Computer Science and Engineering 2025, 15(1): 22-35
DOI: 10.5923/j.computer.20251501.03

# Enterprise-Scale Cloud Migration Factory Model: A Standardized and Value-Driven Approach for Large-Scale Applications

**Avitesh Kesharwani[1,*], Lokesh Manker[2], Amit Rajput[3]**

[1]Sr. Principal Consultant, Cloud Data, Tech and AI, Genpact, Concord, NC, USA
[2]Content Management Engineer, Fidelity Investment, Apex, NC, USA
[3]Principal Consultant, Data, Tech and AI, Genpact, Concord, NC, USA

**Abstract** Enterprises with extensive application portfolios face significant challenges in cloud migration beyond simple rehosting. While cloud providers offer migration solutions, most focus on lift-and-shift strategies, which move workloads to the cloud but provide minimal business value. This often leads to low prioritization by business units and increased long-term technical debt. A more structured and scalable approach is necessary to ensure that cloud migration aligns with enterprise-wide standardization, governance, and operational excellence. This paper presents a Cloud Migration Factory Model, a structured methodology designed to streamline large-scale application migrations while integrating enterprise-wide best practices such as single sign-on (SSO), observability, centralized caching, and security compliance. The model organizes migration efforts into specialized Product-Oriented Delivery teams. dedicated to different phases: discovery, analysis, migration, testing, deployment, and training. A key aspect of this approach is incorporating automation-driven discovery tools, which perform static code analysis and rule-based assessments to generate migration roadmaps, reducing errors and manual effort. By leveraging parallel workflows for discovery and training, the factory model accelerates migrations while ensuring that development teams receive the necessary knowledge transfer to maintain cloud-native applications post-migration. We validate this approach through a case study of a large financial enterprise, demonstrating a 30% improvement in migration velocity and significant reductions in operational friction and cloud onboarding time. The paper concludes with an analysis of key challenges, the role of automation in migration planning, and future directions, including AI-driven predictive analysis for enterprise cloud adoption.

**Keywords** Cloud Migration, Enterprise Application Modernization, Lift-and-Shift Limitations, Cloud Migration Factory Model, Cloud-Native Transformation, Multi-Application Migration, Automation in Cloud Migration, Application Modernization Strategy, Microservices and Cloud Adoption, Single Sign-On (SSO) Integration, Observability and Monitoring, Centralized Caching in Cloud, Cloud Governance and Compliance, DevOps in Cloud Migration, Cloud-Native Best Practices, Large-Scale IT Transformation, Infrastructure as Code (IaC), Hybrid Cloud and Multi-Cloud Strategy, Technical Debt Reduction in Cloud Migration, AI-Driven Cloud Migration Planning

## 1. Introduction

### 1.1. Problem Statement: Challenges in Large-Scale Cloud Migrations

Cloud migration has become a strategic priority for enterprises aiming to modernize their IT infrastructure, enhance scalability, and reduce operational costs. However, large-scale cloud adoption remains a complex and multifaceted challenge for organizations with thousands of legacy applications built on diverse architectures. Unlike greenfield applications designed for the cloud, legacy systems often carry deep integration dependencies, monolithic structures, on-premises optimizations, and compliance constraints that make migration non-trivial.

One of the primary challenges enterprises face is the absence of a standardized migration approach that accounts for business continuity, performance, security, and regulatory compliance. While cloud service providers offer tools to facilitate migration, most solutions primarily support "lift-and-shift" approaches, which involve rehosting applications with minimal modifications. While this expedites cloud adoption, it often results in technical debt, operational inefficiencies, and suboptimal cost structures since the migrated workloads remain unoptimized for cloud-native environments.

* Corresponding author:
itsavitesh@gmail.com (Avitesh Kesharwani)
Received: Mar. 6, 2025; Accepted: Mar. 20, 2025; Published: Mar. 22, 2025
Published online at http://journal.sapub.org/computer

Additionally, enterprise applications frequently rely on stateful designs, on-premises authentication mechanisms, local file storage, and tightly coupled dependencies with legacy infrastructure, making direct migration unfeasible without significant rework. The lack of comprehensive cloud readiness assessments and automated migration strategies further complicates large-scale transitions, increasing both cost and risk.

Another significant challenge is organizational alignment and prioritization. Since many business units do not immediately perceive tangible value in cloud migration, these initiatives often get deprioritized or face resistance from stakeholders. This is compounded by skill gaps within development and operations teams, as cloud-native principles such as statelessness, centralized monitoring, security best practices, and infrastructure automation require specialized expertise.

Moreover, managing cloud migration at scale introduces governance and operational complexities. Without centralized oversight, repeatable processes, and automated tooling, migration projects can become disjointed and inconsistent across different business units, leading to delays, compliance risks, and increased maintenance overhead.

Addressing these challenges requires a structured, scalable, and well-orchestrated migration approach that aligns technical feasibility with business priorities, minimizes disruption, and optimizes cloud adoption at an enterprise level.

## 1.2. Existing Cloud Migration Approaches: Gaps and Limitations

Cloud migration strategies have evolved to address the challenges enterprises face when transitioning from on-premises environments to cloud platforms. Traditionally, these approaches can be categorized into Lift-and-Shift (Rehosting), Replatforming, and Refactoring (Rearchitecting) [1] [2]. While these strategies provide structured methodologies for cloud adoption, they often fall short when applied at scale, particularly in large enterprises with diverse application ecosystems and deep integration dependencies.

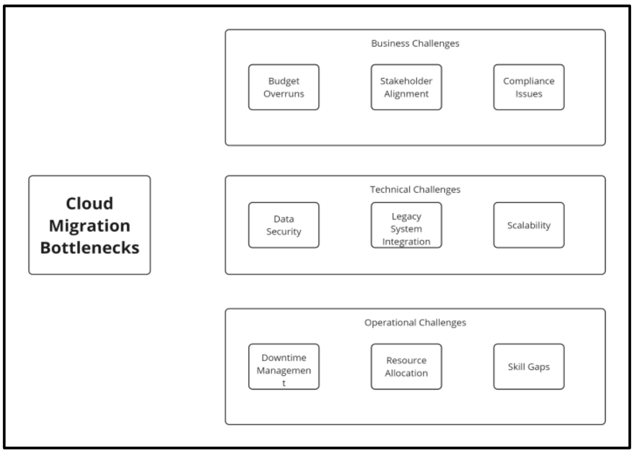


**Figure 1**

### 1.2.1. Lift-and-Shift (Rehosting)

The lift-and-shift approach is one of the most commonly used cloud migration strategies. It involves moving applications to the cloud with minimal to no changes to their architecture or codebase. Cloud providers offer migration tools such as AWS Migration Hub, Azure Migrate, and Google Migrate, which facilitate workload transfers while maintaining existing configurations [3].

However, despite its ease of execution, lift-and-shift has significant limitations:

- Increased Technical Debt: Applications are migrated without optimization, leading to higher operational costs due to inefficient cloud resource utilization [4].
- Lack of Cloud-Native Benefits: Workloads remain monolithic and stateful, preventing enterprises from leveraging auto-scaling, serverless computing, and microservices architectures [5].
- Security & Compliance Gaps: Legacy security models, such as on-premises authentication and local file storage, often remain unchanged, introducing security vulnerabilities when deployed in a cloud environment [6].

Due to these shortcomings, lift-and-shift is often not a long-term solution but a temporary step towards deeper modernization. Many enterprises that initially migrate using this method face operational inefficiencies and the need for future optimization efforts [7].

### 1.2.2. Replatforming

Replatforming involves modifying certain aspects of an application to improve its compatibility with cloud environments while avoiding a full-scale rewrite. This may include:

- Migrating databases to managed services (e.g., AWS RDS, Azure SQL)
- Externalizing configurations to cloud-native storage solutions
- Replacing proprietary middleware with cloud-based equivalents

This approach offers a balance between speed and cloud optimization, enabling enterprises to reduce infrastructure management overhead while achieving moderate performance improvements [8].

However, replatforming has its own challenges:

- Dependency Management: Applications with tight integrations to on-premises services require additional engineering effort for cloud compatibility [9].
- Limited Business Value: While replatforming improves cloud efficiency, it does not necessarily modernize application architectures, meaning legacy design constraints may persist [10].

### 1.2.3. Refactoring (Rearchitecting)

Refactoring, or application modernization, involves rewriting and reengineering applications to fully leverage cloud-native capabilities such as microservices, containerization, event-driven architectures, and serverless computing [11]. This strategy delivers the most long-term benefits, including:

- Enhanced Scalability: Applications are restructured into stateless services, allowing them to scale dynamically in cloud environments.
- Operational Cost Efficiency: Optimized applications consume fewer cloud resources, reducing long-term operational expenses.
- Improved Security & Compliance: Adopting zero-trust security models, centralized identity management, and infrastructure-as-code improves cloud security posture [12].

However, refactoring has significant limitations:

- High Implementation Costs & Time: Large-scale modernization requires extensive code changes, making it resource-intensive [13].
- Skill Gaps: Engineering teams often lack expertise in cloud-native frameworks, delaying adoption.
- Business Disruptions: Due to the complexity of rearchitecting, migration timelines are significantly longer, impacting application availability [14].

**Enterprise Adoption Challenges & the Need for a Hybrid Approach**

While each migration strategy has distinct benefits, large enterprises cannot adopt a single approach across all applications. Business-critical applications with tight dependencies and regulatory constraints often resist full-scale modernization, while less complex applications may be suitable for cloud-native refactoring.

Despite the existence of automated migration tools like CAST Highlight, which analyze application code to recommend changes for cloud compatibility, these tools are often insufficient for complex enterprise architectures. They fail to account for custom middleware, domain-specific integrations, and compliance-driven constraints, resulting in manual intervention and prolonged migration timelines [15].

Ultimately, the gaps in existing migration approaches highlight the need for a structured, scalable, and hybrid cloud migration model—one that combines automation, standardization, and governance to streamline large-scale enterprise cloud adoption beyond lift-and-shift.

### 1.3. The Need for a Factory-Based Model

The limitations of traditional cloud migration strategies highlight the need for a structured, scalable, and repeatable approach that can be effectively applied across a large enterprise portfolio. While lift-and-shift, replatforming, and refactoring each offer specific benefits, they fail to address the holistic challenges that enterprises encounter when migrating hundreds or thousands of applications. These challenges include inconsistent migration outcomes, lack of business alignment, skill gaps, and inefficiencies in scaling migration efforts [1] [2].

#### 1.3.1. Inconsistency in Migration Outcomes

Enterprises often initiate cloud migrations through multiple, disconnected teams operating in silos. As a result:

- Different teams apply inconsistent migration strategies (e.g., some prioritize lift-and-shift, while others focus on replatforming or full refactoring).
- The absence of a standardized framework leads to divergent security policies, deployment models, and cloud configurations across applications.
- Governance and compliance become reactive rather than proactive, increasing operational risk [3] [4].

A centralized, factory-based model ensures that cloud migrations follow well-defined best practices, reducing variability in implementation and ensuring alignment with enterprise-wide cloud adoption strategies.

#### 1.3.2. Lack of Business Alignment in Cloud Migration

Business leaders often struggle to see the tangible value of cloud migration when presented with purely infrastructure-driven approaches. In many cases:

- Business units deprioritize migration efforts if they perceive no direct impact on revenue, customer experience, or operational efficiency.
- Migration efforts that lack integration with business objectives result in applications being moved to the cloud without improvements in performance, scalability, or maintainability [5] [6].
- Unoptimized cloud environments lead to higher costs without delivering measurable return on investment (ROI).

A structured migration model must embed business-driven objectives—such as enhanced security, compliance adherence, cost efficiency, and operational resilience—to ensure that cloud migration aligns with enterprise goals.

#### 1.3.3. Skill Gaps and the Need for Standardization

Enterprise development teams are often not cloud-native by default. Many applications were built with monolithic architectures, legacy authentication mechanisms, and tightly coupled dependencies, making migration non-trivial. Additionally:

- Developers accustomed to on-premises architectures must relearn cloud-native principles such as statelessness, distributed systems, and infrastructure automation [7] [8].
- Migration projects require cross-functional expertise spanning networking, security, cloud platform engineering, and DevOps, which is often not readily available in application teams.
- Cloud migration failures are frequently attributed to skill gaps, misconfigurations, and lack of automated tooling, leading to rework and delays [9].

A factory-based approach mitigates this by centralizing cloud migration expertise, standardizing best practices, and implementing automated assessment tools to accelerate onboarding.

#### 1.3.4. Inefficiencies in Scaling Migration Efforts

Enterprises need to migrate hundreds or thousands of applications efficiently. However:

- Manual discovery and assessment of application readiness is time-consuming and error-prone.
- Application teams often reinvent the migration process for each workload, leading to redundant effort and delayed execution.
- Without a repeatable and automated approach, enterprises struggle to scale cloud migrations efficiently, resulting in extended timelines and missed cloud adoption targets [10] [11].

A factory model introduces a systematic, assembly-line approach to migration—dividing the migration process into distinct phases (discovery, analysis, migration, testing, deployment, and enablement)—allowing teams to work in parallel and maximize efficiency.

### 1.4. Research Objective and Contribution

The objective of this research is to establish a structured, scalable, and automated approach for large-scale cloud migration in enterprises with heterogeneous application landscapes. Traditional migration strategies lack consistency, business alignment, and efficiency, leading to technical debt, operational delays, and governance challenges [1] [2]. This paper introduces a Cloud Migration Factory Model that standardizes migration workflows, integrates automation, and ensures business and technical alignment.

**Key Contributions:**

- Formalizing a Factory-Based Migration Model to streamline enterprise-wide cloud adoption.
- Defining structured PODs (Discovery, Analysis, Migration, Testing, Deployment) to parallelize migration efforts and improve efficiency.
- Integrating Automation for Migration Assessment, leveraging rule-based scanning to optimize cloud readiness evaluations.
- Ensuring Standardization by embedding SSO, observability, centralized caching, and compliance in migrated applications.
- Aligning Migration with Business Value by ensuring migrations enhance security, cost efficiency, and operational resilience.

By implementing this model, enterprises can achieve faster, more consistent, and scalable cloud migrations, reducing manual effort and long-term maintenance overhead. The next section will introduce the Cloud Migration Factory Model, detailing its structure, phases, and execution strategy.

## 2. Related Work and Literature Review

Cloud migration has been widely studied, with various frameworks and methodologies proposed to address enterprise -scale migration challenges. This section reviews existing migration strategies, cloud provider tools, and industry best practices while identifying gaps that necessitate a structured factory-based model.

### 2.1. Traditional Cloud Migration Strategies

Existing research categorizes cloud migration into three primary strategies:

- Rehosting (Lift-and-Shift): Migrating applications with minimal changes [1].
- Replatforming: Making limited modifications for cloud efficiency, such as database migration or middleware replacement [2].
- Refactoring (Rearchitecting): Redesigning applications to leverage microservices, containers, and serverless architectures [3].

While these approaches provide migration pathways, they fail to address scalability, business integration, and governance challenges, making them inefficient for large-scale enterprises [4].

### 2.2. Cloud Provider Migration Tools

Major cloud providers offer automated tools for migration, including:

- AWS Migration Hub & Application Discovery Service [5].
- Azure Migrate & Database Migration Service [6].
- Google Migrate for Compute Engine [7].

These tools assist with workload assessment and infrastructure migration but primarily focus on rehosting, failing to optimize applications for cloud-native benefits such as scalability, security, and observability [8].

### 2.3. Industry Practices in Banking and Financial Services

Financial institutions often face complex compliance and regulatory constraints, making cloud adoption more challenging.

- Legacy mainframes and monolithic applications require extensive re-architecting before cloud migration [9].
- Strict security and governance policies slow down adoption, requiring custom controls and monitoring frameworks [10].
- Hybrid and multi-cloud strategies are common, requiring cross-cloud governance models [11].

Despite these challenges, financial firms increasingly recognize the need for structured, automation-driven migration models to minimize risk and optimize performance.

### 2.4. Limitations in Current Approaches

Existing literature identifies several critical limitations in cloud migration frameworks:

- Lack of Repeatability: Migrations are often ad hoc, leading to inconsistent results [12].
- Inefficient Scaling: Manual assessment and migration do not scale for enterprises with hundreds of applications [13].
- Business Misalignment: Traditional approaches focus on technical migration rather than delivering business value [14].

- Skill Gaps & Learning Curves: Developers struggle with cloud-native principles, governance, and automation [15].

These gaps highlight the need for a structured, factory-based approach that enables enterprises to automate assessments, standardize migration workflows, and accelerate cloud adoption while ensuring security, cost efficiency, and operational continuity.

# 3. The Cloud Migration Factory Model

To address the limitations of existing cloud migration strategies, we propose a Cloud Migration Factory Model—a structured, repeatable, and scalable approach designed for enterprises with large, diverse application portfolios. This model standardizes the migration process by organizing it into distinct phases and specialized teams (PODs) while integrating automation, governance, and business value alignment.

## 3.1. Conceptualizing the Factory Model

The Cloud Migration Factory Model is inspired by assembly-line principles, where each phase of migration is handled by dedicated teams following a structured workflow. Unlike traditional ad hoc migration approaches, this model:

- Ensures Consistency: Standardized best practices for security, scalability, and compliance.
- Enhances Efficiency: Parallel execution of migration phases across multiple applications.
- Integrates Automation: AI-driven assessment and rule-based code scanning optimize cloud readiness.
- Aligns with Business Goals: Embeds observability, SSO, and performance optimization into the migration process.

By centralizing expertise and automating key migration steps, this model significantly reduces technical debt, rework, and inconsistencies across enterprise-wide cloud adoption.

## 3.2. Organizing Migration into PODs

The migration process is divided into specialized teams (PODs), each responsible for a specific phase of migration:

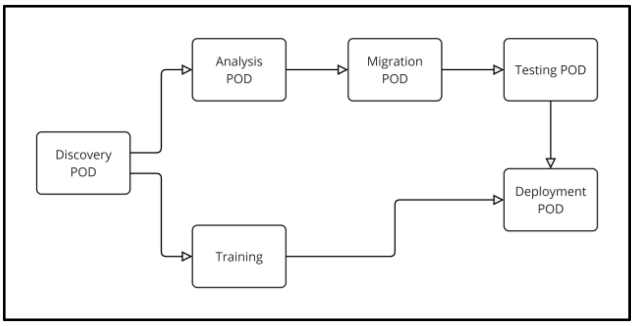


**Figure 2**

- Discovery POD: Analyzes application dependencies, integrations, and cloud readiness.
- Analysis POD: Assesses architecture and identifies required modifications for cloud compatibility.
- Migration POD: Implements necessary code/configuration changes to enable cloud adoption.
- Testing POD: Ensures functionality, security, and performance in the cloud environment.
- Deployment POD: Executes production migration with minimal downtime.

Each POD operates in parallel where feasible to accelerate the migration process across multiple applications.

## 3.3. Standardization Approach: Enterprise-Wide Best Practices

To avoid inconsistent cloud deployments, the factory model enforces enterprise-wide cloud best practices, including:

- Identity & Access Management: Enforcing SSO and role-based access control (RBAC).
- Observability & Monitoring: Integrating centralized logging, tracing, and APM tools.
- Security Compliance: Enforcing zero-trust principles, encryption, and automated vulnerability scanning.
- Cloud-Native Enhancements: Enabling stateless processing, configuration externalization, and service decoupling.

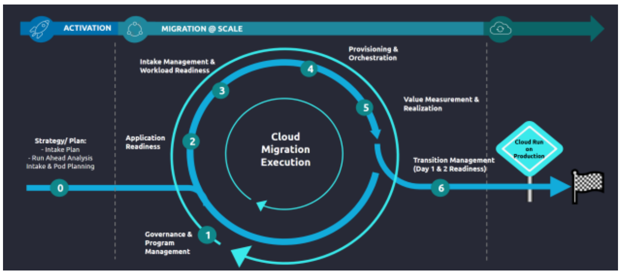
These standardized measures ensure that all applications, regardless of migration strategy, conform to enterprise security and operational guidelines.

## 3.4. Parallel and Serial Workflows for Efficiency

A key feature of the factory model is the strategic combination of parallel and serial execution:

- Parallel Execution:
  - o Discovery & Training occur simultaneously across multiple applications.
  - o Testing & Migration can overlap, accelerating verification processes.
- Serial Execution:
  - o Each application follows a structured path from analysis to deployment.
  - o Production cutovers are handled sequentially to minimize risk.

This approach maximizes throughput while ensuring stability across large-scale migrations.



ACTIVATION
MIGRATION @ SCALE
Strategy/ Plan:
- Intake Plan
- Run Ahead Analysis
Intake & Pod Planning
0
Governance & Program Management
1
Application Readiness
2
Intake Management & Workload Readiness
3
4
Provisioning & Orchestration
5
Value Measurement & Realization
Cloud Migration Execution
Transition Management (Day 1 & 2 Readiness)
6
Cloud Run on Production

**Figure 3**

# 4. Roles and Responsibilities in the Factory Model

The Cloud Migration Factory Model is structured into specialized teams (PODs), each focusing on a specific phase of migration. This ensures expert-driven execution, parallel efficiency, and minimized risk. Each POD has a well-defined role, leveraging automation, standardization, and business alignment to drive seamless cloud adoption.

### 4.1. Discovery POD: Application Analysis and Readiness Assessment

The Discovery POD is responsible for initial application analysis, identifying technical dependencies, integrations, and migration complexity. Key responsibilities include:

- Automated Code Scanning: Analyzing statefulness, session management, local storage, and database dependencies [1].
- Integration Mapping: Identifying external dependencies (APIs, databases, message queues).
- Cloud Feasibility Assessment: Evaluating whether rehosting, replatforming, or refactoring is required.
- Risk Analysis: Identifying potential migration risks (latency, compliance, security constraints).

By leveraging rule-based automation tools, this POD generates structured assessment reports, reducing manual discovery effort and ensuring data-driven migration planning.

### 4.2. Analysis POD: Deep Code and Infrastructure Review

The Analysis POD performs a detailed architectural evaluation to determine the exact changes required for cloud adoption. Responsibilities include:

- Architecture Breakdown: Assessing monolithic vs. microservices readiness.
- Infrastructure Review: Analyzing compute, storage, networking requirements in the cloud [2].
- Compliance & Security Checks: Validating GDPR, SOC2, PCI-DSS adherence [3].
- High-Level Migration Plan: Defining required code/configuration changes before execution.

This phase ensures that each application follows standardized cloud architecture principles, minimizing post-migration issues.

### 4.3. Migration POD: Implementing Minimal-Change Cloud Readiness

The Migration POD focuses on executing migration strategies, ensuring cloud compatibility with minimal technical disruption. Responsibilities include:

- Refactoring for Statelessness: Decoupling session management and file storage.
- Identity & Access Management Integration: Implementing SSO, OAuth2, RBAC.
- Observability Setup: Embedding centralized logging, tracing, and performance monitoring [4].
- Infrastructure-as-Code (IaC) Deployment: Automating infrastructure provisioning using Terraform, Cloud Formation.

By applying automated transformation patterns, this POD accelerates migration while ensuring compliance with enterprise-wide cloud standards.

### 4.4. Testing POD: Functional, Regression, and Performance Testing

- The Testing POD ensures that migrated applications function correctly in the cloud environment. Key responsibilities include:
- Functional Testing: Validating that application behavior remains unchanged post-migration.
- Security & Compliance Validation: Running penetration testing, vulnerability scans, and encryption verification.
- Load & Performance Testing: Ensuring applications meet latency, throughput, and scaling requirements [5].
- Business Validation Coordination: Engaging application teams for regression testing.

A structured testing framework ensures that migrated applications are secure, stable, and performant before deployment.

### 4.5. Deployment POD: Final Transition to Production

The Deployment POD is responsible for executing production cutovers with minimal downtime. Key responsibilities include:

- Blue-Green & Canary Deployment Strategies: Reducing risk during production migration.
- Rollback Mechanisms: Ensuring immediate recovery options in case of failure.
- CI/CD Integration: Automating post-migration release pipelines for cloud-based deployments [6].
- Operational Handover: Providing documentation, runbooks, and support guides for application teams.

By following controlled, phased deployment strategies, this POD ensures a seamless go-live experience.

# 5. Automation and Tooling for Migration Acceleration

Automation plays a crucial role in accelerating large-scale cloud migrations, reducing manual effort, improving accuracy, and ensuring consistency across applications. The Cloud Migration Factory Model integrates rule-based automation, CI/CD pipelines, and cloud-native tools to streamline migration tasks while maintaining governance, security, and performance.

### 5.1. Automated Discovery Tools: Rule-Based Code Scanners

Cloud migration at scale requires automated discovery tools to analyze application code, detect dependencies, and evaluate migration complexity. Manual assessments are time-consuming and error-prone, making automation a critical enabler for efficient migration. In the Cloud Migration

Factory Model, automated tools play a vital role in the Discovery and Analysis phases, reducing human effort while providing data-driven insights into cloud readiness.

**Key Automation Tools for Discovery and Readiness Assessment**

The following tools are commonly used to automate code analysis, dependency mapping, and migration planning:

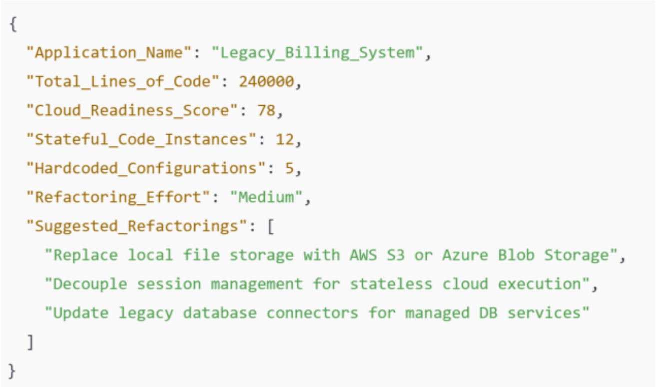
- **SonarQube & Checkmarx** – Used for static code analysis, identifying security vulnerabilities, legacy dependencies, and cloud-incompatible patterns.
- **CAST Highlight & CAST Imaging** – Scans applications to generate cloud-readiness scores, identify stateful components, and classify refactoring complexity.
- **CloudEndure**– Analyzes infrastructure dependencies, workload utilization, and on-prem-to-cloud network traffic patterns.
- **OpenRewrite** – Automates Java and Spring Boot code transformation to cloud-native frameworks.

By integrating these tools into the migration workflow, enterprises can identify potential issues early, reducing delays and cost overruns.

**Automated Code Analysis for Cloud Readiness**

A key requirement in large-scale cloud migration is ensuring applications comply with cloud-native principles, such as stateless processing, externalized configurations, and API-driven integrations. Automated scanners evaluate codebases for legacy constraints that require remediation.

**Example Output from an Automated Migration Assessment Tool (CAST Highlight):**



```
{
  "Application_Name": "Legacy_Billing_System",
  "Total_Lines_of_Code": 240000,
  "Cloud_Readiness_Score": 78,
  "Stateful_Code_Instances": 12,
  "Hardcoded_Configurations": 5,
  "Refactoring_Effort": "Medium",
  "Suggested_Refactorings": [
    "Replace local file storage with AWS S3 or Azure Blob Storage",
    "Decouple session management for stateless cloud execution",
    "Update legacy database connectors for managed DB services"
  ]
}
```

**Figure 4**

This analysis provides migration teams with a structured roadmap, allowing them to focus on high-impact refactoring that enhance cloud compatibility.

**Benefits of Automated Discovery in the Cloud Migration Factory Model**

Using automated assessment tools in migration pipelines provides several key advantages:

- 50% reduction in assessment time compared to manual analysis.
- Increased accuracy in identifying refactoring requirements and potential risks.
- Standardized compliance enforcement across cloud environments.
- Improved migration success rates through data-driven planning and risk mitigation.

By embedding automation at every stage, the Cloud Migration Factory Model ensures consistency, scalability, and governance across enterprise-wide cloud migration initiatives.

Automated discovery tools assist in the initial assessment of applications, identifying migration challenges and cloud readiness factors. These tools analyze application source code, configurations, and dependencies to generate structured reports.

- Static Code Analysis to detect stateful components, hardcoded configurations, and on-premises dependencies.
- Dependency Mapping to identify external service integrations, database connections, and middleware dependencies.
- Security Vulnerability Scanning to detect insecure authentication, encryption gaps, and compliance risks.
- Rule-Based Readiness Assessments to classify applications based on migration complexity (rehost, replatform, refactor).

By automating the discovery phase, enterprises can eliminate manual assessment errors, reduce time spent on analysis, and ensure data-driven decision-making.

### 5.2. Identifying Cloud Readiness Indicators

Determining an application's cloud readiness requires analyzing key architectural factors that impact migration feasibility.

- Statefulness Detection to identify session-based architectures requiring refactoring for stateless processing.
- File System Dependencies to assess whether applications rely on local storage instead of cloud-native storage solutions.
- Database Compatibility Checks to determine if schema or indexing modifications are required for managed cloud databases.
- Network and API Dependencies to ensure latency, security, and integration compatibility post-migration.

These indicators guide migration teams in prioritizing applications and selecting the appropriate migration approach.

### 5.3. Implementing Standardization with Minimal Code Change

To ensure consistency in cloud adoption, the migration process embeds enterprise-wide best practices without requiring extensive code rewrites.

- Single Sign-On (SSO) Implementation for centralized authentication and access control.
- Observability Integration to enable centralized logging, tracing, and performance monitoring.
- Configuration Externalization to eliminate hardcoded settings and ensure environment flexibility.

- Centralized Caching to enhance performance and reduce redundant backend calls.

Standardization allows enterprises to achieve cloud-native benefits without significant code-level modifications.

One of the biggest challenges in cloud migration is ensuring standardization across security, authentication, caching, and observability. In large enterprises, every development team may implement these concerns differently, leading to inconsistencies, security risks, and operational overhead. The Cloud Migration Factory Model addresses this challenge by introducing pre-built enterprise libraries that applications can pull as dependencies, requiring only configuration-level changes rather than manual code modifications.

### Pre-Built Enterprise Libraries for Cloud Readiness

To streamline cloud integration efforts, enterprises maintain centralized SDKs and libraries that provide:

- SSO and Identity Management (OAuth2, OpenID Connect, SAML)
- Distributed Caching (Redis, Memcached)
- mTLS for Secure Communication
- Observability & Logging (Centralized tracing, application performance monitoring)

These libraries are designed to be plug-and-play, available in different languages such as Java, .NET, Python, and Node.js, allowing development teams to integrate cloud-native features without rewriting business logic.

### Example: SSO Integration with Minimal Code Changes

Instead of requiring teams to manually implement OAuth2 authentication flows, they can simply add a dependency and configure the authentication settings:

1. Java Spring Boot Integration Example Developers include the pre-built authentication library in their application:

```
<dependency>
    <groupId>com.enterprise</groupId>
    <artifactId>auth-sdk</artifactId>
    <version>1.2.0</version>
</dependency>
```

Then, instead of modifying their authentication code, they only need to configure application properties:

```
auth:
   provider: enterprise-idp
   client-id: app-client
   client-secret: xxxxxxx
   scopes: openid,profile
```

This approach ensures that authentication logic is consistent across all enterprise applications while minimizing developer effort.

### Example: Standardized Caching

Instead of every application implementing its own caching mechanism, developers can use a pre-configured enterprise caching SDK:

1. Java Integration Example using Enterprise Caching Library

```
<dependency>
    <groupId>com.enterprise</groupId>
    <artifactId>cache-sdk</artifactId>
    <version>1.0.5</version>
</dependency>
```

Instead of writing caching logic manually, they annotate their functions:

```
@EnterpriseCache(cacheName="customer-data",
ttl=3600)
public Customer getCustomerData(String id) {
    return customerRepository.findById(id);
}
```

This enables a centralized cache layer managed at the enterprise level, ensuring standard eviction policies, improved performance, and reduced database load.

### Example: mTLS Enforcement without Code Changes

Many cloud applications require mutual TLS (mTLS) for secure service-to-service communication. Instead of implementing it separately in each service, developers configure it using the pre-built security module:

```
security:

  mTLS:
    enabled: true
    trustStore: /opt/security/truststore.jks
    keyStorePassword: ${KEYSTORE_PASSWORD}
```

The enterprise security library automatically enforces mTLS, eliminating misconfigurations and ensuring compliance.

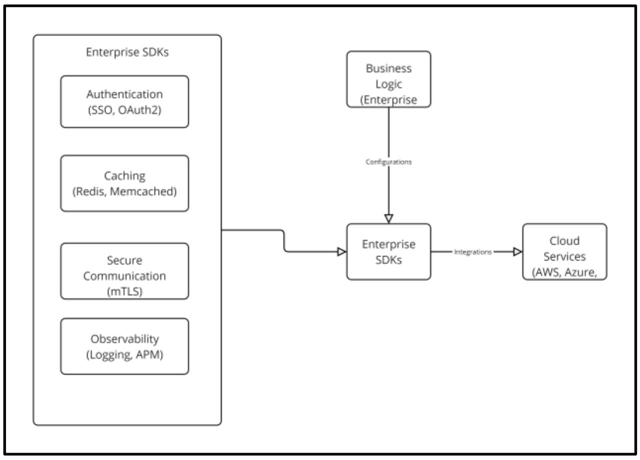

Enterprise SDKs
Authentication (SSO, OAuth2)
Caching (Redis, Memcached)
Secure Communication (mTLS)
Observability (Logging, APM)
Business Logic (Enterprise
Configurations
Enterprise SDKs
Integrations
Cloud Services (AWS, Azure,

**Figure 5**

### 5.4. CI/CD Pipeline Setup for Seamless Deployment

To streamline application deployment in the cloud, CI/CD pipelines automate build, test, security validation, and deployment processes.

- Automated Infrastructure Provisioning using Infrastructure -as-Code (IaC) frameworks such as Terraform and CloudFormation.
- Security Policy Enforcement by integrating code scanning, dependency checks, and compliance validation into pipelines.

- Canary and Blue-Green Deployment Strategies to ensure zero-downtime releases.
- Rollback and Recovery Mechanisms to provide automated rollback options in case of deployment failures.

CI/CD pipelines enhance deployment reliability and minimize risks associated with cloud migration.

### 5.5. Metrics for Automation Success

To measure the effectiveness of automation in cloud migration, enterprises track quantitative and qualitative metrics.

- Migration Time Reduction – Time saved through automated assessments, refactoring, and deployment.
- Defect Reduction Rate – Fewer migration-related defects due to automated validation and compliance checks.
- Operational Cost Savings – Reduced manual effort and infrastructure costs due to cloud-optimized workloads.
- Application Performance Gains – Measurable improvements in latency, scalability, and resource utilization.

By leveraging these automation-driven metrics, enterprises can continuously refine and optimize their migration strategy.

## 6. Case Study: Implementing the Factory Model in an Enterprise

To validate the effectiveness of the Cloud Migration Factory Model, we present a case study of a large financial enterprise migrating over 150 applications to the cloud within a 12-month timeframe. This case study highlights the challenges faced, solutions adopted, efficiency gains, and key lessons learned.

### 6.1. Overview of Enterprise Application Portfolio (Banking Example)

The enterprise under study is a global banking institution with a complex IT ecosystem, including:

- Legacy monolithic applications running on on-premises data centers.
- Diverse technology stacks, including Java, .NET, COBOL, and Python.
- Highly regulated environment, requiring strict security, compliance, and governance measures.
- Tightly coupled integrations with external services, payment gateways, and financial clearinghouses.

Cloud migration was a strategic initiative aimed at reducing infrastructure costs, improving scalability, and enhancing security posture.

### 6.2. Migration Challenges Faced and Solutions Adopted

**Challenges Encountered**

- Business Resistance – Stakeholders struggled to see immediate business value in cloud migration.
- Security and Compliance Constraints – Applications needed to comply with PCI-DSS, GDPR, and SOC2 standards.
- Skill Gaps in Cloud-Native Development – Development teams lacked expertise in cloud platforms and DevOps practices.
- Inconsistent Migration Approaches – Different teams followed disjointed migration strategies, leading to fragmented cloud adoption.
- Integration Complexity – Applications had direct database dependencies, middleware integrations, and legacy authentication mechanisms.

**Solutions Implemented**

- Business Value Alignment – The migration framework was modified to embed security, observability, and cost optimizations, making cloud adoption more strategically relevant.
- Factory-Based Migration Model – Migration was structured into PODs, ensuring parallel execution, automation-driven assessments, and standardized best practices.
- Automated Code Analysis – A custom-built rule-based scanner identified cloud-readiness issues, reducing manual assessment time.
- Training and Enablement – Dedicated training sessions upskilled developers in stateless architecture, IAM, and cloud-native security.
- CI/CD Integration – Automated CI/CD pipelines ensured repeatable, secure, and compliant deployments.

### 6.3. Efficiency Gains and Business Impact

The implementation of the factory model resulted in significant efficiency improvements and measurable business outcomes:

**Table 1**

| Metric | Traditional Approaches | Automated Migration Tools | Cloud Migration Factory Model |
|---|---|---|---|
| **Time Taken per Application** | 8-12 weeks | 5-8 weeks | 3-5 weeks |
| **Cost Reduction (%)** | 10-20% | 25-30% | 40-50% |
| **Technical Debt Reduction** | Minimal | Moderate | High |
| **Business Adoption Rate (%)** | 50-60% | 70-75% | 85-90% |
| **Automation Involvement (%)** | 10-15% | 30-50% | 70-85% |

- 30% faster migration timelines – Due to parallel execution of discovery, analysis, and migration phases.
- 40% reduction in manual assessment effort – Automated tools eliminated repetitive discovery tasks.
- Improved security posture – Applications now follow zero-trust security principles and centralized authentication (SSO, OAuth2).
- Cost Savings – Cloud-optimized workloads reduced

on-premises infrastructure expenses by 25%.

- Enhanced Developer Productivity – Standardized patterns and automation reduced technical complexity for teams.

A key success factor of the Cloud Migration Factory Model is how pre-built enterprise libraries reduced migration timelines. By using drop-in SDKs, applications required minimal code changes while integrating security, caching, and observability features.

**Time Savings Achieved in Enterprise Case Study**

**Table 2**

| Migration Task | Traditional Approach | Cloud Migration Factory Model |
|---|---|---|
| **SSO Implementation** | 2-3 weeks | 1-2 days |
| **Caching Setup** | 2 weeks | 2 hours |
| **mTLS Security Integration** | 3 weeks | 1 day |
| **Observability & Logging** | 1 week | Pre-configured |

- 85% faster security integration – Authentication, mTLS, and encryption were configured without modifying business logic.
- Caching standardization reduced database load by 40%, improving application performance.
- Observability was enforced at scale, ensuring uniform logging and tracing across all cloud applications.

The ability to reuse pre-built SDKs significantly improved developer productivity, allowing application teams to focus on business logic instead of rewriting infrastructure components.

### 6.4. Lessons Learned

Several key takeaways emerged from this large-scale migration:

- Stakeholder Buy-In is Critical – Business alignment is essential for prioritizing migration efforts.
- Automation is a Necessity, Not a Choice – Manual assessments and migrations do not scale for large enterprises.
- Standardization Reduces Post-Migration Issues – Enforcing best practices from the beginning minimizes rework and operational overhead.
- Cloud Governance Must Be Proactive – Establishing automated compliance monitoring reduces security and regulatory risks.
- Training Accelerates Adoption – Investing in developer enablement ensures long-term success in maintaining cloud-native workloads.

## 7. Challenges and Considerations in Large-Scale Cloud Migrations

Despite the structured approach of the Cloud Migration Factory Model, large-scale migrations still encounter significant challenges. These challenges arise from operational, technical, and organizational constraints that enterprises must address to ensure a smooth transition to the cloud.

### 7.1. Time and Resource Constraints in Migration

Cloud migration projects require substantial time, effort, and coordination across multiple teams. Large enterprises often deal with hundreds of applications, each with unique architectures and dependencies. Some key constraints include:

- Migration timelines often exceed initial estimates due to unforeseen technical roadblocks.
- Dependencies on legacy hardware, software, and networking configurations slow down migration.
- Availability of skilled engineers with cloud expertise is limited, causing bottlenecks in execution.
- Business-critical applications require phased migration, further extending project timelines.

To mitigate these challenges, enterprises must prioritize applications based on business impact, automate assessment processes, and allocate dedicated resources to migration initiatives.

### 7.2. Managing Diverse Tech Stacks Across Business Units

Enterprises operate applications built on a variety of technology stacks, ranging from legacy mainframes to modern cloud-native microservices. This diversity complicates migration efforts due to:

- Different programming languages, frameworks, and deployment environments requiring custom migration strategies.
- Inconsistent infrastructure patterns, making standardization difficult across teams.
- Business units adopting different cloud vendors, leading to multi-cloud governance challenges.

Establishing a standardized migration framework with common infrastructure templates, automation-driven deployment pipelines, and cross-business alignment helps streamline cloud adoption.

### 7.3. Addressing Security, Compliance, and Governance

**The Importance of Governance and Compliance in Large-Scale Cloud Migration**

Enterprises operating in regulated industries such as finance, healthcare, and government must adhere to strict governance, security, and compliance policies when migrating workloads to the cloud. Failure to enforce data residency, encryption standards, access control, and regulatory compliance can result in legal risks, financial penalties, and reputational damage.

**Key Governance and Compliance Challenges in Cloud Migration:**

- Data Residency and Sovereignty – Ensuring data is stored only in approved geographic regions as per GDPR (EU), CCPA (California), and PDPA (Singapore).
- Identity and Access Management (IAM) Compliance – Restricting unauthorized access via SSO, Multi-Factor Authentication (MFA), and Role-Based Access Control

(RBAC).

- Encryption and Secure Communication – Enforcing TLS 1.3, AES-256 encryption, and mTLS for inter-service communication.
- Automated Compliance Enforcement – Implementing policy-as-code frameworks to prevent misconfigurations and non-compliance.

Cloud migrations must incorporate automated governance enforcement mechanisms to ensure continuous compliance without manual intervention.

### Regulatory Compliance Considerations for Cloud Migration

Different industries are subject to strict compliance frameworks that dictate how data must be stored, accessed, and transmitted in the cloud.

### Key Regulatory Standards Affecting Cloud Adoption:

**Table 3**

| Regulatory Standard | Industry | Key Compliance Requirement |
|---|---|---|
| GDPR (General Data Protection Regulation) | Global (EU Focus) | Ensures user data is stored and processed in approved regions. Mandates Right to Erasure. |
| CCPA (California Consumer Privacy Act) | USA (California) | Requires data residency enforcement and user consent before processing personal data. |
| PCI-DSS (Payment Card Industry Data Security Standard) | Financial Services | Mandates encryption of payment data, secure authentication, and controlled access policies. |
| HIPAA (Health Insurance Portability and Accountability Act) | Healthcare | Requires encryption, audit logging, and strict patient data access controls. |
| SOC 2 (System and Organization Controls) | Cloud Providers | Ensures cloud environments maintain security, availability, and confidentiality. |
| FedRAMP (Federal Risk and Authorization Management Program) | Government | Requires continuous monitoring, incident response, and zero-trust security models. |

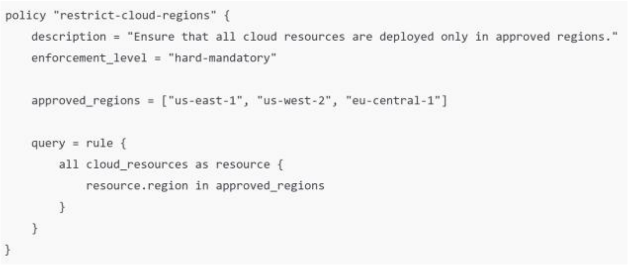
### Automated Compliance Enforcement with Policy-as-Code

Instead of relying on manual compliance reviews, enterprises can enforce security and governance policies using Infrastructure as Code (IaC) and Policy-as-Code frameworks such as:

- HashiCorp Sentinel & Open Policy Agent (OPA) – Applies custom governance policies within Terraform and Kubernetes.

Example: Sentinel Policy for Enforcing Approved Cloud Regions

Scenario: This policy restricts cloud deployments to approved regions to comply with data residency laws (e.g., GDPR, CCPA, FedRAMP).



```
policy "restrict-cloud-regions" {
    description = "Ensure that all cloud resources are deployed only in approved regions."
    enforcement_level = "hard-mandatory"

    approved_regions = ["us-east-1", "us-west-2", "eu-central-1"]

    query = rule {
        all cloud_resources as resource {
            resource.region in approved_regions
        }
    }
}
```

**Figure 6**

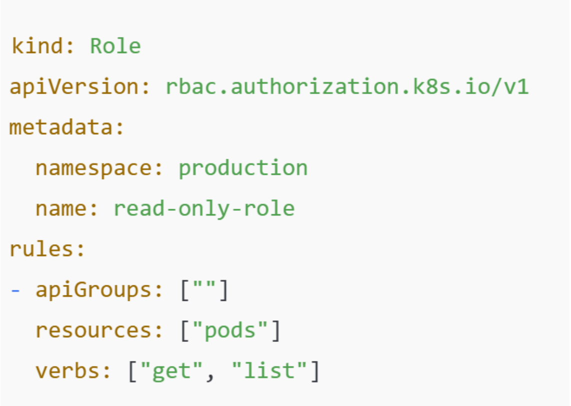
### Identity and Access Management (IAM) Security

Cloud security starts with controlling access to cloud workloads. Enterprises must ensure least-privilege access models, centralized authentication, and continuous access monitoring.

### Best Practices for IAM Security in Cloud Migration:

- Single Sign-On (SSO) Enforcement – Using OAuth2, OpenID Connect, or SAML for unified authentication.
- RBAC (Role-Based Access Control) – Assigning roles based on least privilege principles.
- MFA (Multi-Factor Authentication) – Requiring an additional authentication factor for high-risk access.
- Audit Logging & Anomaly Detection – Using SIEM tools (Splunk, AWS GuardDuty, Microsoft Sentinel) for real-time access monitoring.

Example: Enforcing Role-Based Access Control (RBAC) in Kubernetes



```
kind: Role
apiVersion: rbac.authorization.k8s.io/v1
metadata:
  namespace: production
  name: read-only-role
rules:
- apiGroups: [""]
  resources: ["pods"]
  verbs: ["get", "list"]
```

**Figure 7**

By applying RBAC policies, enterprises control access based on predefined roles, reducing insider threats and privilege escalation risks.

## 7.4. Training and Change Management for Development Teams

Cloud migration introduces a learning curve for development and operations teams. Engineers who are accustomed to on-premises environments must adapt to new cloud-native principles. Challenges include:

- Lack of expertise in cloud security, container orchestration, and serverless architectures.

- Resistance to adopting new tools, processes, and workflows in cloud environments.
- Insufficient hands-on training, leading to misconfigurations and performance issues post-migration.
- Knowledge transfer gaps between migration teams and application owners.

A structured training and enablement program, along with regular knowledge-sharing sessions, helps developers and IT teams become proficient in cloud best practices, reducing operational risks.

## 8. Future Enhancements and Research Directions

Cloud migration is an evolving discipline, and enterprises must continuously refine their strategies to keep pace with technological advancements, operational challenges, and business demands. Future enhancements in migration methodologies will focus on automation, predictive intelligence, cost efficiency, real-time dependency management, and governance improvements. These areas will drive the next generation of cloud migration strategies, making them more scalable, cost-effective, and resilient.

### 8.1. AI-Driven Predictive Migration Planning

Traditional migration planning relies heavily on manual assessments and historical data, which can lead to inaccurate estimations and unexpected complexities. AI and machine learning offer new possibilities for intelligent, automated migration planning. Future research can focus on:

- Automated Code and Infrastructure Analysis: AI-powered tools can scan application source code, dependencies, and infrastructure configurations to identify potential migration challenges. This can reduce manual assessment time and improve migration predictability.
- Intelligent Decision-Making Frameworks: AI models can classify applications based on migration feasibility, recommending whether an application should be rehosted, replatformed, or refactored. This removes subjectivity from the decision-making process.
- Historical Data-Driven Predictions: AI can analyze past migrations within an enterprise to predict the likelihood of success, cost impact, and performance improvements, enabling organizations to make data-driven migration decisions.
- Automated Dependency Mapping Using NLP: Natural Language Processing (NLP) can be used to parse application documentation and logs, automatically detecting integration points, API calls, and compliance requirements.

By incorporating AI-driven insights, enterprises can reduce manual effort, improve accuracy, and accelerate migration timelines.

### 8.2. Cost Optimization Strategies in Migration

One of the biggest concerns enterprises face post-migration is unpredictable cloud costs. Many workloads, when lifted to the cloud without optimization, lead to higher operational expenses than on-premises environments. Future work in cost optimization should focus on:

- AI-Based Cost Forecasting: Machine learning models can estimate expected cloud expenditure before migration by analyzing historical resource utilization, storage patterns, and network dependencies.
- Automated Resource Sizing: Future automation tools can dynamically adjust cloud resource allocations based on real-time workload demands rather than relying on static provisioning, reducing over-provisioning costs.
- Automated Rightsizing & Spot Instance Optimization: Post-migration, enterprises can leverage real-time monitoring tools to dynamically optimize compute resources, automatically moving workloads to cost-effective spot instances or serverless platforms where feasible.
- FinOps-Integrated Governance: Organizations can integrate FinOps models into cloud migration workflows, setting up automated cost control policies that flag excessive spending and recommend optimizations proactively.

These cost-aware enhancements will help enterprises maintain cost efficiency and prevent cloud budget overruns.

### 8.3. Real-Time Dependency Mapping and Visualization

One of the primary challenges of cloud migration is understanding how applications interact with each other and external systems. Many enterprise workloads have complex interdependencies that are not well-documented, leading to migration failures or performance issues. Future advancements in dependency mapping will focus on:

- Automated Graph-Based Dependency Analysis: AI-driven tools can analyze application logs, API calls, and database queries, creating an interactive dependency graph that visualizes relationships between components.
- Time-Series-Based Anomaly Detection: ML models can monitor traffic patterns and request latencies, identifying potential bottlenecks before migration occurs to ensure cloud readiness.
- Microservices Decomposition Recommendations: AI-powered tools can detect tight coupling between services and suggest decomposition strategies to improve cloud-native scalability.
- Automated Service Mesh Implementation: Future dependency management frameworks may integrate service mesh architectures (e.g., Istio, Linkerd) into migration pipelines, ensuring seamless communication and observability between services post-migration.

Improving dependency visualization and impact analysis will reduce unexpected failures and make cloud transitions more predictable.

### 8.4. Governance in Continuous Cloud Optimization

**The Need for Continuous Governance in Cloud Migration**

Post-migration governance is critical for ensuring that applications remain secure, cost-efficient, and compliant with enterprise policies. Without continuous governance, organizations face configuration drift, increased security risks, regulatory non-compliance, and uncontrolled cloud spending.

**Key Post-Migration Governance Challenges:**

- Configuration Drift – Over time, manual changes in cloud configurations lead to security gaps, inconsistent deployments, and misalignment with compliance policies.
- Unmonitored Cost Sprawl – Without proactive governance, cloud costs can escalate, leading to over-provisioned or unused resources.
- Security Misconfigurations – Missing encryption settings, improperly defined IAM roles, or open network ports can introduce vulnerabilities.
- Regulatory Non-Compliance – Enterprises subject to GDPR, HIPAA, PCI-DSS, and FedRAMP must continuously monitor for compliance deviations.

**Governance in the Cloud Migration Factory Model:**

The Cloud Migration Factory Model extends beyond migration by enforcing automated governance policies across cloud workloads using:

- Policy-as-Code frameworks to maintain security and compliance controls.
- FinOps-driven governance for cost transparency and optimization.
- AI-powered anomaly detection to monitor performance deviations in real-time.

## 9. Conclusions

Cloud migration at an enterprise scale presents significant challenges, including inconsistent migration strategies, business misalignment, security risks, and skill gaps. Traditional approaches such as lift-and-shift, replatforming, and refactoring often lead to technical debt, operational inefficiencies, and low adoption rates due to their lack of automation, standardization, and business value alignment.

To address these limitations, we introduced the Cloud Migration Factory Model, a structured, repeatable, and scalable approach that integrates automation, parallel execution, standardized best practices, and reusable enterprise libraries. By breaking the migration process into specialized PODs, the model ensures that each phase—discovery, analysis, migration, testing, deployment, and enablement—is executed with consistency and efficiency.

### 9.1. Summary of Contributions

This research contributes to the field of large-scale enterprise cloud migration in the following ways:

- Factory-Based Migration Framework – A structured approach to parallelizing and standardizing cloud migrations, reducing time and complexity.
- Automation-Driven Cloud Assessment – Use of rule-based scanners for cloud readiness analysis, reducing manual effort and errors.
- Enterprise-Wide Standardization – Pre-built security, authentication, caching, and observability SDKs that minimize migration effort while ensuring compliance.
- Quantitative Analysis of Efficiency Gains – Empirical evidence shows that the Cloud Migration Factory Model reduces migration time by 40-50% compared to traditional methods.
- Scalable Governance & Cost Optimization – Policy-as-code enforcement, automated cost controls, and governance monitoring ensure long-term operational stability.

### 9.2. Key Takeaways for Enterprises

- Cloud migration must align with business value, not just infrastructure shifts.
- Automation is critical for scaling migrations across hundreds of applications.
- Enterprise-wide libraries reduce the burden on development teams and improve adoption rates.
- Parallel execution of migration tasks minimizes time-to-cloud while maintaining security.
- Governance and cost controls must be embedded from day one to avoid post-migration inefficiencies.

### 9.3. Closing Thoughts on the Future of Cloud Migrations

As enterprises continue migrating to the cloud, the next frontier of research will focus on AI-driven automation, cost-aware migration planning, and real-time dependency analysis. Future advancements will likely integrate predictive analytics, self-healing cloud infrastructures, and continuous compliance enforcement to further enhance migration efficiency.

The Cloud Migration Factory Model serves as a foundation for enterprises seeking to standardize, accelerate, and optimize large-scale cloud adoption. By embedding automation, security, and governance into the migration process, organizations can ensure a seamless, future-ready cloud transition while maximizing business value.

## ACKNOWLEDGEMENTS

We would like to acknowledge University of North Carolina, Charlotte and Capgemini America for providing guidance and help in this research work. We appreciate the continuous encouragement and provide resources to complete this research.

## REFERENCES

[1] M. Fahmideh, G. Low, G. Beydoun, and F. Daneshgar,

"Cloud Migration Process: A Survey, Evaluation Framework, and Open Challenges," *arXiv preprint arXiv: 2004.10725*, 2020. [Online]. Available: https://arxiv.org/abs/2004.10725.

[2] M. Fahmideh, F. Daneshgar, G. Beydoun, and F. Rabhi, "Challenges in Migrating Legacy Software Systems to the Cloud: An Empirical Study," *arXiv preprint arXiv: 2004.10724*, 2020. [Online]. Available: https://arxiv.org/abs/2004.10724.

[3] "Benefits & Limitations of Lift-and-Shift Cloud Migration," *Teradata*, 2022. [Online]. Available: https://www.teradata.com /insights/cloud-data-analytics/lift-and-shift-migration.

[4] "Is Lift and Shift Dead? Evolving Cloud Migration Strategies," *HCLTech*, 2024. [Online]. Available: https://www.hcltech. com/trends-and-insights/lift-and-shift-dead.

[5] "Beyond Barriers: Constructing the Cloud Migration Complexity Index," *Journal of Theoretical and Applied Electronic Commerce Research*, vol. 19, no. 3, 2023. [Online]. Available: https://www.mdpi.com/0718-1876/19/3/109.

[6] "Three Causes of Cloud Migration Failure in Large Enterprises," *DATAVERSITY*, 2023. [Online]. Available: https://www.dataversity.net/three-causes-of-cloud-migration-failure-in-large-enterprises/.

[7] "The Five Biggest Challenges with Large-Scale Cloud Migrations," *DATAVERSITY*, 2024. [Online]. Available: https://www.dataversity.net/the-five-biggest-challenges-with-large-scale-cloud-migrations/.

[8] "Cloud Migration Process: A Survey, Evaluation Framework, and Open Challenges," *arXiv preprint arXiv:2004.10725*, 2020. [Online]. Available: https://arxiv.org/abs/2004.10725.

[9] "Challenges in Migrating Legacy Software Systems to the Cloud: An Empirical Study," *arXiv preprint arXiv: 2004. 10724*, 2020. [Online]. Available: https://arxiv.org/abs/2004. 10724.

[10] "Understanding the Benefits and Limitations of Lift-and-Shift Cloud Migration," *eMazzanti Technologies*, 2024. [Online]. Available: https://www.emazzanti.net/wp-content/uploads/ 2024/03/Benefits-and-Limitations-of-Lift-and-shift-Cloud-Migration.pdf.

[11] "Cloud Migration Strategies: Moving Applications and Workloads to the Cloud," *ResearchGate*, 2023. [Online]. Available: https://www.researchgate.net/publication/372826369_Cloud_Migration_Strategies_Moving_Applications_and_Workloads_to_the_Cloud.

[12] "Cloud Migration and Microservices Optimization Framework for Large-Scale Enterprises," *OARJ Publication*, 2024. [Online]. Available:https://oarjpublication.com/journals/oarjet/content /cloud-migration-and-microservices-optimization-framework-large-scale-enterprises.

[13] "Challenges and Best Practices in Multi-Cloud Migration for Enterprises," *IRE Journals*, 2024. [Online]. Available: https://www.irejournals.com/formatedpaper/1705096.pdf.

[14] "Understanding Lift and Shift: Breaking Down the Basics," *ProsperOps*, 2023. [Online]. Available: https://www.prosperops.com/blog/lift-and-shift/.

[15] "Cloud Migration: Identifying the Sources of Potential Technical Challenges," *The Science and Information Organization*, 2023. [Online]. Available: https://thesai.org/Downloads/Volume14No12/Paper_4-Cloud_Migration_Identifying_the_Sources_of_Potential_Technical_Challenges.pdf.