Enterprise Strategy Group

CASE STUDY

Banking on Automation: How ARIS Transformed Application Deployment at a Major Financial Institution

Torsten Volk, Principal Analyst

This case study examines how a major North American financial institution with more than 2,500 branches transformed its application deployment process using ARIS. Previously burdened by complex manual compliance reviews across 1,800 enterprise applications, the bank implemented a three-phase solution: converting static documentation to data models, automating business logic, and implementing straight-through processing (this third phase is currently underway). The results were significant: compliance review time reduced from a 5-day SLA window to being nearly instantaneous, senior architect time allocation decreased from 30% to under 5%, and deployment error rates improved from 11% to 1%. This transformation enhanced speed, reduced costs, improved compliance, and provided clear visibility across the financial institution's application portfolio.

Technology Provider
ARIS

Customer
Major North American Financial Institution

Industry
Financial Services

Company Size
Approximately 55,000

Country
USA

Introduction

A North American financial institution with over 2,500 branches, approximately 55,000 employees, and nearly 10,000 ATMs offers retail and business banking services to millions of customers. The bank maintains a portfolio of approximately 1,800 enterprise applications to support its customer-facing services across multiple channels (mobile, web, and branch), its back-office operations (including loan processing and risk management), and its core banking platforms running alongside newer digital initiatives. Operating in the highly regulated financial sector, while competing on the ability to rapidly offer new products and services to a large and diverse customer base, this financial institution needs to follow a rigid process when deploying new applications to production or making changes to existing applications in production.

Situation Analysis

The application deployment process at this financial institution consists of three distinct stages that apply to minor changes to existing application environments as well as to the deployment of new applications:

- **Phase 1: Design Review**

  Before ARIS, the design review process relied on static documentation created in Visio. Approximately 20 senior architects and engineers (dedicating roughly 30% of their time) would manually examine these documents to verify compliance with architectural standards and security requirements. This team operated under a 5-day service-level agreement (SLA) for the initial review of each design submission. If compliance issues were identified, the design would be returned to the originator for corrections, requiring a restart of the 5-day review cycle, potentially adding days or weeks to the approval process. This manual approach led to inconsistent application of standards and missed requirements due to the complexity caused by the high amount of interdependencies among the bank's 1,800 applications.

- **Phase 2: Design-to-provisioning Translation**

  Once a design received approval, a different team would translate the approved design documentation into provisioning specifications. This "swivel chair" process involved staff visually examining the static design documents and manually entering the corresponding infrastructure requirements into the provisioning systems. Team members would interpret the design specifications and select appropriate components (e.g., operating systems, databases, web servers, etc.) from menu lists in systems like Morpheus or Ansible. This translation step was entirely manual, with no automated data transfer between the design approval system and the provisioning tools, creating opportunities for misinterpretation, transcription errors, and overlooked interdependencies.

- **Phase 3: Provisioning and Verification**

  The final phase involved the actual deployment and verification of infrastructure by a third team who would execute the provisioning based on the specifications created in phase two. Using tools like Ansible or Morpheus, this team would configure and deploy the various infrastructure components according to the translated requirements. Following deployment, the team would perform verification checks, comparing the actual deployed environment against the original approved design to ensure compliance. This verification was necessary due to the disconnected nature of the earlier processes and the potential for errors in manual translation. When discrepancies were found, remediation efforts would begin, further extending the overall implementation timeline for new applications or changes.

In a nutshell, the cost, duration, and risk of the manual review process of each individual set of production changes hinder the bank's ability to innovate quickly, create unnecessary operational expenses, and introduce significant hidden risks that could potentially affect customers and regulatory compliance. This inefficient process not only drains valuable technical resources that could be better deployed on strategic initiatives but also extends time to market for new features and services, ultimately affecting competitive positioning in the rapidly evolving and highly competitive financial services marketplace.

The ARIS Solution

The bank mentions four key reasons for their adoption of ARIS:

1. **Speed:** Faster provisioning of new applications and application changes.
2. **Compliance:** Lower deployment risk through improved accuracy.
3. **Cost:** Staff time saved through automation.
4. **Visibility:** Instant portfolio-wide impact analysis.

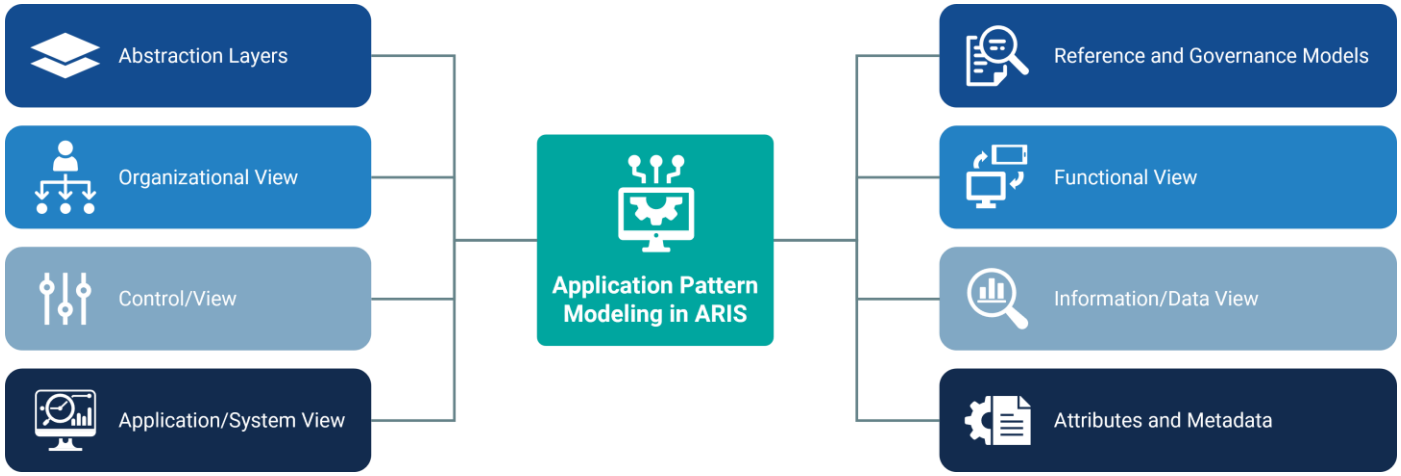
The bank implemented ARIS in three stages (stage 3 is still in progress):

- **Phase 1: Converting Static Documentation to Data Models**

  This phase involved converting all the static documents and images that are part of an application pattern into data models in ARIS. This conversion took approximately 12 months and was completed first. All 1,800 applications are now modeled in ARIS using the ArchiMate modeling language, allowing for queries, analytics, and visualization of the application portfolio. For example, reviewers of a change request can now precisely determine the overall effect of this change on the rest of the 1,800 application environments upfront or instantly determine which applications are affected by specific technology changes, such as database upgrades or the use of a different hypervisor.

> **"If we see most applications of a particular type are using a repeating sort of stack of technology, then we might ask where there are outliers, if those outliers have a reason to be different, or if maybe there are examples of something we want to bring closer to the normal sort of usage."**
>
> – Senior Vice President, Enterprise Architecture-Business & Data Technology, Major Financial Institution

Figure 1. Key Elements of Application Pattern Modeling in ARIS



Source: Enterprise Strategy Group, now part of Omdia

- **Phase 2: Automating Business Logic**

  The second phase involved automating the evaluation of designs by coding all business rules and requirements into ARIS. This phase took approximately 10-11 months and is also complete. The automation of business logic is what enables the near-instantaneous compliance checks that replaced the previous 5-day SLA review process.

Figure 2. Key Elements of Application Pattern Modeling in ARIS

Source: Enterprise Strategy Group, now part of Omdia

- **Phase 3: Straight-through Processing (In Progress)**

  The financial institution is now in the process of automating the provisioning phase of their deployment process where ARIS will directly connect to the provisioning systems, automating the translation between approved designs and provisioning specifications.

Outcomes

The financial institution reported achieving many benefits from ARIS that align with their reasons for adopting the solution:

1. **Speed**
   - Reduced compliance review time from a 5-day SLA to near-instantaneous automated checks.
   - Eliminated multiple review cycles previously required when issues were found.
   - Enabled immediate feedback on design issues, allowing for quick corrections and re-validation.

> **"FTEs were probably spending 30% of their time dedicated to this [reviewing application changes], and this [ARIS] will take it down to probably less than 5% of their time."**
>
> – Senior Vice President, Enterprise Architecture-Business & Data Technology, Major Financial Institution

2. **Compliance**
   - Achieved 60%-70% reduction in staff time spent on regulatory compliance reporting and auditing.
   - Identified 300 instances where a specific requirement pattern was not correctly applied across the 1,800 applications.
   - Discovered and corrected compliance checks that weren't consistently performed in the manual process.
   - Ensured consistent application of all 42 architectural compliance patterns.
   - Improved standardization of technology stacks by identifying outliers and exceptions.
   - Enhanced audit readiness by making compliance documentation instantly available.

> **"The other step that it'll allow us to do is, as we start laying over the business architecture, instead of just saying what systems, I could say what business processes might be impacted."**
>
> - Senior Vice President, Enterprise Architecture-Business & Data Technology, Major Financial Institution

3. **Cost**
   - Reduced senior architect and engineer time allocation from 30% to less than 5% across approximately 20 FTEs.
   - Freed up highly skilled talent to focus on more strategic architectural work rather than repetitive reviews.
   - Eliminated redundant verification steps by ensuring compliance earlier in the process.
   - Decreased the number of testing iterations by lowering failure rates from approximately 11% to 1%.
4. **Visibility**
   - Transformed static documentation into data models for all 1,800 applications.
   - Enabled portfolio-wide analysis that was previously impossible with document-based architecture,
   - Provided clear visualization of dependencies between applications and systems.
   - Supported better strategic planning through improved understanding of architectural patterns.
   - Created the foundation for connecting technical architecture to business processes in future phases.

Conclusion

By implementing ARIS, the financial institution converted static documentation into an interconnected repository of architectural knowledge. Key metrics show remarkable improvements, such as time taken for the compliance test cut from 5 days to being nearly instantaneous. Additionally, resource allocation for reviews dropped from 30% to less than 5% of staff time, and compliance verification improved from an error rate of 11% to 1%. The three-phase implementation approach (document conversion, rule automation, and straight-through processing) enabled systematic transformation of their processes while maintaining operational continuity. The resulting relational repository of application designs now provides dynamic visibility into architectural dependencies, enables data-informed decision-making, and significantly reduces risk.

> **"Let's say I test and have a failure rate of 11%... Instead, maybe I have a failure rate of 1% and maybe it's tied to things that were kind of in process and not yet updated."**
>
> – Senior Vice President, Enterprise Architecture - Business & Data Technology, Major Financial Institution

Lessons for Other Companies and Industries

The lessons from this implementation are widely applicable across industries. First, organizations should prioritize data-driven architecture management through interconnected repositories over isolated documentation to enable dynamic analytics and automated compliance. Second, a phased implementation approach (beginning with model conversion before rule automation) provides manageable change while delivering incremental benefits. Third, standardizing architecture in a relational repository is challenging but foundational, as converting diverse document formats required a significant 12-month effort but created the interconnected environment necessary for automation. Fourth, automation should target high-value, repetitive tasks performed by skilled professionals to maximize ROI. Finally, compliance automation through relational repositories is particularly valuable in regulated industries beyond banking, including healthcare, utilities, and government sectors.

Importance of ARIS for Organizations With Similar Challenges

For organizations facing similar challenges with complex application portfolios and regulatory requirements, ARIS offers a transformative solution beyond just "a place to put your stuff." It serves as an interconnected home for business and architecture data, where the relationships between components create unique value. The platform's ability to model architectures using standardized notation (ArchiMate) within a relational database provides the foundation for automated governance and dynamic analytics.

The repository makes architectural data accessible to decision-makers through dynamic reporting, providing critical insights on impact analysis, compliance conformance, value-of-change assessment, and correlations between architectural elements. This interconnected approach ensures consistent application of standards while reducing dependency on manual reviews. ARIS provides particular value where organizations need to demonstrate compliance, manage large application portfolios, or assess the impact of architectural changes within an integrated business context.