# Deloitte DevOps & Cloud Strategy Guide

## CI/CD for Data Analytics Workloads

Designing CI/CD for Long-Running Jobs

- Use event-driven orchestration (e.g., Apache Airflow, Argo Workflows) integrated with GitOps.

- Implement checkpointing and job state persistence (e.g., via S3 or Azure Blob) to resume failed jobs.

- Use dedicated compute queues (e.g., AWS Batch, Azure Synapse Pipelines) with autoscaling.

Pipeline-as-Code Across 30+ Repos

- Centralize reusable pipeline templates using GitHub Actions reusable workflows, GitLab includes, or Azure DevOps YAML templates.

- Use monorepo or template sync tools (e.g., repo-sync, cookiecutter, templater) to propagate updates.

- Maintain a pipeline governance repo with versioned templates and audit policies.

Auto-Healing Strategies

- Implement retry policies with exponential backoff.

- Use fallback stages (e.g., alternate data sources or compute environments).

- Integrate failure detection hooks to trigger reruns or notify SRE teams via Slack/Teams.

Audit Logging for SOX/GDPR

- Use immutable logging (e.g., AWS CloudTrail, Azure Monitor) for pipeline events.

- Store logs in WORM-compliant storage (Write Once Read Many).

- Integrate with SIEM tools (Splunk, Sentinel) and enforce access control policies via IAM.

## Cloud & Infra (Advanced Scenarios)

Hybrid Network Segmentation (On-Prem + Azure)

- Use Azure Virtual WAN with Hub-Spoke topology.

- Segment workloads using Network Security Groups (NSGs) and Application Security Groups (ASGs).

- Implement ExpressRoute with private peering and firewall appliances (e.g., Palo Alto, Fortinet).

Cross-Account IAM in AWS

- Use IAM roles with external trust policies and STS AssumeRole.

- Implement permission boundaries and resource-based policies.

- Use AWS Organizations SCPs to enforce guardrails.

Resource Tagging Automation

- Enforce tagging via service control policies (SCPs) and tag policies.

- Use tools like Cloud Custodian, Terraform tagging modules, or Azure Policy.

- Integrate tagging into CI/CD pipelines and IaC templates.

Preventing Shadow IT

- Use resource creation policies and approval workflows (e.g., ServiceNow + Terraform Cloud).

- Monitor with CloudTrail, Azure Activity Logs, and GuardDuty.

- Implement budget alerts and automated cleanup scripts for untagged resources.

## Kubernetes & Containers

Node Auto-Scaling Without Disruption

- Use Pod Disruption Budgets (PDBs) and priority classes.

- Schedule scaling during low-traffic windows or use predictive scaling.

- Prefer Karpenter for faster provisioning and bin-packing efficiency.

Noisy Neighbor Detection

- Monitor with cAdvisor, Prometheus, and Kube-state-metrics.

- Use resource quotas, limit ranges, and vertical pod autoscaler.

- Isolate workloads using taints/tolerations and dedicated node pools.

Private Container Registries

- Use ECR with cross-account access, Azure Container Registry with RBAC, or Harbor with OIDC.

- Enforce image signing (e.g., Cosign, Notary v2) and vulnerability scanning.

- Use network policies to restrict registry access.

Karpenter vs Cluster Autoscaler

- Prefer Karpenter for dynamic instance types, faster scale-up, and spot instance support.

- Use Cluster Autoscaler when tight control over node groups is needed or using managed node pools.

## Security & Compliance

Role-Based Boundaries in Terraform/Bicep

- Use Terraform workspaces and RBAC modules.

- Implement OPA policies via Sentinel or Conftest.

- Use Azure Blueprints or Terraform Sentinel for compliance enforcement.

Securing Drift Detection Logs

- Store logs in immutable storage with access logging.

- Use versioned state files and audit trails in Terraform Cloud or Atlantis.

- Integrate with SIEM and alerting systems.

Zero-Trust in Kubernetes

- Use Istio or Linkerd for mTLS and policy enforcement.

- Implement network policies, RBAC, and OPA Gatekeeper.

- Use identity-aware proxies and service mesh authorization policies.

Supply Chain Attack Mitigation

- Use SBOMs (Software Bill of Materials) via Syft.

- Enforce image signing and verification.

- Scan dependencies with Snyk, Trivy, or Grype.

- Use CI/CD pipeline attestation (e.g., Sigstore, in-toto).

## Resilience & Incident Handling

Chaos Testing on AKS

- Use Azure Chaos Studio or LitmusChaos.

- Simulate pod failures, node crashes, and network partitions.

- Automate via GitHub Actions or Azure DevOps pipelines.

Multi-Cloud Failover Simulation

- Use DNS failover (e.g., Route 53 + Azure Traffic Manager).

- Replicate workloads using Terraform multi-provider modules.

- Test with active-active or active-passive setups.

Critical Incident Handling

- Use runbooks, incident response playbooks, and SlackOps integrations.

- Implement automated rollback pipelines and feature flags.

- Use real-time dashboards (e.g., Grafana + Loki + Tempo).

Automated Compliance Reporting

- Integrate post-mortem templates with tools like Jira or Confluence.

- Use CI/CD hooks to generate reports from logs and metrics.

- Store reports in compliant repositories (e.g., SharePoint, GRC tools).