization of approximately 6,000 people, we conservatively estimated that in total 30 million euros is spent each year on risk assessments.

Key pointers

- Configuration drift and unknown data consumers complicate IT Change Management.
- To prevent negative change impact, DevOps teams spend significant time to identify all data consumers.
- Configuration drift also hampers root cause analysis and resolution of (major) incidents.
- With Dataflow-as-Code IT Change Management and troubleshooting goes significantly faster.

Real-world example: A single change to a core system with 100 data consumers required approximately 3,500 hours of change management effort; with Dataflows-as- Code, we estimate that **70% to 90% less time** would have been required.

Use Case 2

Repetition in IT Change Management and Incident Management

Changing IT applications involves more than just adding new features or making technical adjustments.

IT applications support end-to-end processing chains and may have numerous (tens or more) direct and indirect consumers.

To prevent negative change impact and ensure business continuity, it is crucial to identify all data consumers and associated Data Flows to be included in the change effort. Effective change management is often complicated by configuration drift and the blind spots that arise from unknown data consumers. Consequently, DevOps teams need to dedicate a significant amount of time to analyze their data consumers.

In our evaluation of a real-world change to the data model of a bank's core data management system – with over 100 data consumers – approximately 3,500 hours were spent on change management activities.

A large portion of this time was spent on the repetitive task of ensuring the designed configuration remains up-to- date and assessing potential change impact. In this example, we estimate that 70% to 90% less time could have been spent if Data Flows were managed as code.

Beyond complicating IT change management, the issue of configuration drift and unknown consumers also hampers effective incident management. Almost without exception, IT executives can share stories of spending numerous hours or even days in "war rooms" to determine the root cause of major incidents by meticulously plowing through trigger events in end-to-end processing chains.

Data Flows managed as code will significantly accelerate troubleshooting by providing instant and complete insights into the data providers and consumers connected to processing chains. Faster incident resolution reduces the impact on end- customers, minimizes drops in productivity, and can ultimately prevent bad press and reputational damage.

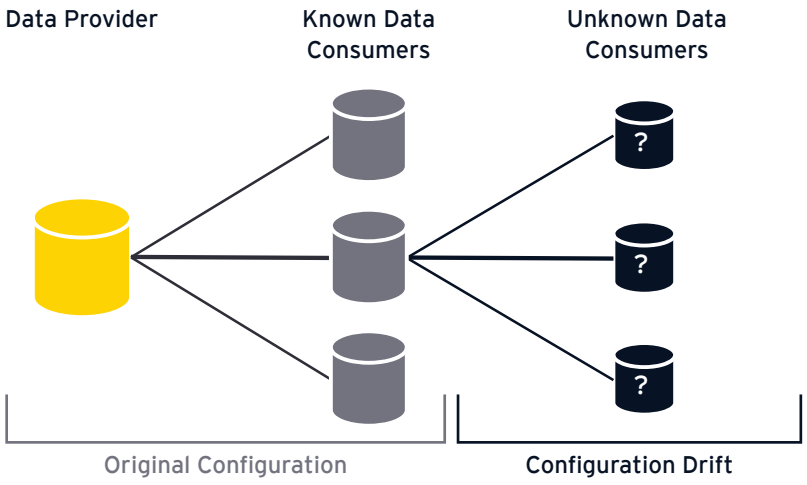
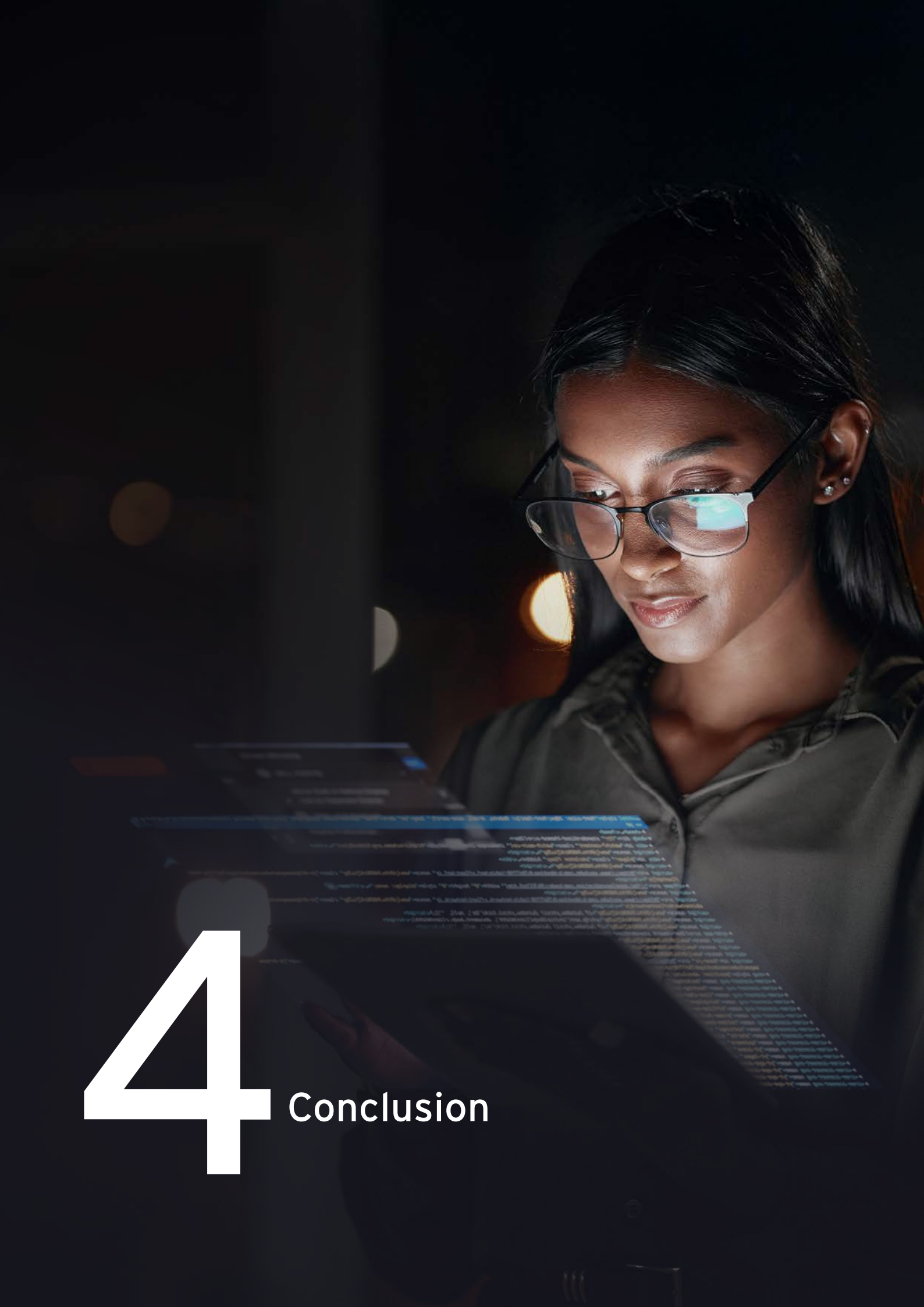


Figure 4: Drifting Data Flows complicate both IT Change- and Incident Management



4 Conclusion

4. Conclusion

To remain competitive, banks need to address their cost baseline. However, this is not an easy task, as much of the excess has already been trimmed by digitalizing customer operations and reducing headcounts in branches, call centers, and back offices.

Meanwhile, new (cyber) risks and regulatory requirements, such as DORA, demand more IT resources and capacity to be invested into staying secure and compliant.

Reducing repetition by smartly combining modern technologies will help banks lower their cost-to- income ratio and simultaneously improve their compliance posture.

We argue that failing to act now will put banks at a serious competitive disadvantage in a few years, from efficiency, agility, and innovation perspectives.

In this first paper, we covered how the Zero Repetition Bank Framework can be applied on the infrastructure layer and on the data layer.

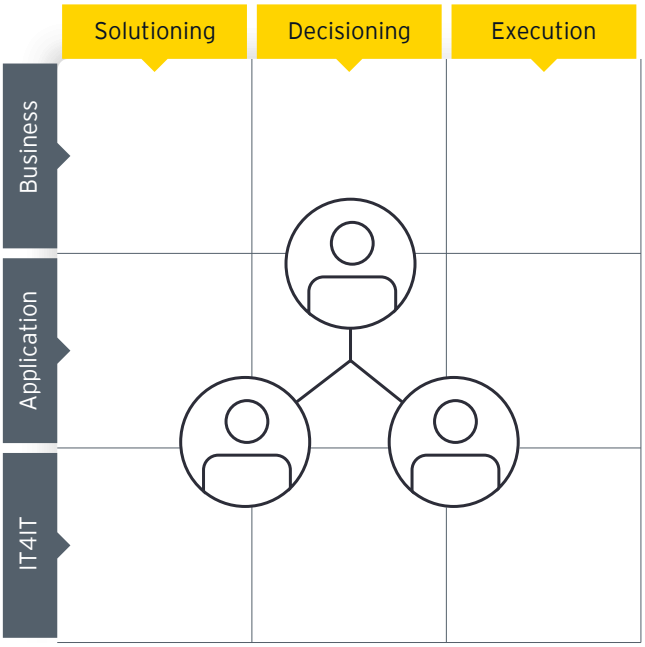
Applying the Zero Repetition Bank Framework on these two layers creates a solid foundation to extend the framework to the application and journey layer. Please follow us on LinkedIn to learn more on Zero Repetition Banking.

Imagine what it could mean for your bank, clients, and employees if you could truly focus on what matters most.

Our field of play

Experience in Cloud Strategy, Cloud Native Operating Models, Scaled Cloud Migration, CSP Vendor Selection, Data Architecture, Legacy Modernization and Mainframe Legacy Modernization and Mainframe Decommissioning.

Our multi-disciplinary approach



Meet our Technology Strategy & Transformation team

Remi Hesterman
Partner
EY Netherlands
E: remi.hesterman@nl.ey.com

Martin Borman
Strategic Solutions Architect
External advisor to EY
E: martin.borman@nl.ey.com

Matthieu Griffioen
Director
EY Netherlands
E: matthieu.griffioen@nl.ey.com

Sven Schenkel
Director
EY Netherlands
E: sven.schenkel@nl.ey.com

EY | Building a better working world

EY is building a better working world by creating new value for clients, people, society and the planet, while building trust in capital markets.

Enabled by data, AI and advanced technology, EY teams help clients shape the future with confidence and develop answers for the most pressing issues of today and tomorrow.

EY teams work across a full spectrum of services in assurance, consulting, tax, strategy and transactions. Fueled by sector insights, a globally connected, multidisciplinary network and diverse ecosystem partners, EY teams can provide services in more than 150 countries and territories.

All in to shape the future with confidence.

EY refers to the global organization, and may refer to one or more, of the member firms of Ernst & Young Global Limited, each of which is a separate legal entity. Ernst & Young Global Limited, a UK company limited by guarantee, does not provide services to clients. Information about how EY collects and uses personal data and a description of the rights individuals have under data protection legislation are available via ey.com/privacy. EY member firms do not practice law where prohibited by local laws. For more information about our organization, please visit ey.com.

© 2025 EY Nederland B.V.
All Rights Reserved.

ED NONE

ey.com