

USE CASE

Test a design change against a mirror first

Model a proposed change against a mirror of your connected estate and see cost, dependencies, downtime risk and control status before anything moves.

THE PROBLEM

The design review asks what a migration costs and what it breaks, and the answer is a spreadsheet nobody trusts. Controls that passed before a change are assumed to pass after it. That is how a migration quietly loses encryption.

HOW ARIBOT HANDLES IT

- 1 Mirror the estate**
Connected accounts produce a twin of what you run.
- 2 Propose a change**
Stage a design decision, such as moving an API to a managed runtime, with nothing applied.
- 3 Simulate the impact**
See the monthly cost change, the dependencies affected and the downtime risk.
- 4 Re-evaluate controls**
Controls are re-checked against the changed model, and any without a result stay not evaluated.

WHAT YOU SEE IN THE DEMO

- A Simulated impact panel with monthly cost change, dependencies affected and controls re-evaluated
- A control status table with rows such as encryption in transit and backup restore tested, including Not evaluated
- A topology of gateway, firewall, services and databases checked node by node
- Pass, warning and fail markers on each component

WHO IT IS FOR

- Enterprise Architect
- Head of Platform Engineering
- Cloud Security Architect

WHAT YOU GET

- Cost and dependency impact before a change ships
- Control status re-evaluated on the changed design
- Scan results that outrank manual sign-offs
- An architecture kept in step with the live cloud, with drift shown

See it: ayurak.com/demos/digital-twin

€30 per scan · €30 per applied fix (dry runs free)