Build Once, Reuse Everywhere

How Saudi Arabia's leading Shariah-compliant bank turned change from invention into assembly

100+
canonical services and APIs in the catalogue

20–40%
faster from architecture intake to design sign-off

60–80%
less time preparing audit evidence for SAMA (Saudi Central Bank)

The Challenge

Bank Albilad serves retail, corporate, treasury and SME customers across the Kingdom of Saudi Arabia through mobile, internet, branches and a contact centre – plus a growing ecosystem of partners and fintechs. Behind that omnichannel promise sat an architecture that had grown one project at a time.

Services were named differently in different domains. The same capability existed more than once. Integration ran on point-to-point contracts negotiated per project, so every new partner meant starting over. Delivery cycles were long because reviews ran sequentially, risk findings arrived late, and preparing evidence for a SAMA (Saudi Central Bank) audit was manual work measured in days.

None of this was a crisis. That is precisely what made it hard to fix: each individual shortcut had been reasonable, and together they had become the cost of doing anything new.

The Solution

In September 2023 the Enterprise Architecture Office began a bank-wide adoption of the BIAN Service Landscape (v12) as the shared reference for business capabilities and service domains. Production followed in March 2025.

The approach was business-first, not tool-first. Products were mapped to BIAN service domains, then onward to business flows, applications, data entities and infrastructure – one traceable chain. ABACUS became the system of record, integrated with ServiceNow CMDB for runtime alignment and Erwin for data lineage. Governance moved into the CI/CD pipeline: API linting, contract tests and security scans run on every change – controls shifted left, so architectural decisions are enforced in the build rather than reviewed after the fact.

Starting Where It Counted

Rather than cataloguing the whole bank, the team began with four high-value journeys – payments, cards, financing and onboarding. Each one was mapped to BIAN service domains, given canonical APIs and data definitions, and put through architecture governance until reuse was the path of least resistance.

Once pilot squads, building in parallel against pattern-based designs instead of queueing for sequential review, showed shorter cycles and fewer integration defects, the model scaled horizontally across domains and out to partners. Legacy systems were modernised progressively around BIAN domains – strangler pattern, nothing switched off in one move.

Why BIAN, Not the Frameworks Already in Use

Bank Albilad already ran a mature EA practice, so the question was not whether to standardise but on what. TOGAF and ArchiMate govern method and notation, but neither is a banking service reference. ISO 20022 standardises payments messaging, not a service landscape. APQC and eTOM classify processes in other industries; FIBO describes financial instruments as an ontology rather than something to build from. BIAN was chosen because it is banking-specific and translates directly into reusable domain services – and because it coexists with the rest rather than replacing it.

What Proved Harder Than Expected

The bank names 5 obstacles, each with the answer it arrived at:

BIAN's words against local product names

Aligning the standard's vocabulary with what products are actually called inside the bank. The answer: a "BIAN-to-Albilad" glossary, with deltas kept deliberately small so standardisation survived the translation.

The risk of cataloguing everything

A service landscape can absorb unlimited effort. The answer: high-value journeys first, coverage expanded iteratively rather than mapped exhaustively up front.

Getting partners onto canonical contracts

Vendors had little reason to change their interfaces. The answer: BIAN-aligned APIs were written into onboarding policy and shipped with reference implementation kits.

Teaching a bank to think in domains

Squads had to learn BIAN, API-first design and domain thinking first. The answer was structural rather than documentary: bootcamps, brown-bag sessions and an EA coach embedded in each domain.

Data semantics under PDPL and NDMO

Canonical definitions had to carry consent and retention attributes to satisfy Saudi data regulation, with lineage tracked in ABACUS and Erwin.

Each obstacle was met with a decision rather than a workaround – which is why the model survived contact with four domains and an external partner ecosystem.

Impact & Results

20–40%

Faster from architecture intake to design sign-off

100+

Canonical services and APIs, with duplicate services retired

15% to 20%

Reuse rate across services and APIs (approximate)

60–80%

Less time preparing audit evidence for SAMA ITGF and NDMO

> *By standardising on BIAN service domains and canonical APIs in ABACUS, we have converted fragmented, project-specific work into reusable building blocks – accelerating time-to-market, cutting integration defects, and embedding compliance by design.*
>
> – Bank Albilad, BIAN Transformation Awards 2025 submission

What's Next

The canonical catalogue, API specifications, patterns and test kits are open to every squad – some 2,000 people across business and IT – and communities of practice evolve the standards rather than freeze them. The bank calls the programme recently started, intends to engage BIAN directly, and will share its governance playbooks with peers.

The Takeaway

Bank Albilad summarises its own programme in four words: build once, reuse everywhere. What changed is not that the bank moved faster on any single project, but that a new product no longer starts with invention. It starts with assembly, the audit trail comes with it – and the capacity freed up by not rebuilding the same thing twice has moved from running the bank to changing it.

Every case study in this series began as a submission form. The BIAN Transformation Awards 2026 are open now. Entries close on 16 March 2027 – find the submission area at bian-services.com

This case study is based on Bank Albilad's submission to the BIAN Transformation Awards 2025, in the "Transformation Champion" category.