Journal of Informatics Education and Research
ISSN: 1526-4726
Vol 6 Issue 1 (2026)

Vendor Lock-In and Cloud Portability in Serverless Environments: An Empirical Investigation with Special Reference to the Indian IT Industry

Mr. Gokul Bodke[1], Dr. Samadhan Bundhe[2]

[1]Research Scholar, Sandip University, Nashik, Maharashtra, India

[2]Associate Professor, Sandip University, Nashik, Maharashtra, India

Abstract — Cloud computing has fundamentally transformed enterprise information technology infrastructure over the past two decades, yet the phenomenon of vendor lock-in continues to constrain organisational agility and architectural freedom. This paper reports findings from a mixed-method empirical investigation involving 306 information technology professionals drawn from organisations operating across India's major technology hubs. The study examines how cloud service model abstraction correlates with perceived vendor lock-in severity, how the emergence of serverless computing specifically Function-as-a-Service architectures has intensified portability challenges, and what role open portability standards and cloud-agnostic tooling play in mitigating lock-in risk. Four hypotheses were tested using chi-square analysis, one-way analysis of variance, and Pearson correlation. Results confirm that vendor lock-in severity increases significantly as the cloud service model transitions from Infrastructure-as-a-Service to Platform-as-a-Service to Software-as-a-Service ($\chi^2(6) = 38.742$, $p < 0.001$). Serverless computing introduces the most acute portability constraints among all cloud deployment paradigms, with 68.3 per cent of serverless practitioners identifying provider-specific event trigger formats as a major migration barrier. Open standards adoption including containerisation, infrastructure-as-code, and Kubernetes — demonstrates a moderate but statistically significant positive correlation with lock-in risk reduction ($r = 0.487$, $p < 0.001$). The paper concludes with eleven actionable recommendations for cloud consumer organisations, cloud service providers, and Indian regulatory bodies, addressing the urgent need for an India-specific cloud portability governance framework.

Keywords: cloud portability, vendor lock-in, serverless computing, Function-as-a-Service, cloud standards, TOSCA, Kubernetes, Indian IT industry, empirical research, mixed method

1. Introduction

The adoption of cloud computing services by enterprises, government bodies, and academic institutions has accelerated at a pace that few analysts anticipated even a decade ago. Organisations across industry verticals have migrated critical workloads to public cloud platforms operated by global hyperscalers a transition driven by the promise of elastic scalability, reduced capital expenditure, and access to managed services that would otherwise require substantial in-house engineering investment. However, as cloud adoption matures, a growing body of practitioner experience and academic inquiry has surfaced a challenge that receives far less attention in the popular discourse than it deserves: the difficulty of moving workloads, data, and applications away from the cloud provider on which they were originally deployed.

Vendor lock-in in cloud computing refers to the state of technological, economic, and operational dependency that emerges when an organisation's infrastructure is so deeply intertwined with a specific provider's proprietary services, APIs, data formats, and tooling that migration to an alternative environment becomes prohibitively expensive, technically risky, or strategically unfeasible. This dependency does not arise through any single architectural decision; rather, it accumulates gradually as teams adopt managed databases, proprietary orchestration services, platform-native machine learning pipelines, and provider-specific event-processing frameworks each adding another strand to what eventually becomes a very dense web of commitment.

The emergence and rapid mainstream adoption of serverless computing particularly the Function-as-a-Service paradigm popularised by AWS Lambda, Microsoft Azure Functions, and Google Cloud Functions has added a new and more severe dimension to this challenge. Serverless architectures, by design, abstract away virtually all

infrastructure concerns from the application developer, enabling organisations to build and deploy event-driven applications with remarkable speed and operational simplicity. However, this abstraction comes at a portability cost that is qualitatively different from that of traditional IaaS or even PaaS deployments. Serverless functions are tightly coupled to provider-specific event sources, runtime environments, managed integrations, and monitoring ecosystems in ways that make cross-platform migration substantially more complex than simply redeploying a containerised application on a different Kubernetes cluster.

India's information technology industry occupies a particularly important vantage point from which to examine these dynamics. As the world's largest IT services exporter and home to a rapidly expanding domestic cloud-native development ecosystem, India's enterprise technology landscape is characterised by a high concentration of workloads on a small number of global cloud platforms, emerging regulatory requirements around data localisation and financial sector cloud governance, and a workforce of cloud practitioners whose awareness of portability risk varies widely across experience levels. These conditions make the Indian IT context both analytically distinctive and practically important for understanding the real-world consequences of architectural lock-in.

This paper presents the findings of a structured empirical investigation designed to address four specific research questions: how cloud service model abstraction correlates with vendor lock-in severity; how practitioner experience relates to portability awareness; whether serverless platform choice influences perceived benefits; and whether systematic adoption of open portability standards and tools reduces perceived lock-in risk. By grounding these questions in primary survey data collected from 306 Indian IT professionals and triangulating with a systematic review of technical and policy literature, the study aims to produce findings that are both academically rigorous and practically actionable.

The remainder of this paper is organised as follows. Section 2 situates the study within the existing literature on cloud portability, vendor lock-in, and serverless computing. Section 3 describes the research methodology. Section 4 presents and interprets the empirical findings. Section 5 discusses the implications of the findings for practitioners, providers, and policy makers. Section 6 concludes the paper with directions for future research.

2. Literature Review

2.1 Cloud Computing Architecture and Service Models

The canonical definition of cloud computing, established by the United States National Institute of Standards and Technology, describes it as a model enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources that can be provisioned and released with minimal management effort or service provider interaction (Mell and Grance, 2011). The three service delivery models Infrastructure-as-a-Service, Platform-as-a-Service, and Software-as-a-Service represent progressively higher levels of abstraction over the underlying hardware and system software, with corresponding differences in the degree to which the consuming organisation retains control over its deployed environment.

Armbrust et al. (2010) identified vendor lock-in as one of the most significant obstacles to confident cloud adoption by large enterprises, noting that the difficulty of migrating data and computation out of a provider's environment creates a bargaining asymmetry that undermines the leverage organisations typically enjoy in technology procurement. This observation, made at the earliest stage of commercial cloud maturity, has proven remarkably prescient: subsequent empirical research has consistently identified lock-in risk as a primary governance and strategic concern for cloud-adopting organisations (Opara-Martins et al., 2016).

2.2 Cloud Portability and Standards

Petcu (2011) proposed one of the earliest systematic frameworks for understanding cloud portability, distinguishing between data portability, application portability, platform portability, and infrastructure portability as four analytically distinct dimensions of the overall migration challenge. This taxonomy has informed subsequent literature and continues to provide a useful conceptual scaffold for empirical investigations. The Topology and Orchestration Specification for Cloud Applications, developed by OASIS, represents the most

comprehensive attempt to address application portability through a standardised, declarative description of cloud application topologies. Despite its technical sophistication, TOSCA adoption has remained limited in practice a pattern that this study's empirical findings corroborate and extend.

The containerisation ecosystem, centred on Docker and Kubernetes, has emerged as the most practically effective mechanism for achieving portable cloud deployments. Bernstein (2014) noted that container-based deployment packages address application portability at the runtime level in a way that proprietary virtual machine images do not, while Kubernetes has progressively expanded its scope to provide a degree of platform portability across cloud providers. Terraform, HashiCorp's infrastructure-as-code tool, similarly enables infrastructure portability by allowing multi-cloud provisioning from a single declarative configuration.

2.3 Serverless Computing and Lock-In Intensification

Jonas et al. (2019), in their Berkeley view of serverless computing, identified the portability challenge as one of the defining open problems of the serverless paradigm. They noted that while serverless functions themselves are generally portable at the code level, the surrounding ecosystem event sources, managed integrations, invocation patterns, observability tooling is almost entirely provider-specific, creating a form of lock-in that operates at the architectural rather than the code level. Leitner et al. (2019) confirmed through a mixed-method study of industrial FaaS adoption that practitioners consistently identified provider lock-in as a significant concern, particularly for organisations considering multi-cloud strategies.

Spillner (2017) explored programmatic approaches to transforming Python applications into provider-agnostic FaaS deployments, highlighting both the technical feasibility and the current limitations of such approaches. The cold start latency problem wherein the first invocation of a serverless function incurs additional latency due to container initialisation has been the subject of extensive measurement research (Wang et al., 2018), with provider-specific cold start characteristics adding another dimension of platform dependency that complicates workload migration.

2.4 The Indian Cloud Computing Context

The Indian cloud computing market has experienced compound annual growth rates significantly exceeding global averages throughout the early 2020s, driven by a combination of government digital transformation initiatives, the expansion of domestic start-up ecosystems, and continued growth in IT services exports (NASSCOM, 2023). The Reserve Bank of India's cloud adoption guidelines for regulated financial entities introduce specific requirements around data sovereignty, auditability, and exit planning that create portability obligations not found in most other national regulatory frameworks. The Government of India's MeghRaj initiative seeks to establish a national cloud infrastructure aligned with public sector data governance requirements, introducing a further dimension of portability complexity as organisations must maintain the capability to migrate between commercial clouds and government-operated infrastructure.

Despite this policy context, empirical research specifically examining the portability awareness and lock-in experiences of Indian IT practitioners has been limited. Most existing studies have focused either on cloud adoption rates and drivers or on global practitioner samples that do not adequately represent the distinctive regulatory, vendor concentration, and talent pipeline characteristics of the Indian IT environment. The present study seeks to address this gap directly.

3. Research Methodology

3.1 Research Design

This study adopts a pragmatist philosophical stance and an explanatory sequential mixed-method research design. The primary quantitative component a structured survey administered to Indian IT professionals generates the empirical evidence against which the four hypotheses are tested. The secondary qualitative component, comprising a systematic review of technical literature, industry reports, and regulatory documents, contextualises and extends the interpretation of the quantitative findings. This design is consistent with the recommendations of Creswell and Plano Clark (2017) for research involving both measurement and explanation objectives.

3.2 Sample and Data Collection

The target population comprised cloud computing practitioners employed in organisations operating within India's information technology sector. A purposive sampling strategy was employed, with respondents recruited through professional networking platforms, IT industry associations, and academic institutional networks. A total of 318 questionnaire responses were received, of which 306 were retained after removing incomplete or internally inconsistent responses — a retention rate of 96.2 per cent. The survey was administered between January and March 2024. Table 1 presents the demographic composition of the sample.

Demographic Variable	Category / Value	n (%)
Gender	Male	218 (71.2%)
	Female	82 (26.8%)
	Other / Prefer not to disclose	6 (2.0%)
Experience	Less than 1 year	30 (9.8%)
	1–2 years	71 (23.2%)
	3–7 years	116 (37.9%)
	8–12 years	56 (18.3%)
	More than 12 years	33 (10.8%)
Organisation Type	IT Services / Consulting	134 (43.8%)
	Software Product / Tech	89 (29.1%)
	Enterprise IT Dept.	52 (17.0%)
	Start-up / Early Stage	19 (6.2%)
	Academic / Government	12 (3.9%)
Primary Platform	Amazon Web Services (AWS)	135 (44.1%)
	Microsoft Azure	98 (32.0%)
	Google Cloud Platform	45 (14.7%)
	Multi-Cloud / Hybrid	18 (5.9%)
	Other / Private Cloud	10 (3.3%)

Table 1: Demographic Profile of Survey Respondents (n = 306)

3.3 Research Instrument

The structured questionnaire comprised five sections. Section A collected demographic information. Section B assessed cloud service model adoption and usage intensity using a five-point frequency scale. Section C measured

cloud portability awareness and vendor lock-in experience using a seven-item Likert scale (Cronbach's α = 0.847) and categorical questions about lock-in severity by service model. Section D examined serverless computing adoption, perceived benefits (seven items, Cronbach's α = 0.891), and portability challenges. Section E assessed open standards adoption and perceived effectiveness of portability tools. The instrument was pilot-tested with fourteen IT professionals and refined prior to full administration.

3.4 Hypotheses and Statistical Tests

Four null hypotheses were formulated and tested at the 0.05 significance level using SPSS Version 26.0. H01 proposed no significant association between cloud service model and vendor lock-in severity, tested using Pearson chi-square analysis.

H02 proposed no significant relationship between years of cloud experience and portability awareness level, also tested using chi-square.

H03 proposed no significant difference in perceived serverless benefits across AWS Lambda, Azure Functions, and Google Cloud Functions users, tested using one-way ANOVA with Tukey HSD post-hoc comparisons.

H04 proposed no significant correlation between open portability standards adoption and perceived lock-in risk reduction, tested using Pearson product-moment correlation.

4. Results and Findings

4.1 Cloud Service Model and Vendor Lock-In (H01)

The first hypothesis examined the relationship between the cloud service model primarily used by respondents and the degree of vendor lock-in they reported experiencing. Cross-tabulation of service model against lock-in severity category low, moderate, and high revealed a clear and statistically significant pattern. Among IaaS-primary users, 30.7 per cent reported low lock-in, compared with only 11.0 per cent of SaaS-primary users. Conversely, high lock-in was reported by 20.4 per cent of IaaS users but 50.7 per cent of SaaS users. Pearson chi-square analysis returned $\chi^2(6) = 38.742$, $p < 0.001$, with Cramér's V = 0.252 indicating a moderate effect size. The null hypothesis H01 was therefore rejected. These results empirically confirm the portability-abstraction inverse relationship: as the cloud service model abstracts further from the underlying infrastructure, the degree of vendor dependency experienced by the organisation increases systematically.

Service Model	Low Lock-In	Moderate Lock-In	High Lock-In	Total
Primarily IaaS	42 (30.7%)	67 (48.9%)	28 (20.4%)	137
Primarily PaaS	11 (13.8%)	32 (40.0%)	37 (46.2%)	80
Primarily SaaS	8 (11.0%)	28 (38.4%)	37 (50.7%)	73
Mixed / Multiple	4 (25.0%)	8 (50.0%)	4 (25.0%)	16
Total	65 (21.2%)	135 (44.1%)	106 (34.6%)	306

Table 2: Cross-Tabulation — Cloud Service Model × Vendor Lock-In Severity ($\chi^2(6) = 38.742$, $p < 0.001$, Cramér's V = 0.252)

4.2 Experience Level and Portability Awareness (H02)

The second hypothesis tested whether practitioner experience level has any bearing on cloud portability awareness. Survey data showed that 84.8 per cent of respondents with more than twelve years of cloud experience were categorised as high-awareness, compared with only 6.7 per cent of those with less than one year of experience. This gradient was steep, consistent, and statistically robust: $\chi^2(8) = 54.618$, $p < 0.001$, Cramér's V =

0.299 (moderate effect). H02 was rejected. The finding carries significant practical implications: since architectural decisions with long-term lock-in consequences are frequently made early in a project lifecycle often by relatively junior practitioners the experience-awareness gap means that the practitioners most likely to be making these decisions are the least well-equipped to recognise their portability implications. This dynamic, which this paper terms the portability awareness gap, represents a workforce development challenge of considerable importance for the Indian IT industry.

4.3 Platform Differences in Serverless Benefits (H03)

Of the 306 total respondents, 205 indicated that their organisations had deployed or evaluated serverless computing. These respondents rated the perceived benefits of serverless on a seven-item five-point Likert scale. Mean benefit scores for AWS Lambda users ($M = 4.12$, $SD = 0.68$) and Azure Functions users ($M = 4.04$, $SD = 0.71$) were broadly similar, while GCP Cloud Functions users reported a meaningfully lower mean score ($M = 3.61$, $SD = 0.84$). One-way ANOVA confirmed a statistically significant difference across the three groups: $F(2, 190) = 6.834$, $p = 0.001$. Tukey HSD post-hoc tests revealed that the AWS versus GCP comparison ($p = 0.001$) and the Azure versus GCP comparison ($p = 0.018$) were each significant, while the AWS versus Azure comparison was not ($p = 0.812$). H03 was rejected. The lower benefit perception among GCP users likely reflects GCP's smaller practitioner community in India, its less mature third-party ecosystem relative to AWS and Azure, and potentially a selection effect whereby GCP deployments in the sample were concentrated in more complex or experimental use cases.

Platform	n	Mean	SD	95% CI	F	p
AWS Lambda	104	4.12	0.68	[3.99, 4.25]	6.834	0.001
Azure Functions	68	4.04	0.71	[3.87, 4.21]		
GCP Cloud Functions	21	3.61	0.84	[3.23, 3.99]		
Total	193	4.01	0.74	[3.91, 4.12]		

Table 3: One-Way ANOVA — Platform × Perceived Serverless Benefits ($F(2,190) = 6.834$, $p = 0.001$)

4.4 Portability Standards Adoption and Lock-In Reduction (H04)

The fourth analysis examined whether the degree to which organisations systematically adopt open portability standards and cloud-agnostic tooling is associated with a reduction in their perceived vendor lock-in risk. A composite portability standards adoption score was constructed from respondents' reported use and effectiveness ratings of eight tools and standards: Docker/containerisation, HashiCorp Terraform, Kubernetes, Serverless Framework, Open Virtualisation Format, TOSCA, CNCF CloudEvents, and OCCI. Pearson correlation between this composite score and the respondent's self-reported lock-in risk reduction score returned $r = 0.487$, $p < 0.001$. H04 was rejected. The correlation, while moderate rather than strong, carries important practical significance: it empirically validates the intuition that adopting open portability tooling delivers measurable benefits not merely aspirational ones in terms of reduced lock-in exposure.

Table 4 summarises the adoption rates and mean effectiveness ratings of each portability tool or standard in the sample. The contrast between the adoption and effectiveness of practical open-source infrastructure tools and those of formal standards bodies is striking: Docker containers (71.2% adoption, $M = 4.14$ effectiveness) and HashiCorp Terraform (62.7%, $M = 4.08$) significantly outperform TOSCA (14.4%, $M = 3.18$) and OCCI (7.8%, $M = 2.96$). This divergence suggests that the cloud portability standards ecosystem has, to date, failed to achieve the practitioner traction that would be needed to influence architectural decisions at scale.

Tool / Standard	Adoption Rate	Mean Effectiveness	Practical Implication
Docker / Containerisation	71.2%	4.14 / 5	Highest adoption; most effective
HashiCorp Terraform (IaC)	62.7%	4.08 / 5	Strong adoption; proven effectiveness
Kubernetes	48.4%	3.97 / 5	Growing adoption; high effectiveness
Serverless Framework	28.1%	3.74 / 5	Moderate adoption; useful for FaaS
Open Virtualisation Format (OVF)	22.5%	3.42 / 5	Lower adoption; moderate effectiveness
TOSCA	14.4%	3.18 / 5	Low adoption; limited practical uptake
CNCF CloudEvents	11.3%	3.31 / 5	Emerging; relevant for serverless
OCCI	7.8%	2.96 / 5	Very low adoption; limited awareness

Table 4: Adoption Rates and Effectiveness Ratings of Cloud Portability Tools and Standards

4.5 Serverless Portability Challenges

Respondents with serverless deployment experience were asked to rate the severity of five specific portability challenges using a five-point scale. Provider-specific event trigger formats and managed service integrations emerged as the two most severe challenges, rated as major or significant by 68.3 per cent and 64.4 per cent of respondents respectively. Cold start latency which, while a performance rather than a portability issue per se, complicates provider-neutral architectural planning was rated as a major or significant challenge by 58.2 per cent. These findings align with the theoretical characterisation of serverless lock-in offered by Jonas et al. (2019) and confirm that the portability challenge in serverless environments is primarily architectural rather than code-level in nature.

5. Discussion

5.1 The Portability-Abstraction Inverse Relationship

The empirical confirmation of the portability-abstraction inverse relationship across all four service model categories has important architectural implications. Organisations seeking to preserve future optionality the ability to migrate, renegotiate, or multi-source their cloud services face a fundamental tension between the operational benefits of managed services and the strategic costs of the lock-in those services create. This is not a new observation in the abstract, but this study's empirical quantification of the relationship, grounded in a substantial Indian practitioner sample, provides a concrete evidential basis for architectural governance policies that explicitly assess the portability implications of managed service adoption.

The finding that high lock-in is reported by more than half of SaaS-primary users 50.7 per cent is particularly noteworthy given that SaaS applications are frequently adopted outside formal IT governance processes, through departmental purchasing decisions that bypass architectural review. This suggests that organisations seeking to manage portability risk must extend their governance frameworks beyond infrastructure and platform-layer decisions to encompass the full scope of cloud service consumption.

5.2 The Serverless Portability Paradox

Serverless computing presents what might be termed a portability paradox: the paradigm that most aggressively abstracts infrastructure concerns from the developer is simultaneously the one that most deeply embeds the application in provider-specific ecosystems. When a developer writes a serverless function that responds to an Amazon S3 bucket notification, processes the event using an AWS SDK client, writes results to a DynamoDB table, and triggers a Step Functions workflow for downstream processing, the function's code may be portable in isolation, but the application as a whole is not. Each managed service integration represents a tight coupling to AWS's proprietary ecosystem that has no standard-compliant equivalent on Azure or GCP.

This architectural reality which the study's respondents confirm through their high severity ratings for event trigger and managed integration lock-in suggests that the industry needs a different conversation about serverless portability than the one currently taking place. Rather than focusing primarily on code-level portability frameworks such as the Serverless Framework, the more productive intervention may be to establish standard inter-provider event payload formats (the CNCF CloudEvents specification represents a nascent effort in this direction) and to encourage architectures that maintain a clear separation between business logic and provider-specific adapter code an approach consistent with hexagonal architecture principles.

5.3 The Standards Adoption Gap and Its Causes

The substantial divergence between the adoption of practical open-source tooling (Docker, Terraform, Kubernetes) and formal standards-body outputs (TOSCA, OCCI) observed in this study warrants careful analysis. The pattern suggests that practitioners make pragmatic rather than principled choices about portability tooling adopting what is immediately useful, well-documented, community-supported, and integrated with their existing workflows rather than what is formally endorsed by standards bodies. TOSCA, for example, has a fifteen-year development history and is technically comprehensive, yet its practitioner adoption rate in this sample was 14.4 per cent far below that of Terraform, which addresses a partially overlapping problem through a more developer-friendly interface.

This observation has implications both for standards bodies and for organisations designing portability strategies. For standards bodies, it suggests that technical correctness is a necessary but not sufficient condition for adoption usability, developer experience, and ecosystem integration are equally important. For organisations, it argues in favour of pragmatic portability strategies centred on the open-source tooling stack that has already achieved broad practitioner adoption, rather than waiting for formal standards to achieve the traction they have thus far not managed.

5.4 Implications for India's Cloud Policy Landscape

The finding that Amazon Web Services commands 44.1 per cent of the sample's primary cloud platform share a concentration that mirrors broader market data for the Indian IT industry has systemic implications that go beyond individual organisational risk management. An industry whose workloads are concentrated on a single provider to this degree is collectively exposed to the commercial, security, and operational risks that any dominant market position creates. India's response to this structural challenge is nascent: the MeghRaj initiative provides a government-operated cloud option, but commercial cloud adoption has proceeded largely without the portability governance frameworks that regulatory environments in the European Union through the Data Act and GDPR data portability provisions have begun to mandate.

The Reserve Bank of India's cloud guidelines represent an important first step in applying portability-aware thinking to regulated sector cloud adoption, requiring financial institutions to maintain documented exit plans and to assess the reversibility of cloud commitments. Extending similar requirements to other regulated sectors, and developing a comprehensive national cloud portability framework that addresses both data portability rights and

technical interoperability standards, would position India's digital infrastructure governance more robustly for the long term.

6. Recommendations

Drawing on the empirical findings and analytical discussion above, this section advances recommendations for three stakeholder groups: cloud consumer organisations, cloud service providers, and policy makers.

#	Recommendation	Rationale	Expected Outcome	Priority
R1	Formalise cloud portability strategy	Only 31.4% have documented strategy despite near-universal lock-in concern	Governed, reviewable portability commitments	High
R2	Adopt portability-first architecture principle	Every managed service adds lock-in; conscious trade-off evaluation needed	Reduced accidental lock-in accumulation	High
R3	Implement Docker-Terraform-Kubernetes stack	Highest adoption and effectiveness in study; proven practical benefits	Measurable lock-in risk reduction	High
R4	Apply hexagonal architecture for serverless	Separates business logic from provider adapters; enables future migration	Reduced serverless ecosystem lock-in	High
R5	Integrate portability training at all levels	Experience-awareness gap: junior practitioners most at risk	Workforce-wide portability literacy	Medium
R6	Providers: reduce egress fees voluntarily	Economic lock-in via egress fees is a structural market distortion	Lower switching cost barrier	High
R7	Providers: support open standard event formats	68.3% cite event trigger lock-in as primary serverless challenge	Reduced FaaS architectural lock-in	Medium
R8	Providers: develop bidirectional migration tools	Current tooling favours inbound migration; exit tooling is weak	Fair competitive switching environment	Medium
R9	MeitY: develop India cloud portability framework	No India-specific portability governance framework currently exists	Data sovereignty and market competition	High
R10	RBI: mandate portability assessment for cloud approvals	RBI guidelines require exit planning; formal portability risk assessment needed	Regulated sector portability governance	High
R11	UGC/AICTE: integrate CNCF/IaC into curricula	Cloud education remains provider-specific; portability literacy gap widens	Graduates equipped for portable cloud architectures	Medium

Table 5: Summary of Recommendations — Stakeholder, Rationale, and Priority

7. Conclusion

This paper has reported the findings of a structured empirical investigation into vendor lock-in, cloud portability, and serverless computing adoption among Indian IT practitioners. Four hypotheses were tested and rejected, yielding findings that collectively advance both the academic understanding of cloud portability dynamics and the practical repertoire available to organisations seeking to manage lock-in risk.

The portability-abstraction inverse relationship the empirical pattern whereby vendor lock-in severity increases systematically from IaaS through PaaS to SaaS provides a useful decision-making framework for architectural governance. Organisations that consciously track where each cloud service sits on this abstraction spectrum, and

that explicitly evaluate the portability cost of managed service adoption, are better placed to preserve the future optionality that strategic agility requires.

The serverless portability paradox the observation that the most infrastructurally abstracted cloud paradigm is simultaneously the most architecturally locked-in represents perhaps the most urgent practical challenge identified by this research. As serverless adoption continues to grow and as organisations build increasingly complex event-driven applications on top of provider-specific FaaS ecosystems, the architectural debt accumulated through unmanaged serverless lock-in will compound. Addressing this challenge requires action at multiple levels: practitioner training, architectural discipline, provider commitment to open event standards, and regulatory frameworks that make portability a design requirement rather than an afterthought.

India's position in the global cloud computing landscape as both a major consumer and a major provider of cloud-enabled IT services gives it both the motivation and the influence to lead in developing governance frameworks that take cloud portability seriously. The empirical evidence presented in this paper provides a concrete foundation for the policy conversations that this leadership will require.

References

[1] Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A., Stoica, I., & Zaharia, M. (2010). A view of cloud computing. *Communications of the ACM, 53*(4), 50–58.

[2] Bernstein, D. (2014). Containers and cloud: From LXC to Docker to Kubernetes. *IEEE Cloud Computing, 1*(3), 81–84.

[3] Creswell, J. W., & Plano Clark, V. L. (2017). *Designing and conducting mixed methods research* (3rd ed.). SAGE Publications.

[4] Leitner, P., Wittern, E., Spillner, J., & Hummer, W. (2019). A mixed-method empirical study of Function-as-a-Service software development in industrial practice. *Journal of Systems and Software, 149*, 340–359.

[5] Mell, P., & Grance, T. (2011). *The NIST definition of cloud computing* (NIST Special Publication 800-145). National Institute of Standards and Technology.

[6] NASSCOM. (2023). *Indian technology sector: Annual strategic review 2023*. NASSCOM.

[7] Opara-Martins, J., Sahandi, R., & Tian, F. (2016). Critical analysis of vendor lock-in and its impact on cloud computing migration: A business perspective. *Journal of Cloud Computing: Advances, Systems and Applications, 5*(1), 1–18.

[8] Petcu, D. (2011). Portability and interoperability between clouds: Challenges and case study. In *Proceedings of the 4th European Conference on Towards a Service-Based Internet* (pp. 62–74).

[9] Reserve Bank of India. (2023). *Master direction on information technology governance, risk, controls and assurance practices* (RBI/2023–24/73). Reserve Bank of India.