BMDIGITALS

Reliability & Resilience

Reference Architecture

This brief outlines a reusable reference pattern for discovery, architecture, implementation, and operational rollout.

Problem

Critical platforms need clear visibility into health, dependencies, latency, error patterns, incidents, recovery posture, and resilience objectives.

A command center helps teams understand risk and coordinate recovery with evidence.

Typical Architecture

Client traffic reaches gateway and application services backed by dependencies, databases, messaging, cache, observability, and recovery automation.

Components

Service map, health model, SLO views, incident states, RTO/RPO indicators, failure simulations, recovery runs, and post-incident records.

Security Considerations

Protect operational telemetry, restrict incident controls, audit recovery actions, and isolate production credentials from diagnostic views.

Integration Pattern

Read observability and platform signals through least-privilege integrations before adding controlled automation for repeatable recovery steps.

Implementation Approach

Map dependencies, define SLO and recovery targets, build health views, simulate failures, validate playbooks, and iterate after real incidents.

Operational KPIs

Availability, latency, error rate, SLO burn, incident count, mean time to detect, mean time to recover, RTO, and RPO.

Deployment Model

A secure internal command surface connected to observability systems, incident tools, and controlled automation endpoints.

BMDigitals Engagement Model

Reliability assessment, dependency mapping, SLO design, command center build, resilience testing, and operational coaching.

Reference architecture. Final implementation depends on the client's operational and technical environment.