mathematics

MDPI

Article

CVL: A Cloud Vendor Lock-In Prediction Framework

Amal Alhosban *, Saichand Pesingu and Krishnaveni Kalyanam

Computer Science and Director Academic Program, College of Innovation and Technology, University of Michigan-Flint, Flint, MI 48502, USA; saichanp@umich.edu (S.P.); krishkal@umich.edu (K.K.)
* Correspondence: alhosban@umich.edu

Abstract: This paper presents the cloud vendor lock-in prediction framework (CVL), which aims to address the challenges that arise from vendor lock-in in cloud computing. The framework provides a systematic approach to evaluate the extent of dependency between service providers and consumers and offers predictive risk analysis and detailed cost assessments. At the heart of the CVL framework is the Dependency Module, which enables service consumers to input weighted factors that are critical to their reliance on cloud service providers. These factors include service costs, data transfer expenses, security features, compliance adherence, scalability, and technical integrations. The research delves into the critical factors that are necessary for dependency calculation and cost analysis, providing insights into determining dependency levels and associated financial implications. Experimental results showcase dependency levels among service providers and consumers, highlighting the framework's utility in guiding strategic decision-making processes. The CVL is a powerful tool that empowers service consumers to proactively navigate the complexities of cloud vendor lock-in. By offering valuable insights into dependency levels and financial implications, the CVL aids in risk mitigation and facilitates informed decision-making.

Keywords: cloud vendor lock-in; dependency analysis; cost evaluation; cloud service providers; risk mitigation

MSC: 68N30

Citation: Alhosban, A.; Pesingu, S.; Kalyanam, K. CVL: A Cloud Vendor Lock-In Prediction Framework. *Mathematics* **2024**, *12*, 387. https://doi.org/10.3390/math12030387

Academic Editors: Zaki Malik and M. Mustafa Rafique

Received: 30 November 2023
Revised: 3 January 2024
Accepted: 22 January 2024
Published: 25 January 2024

1. Introduction

Cloud computing has become an integral part of businesses and personal lives due to its scalability, cost-effectiveness, accessibility, and user-friendliness. Its popularity has led to widespread adoption by people and businesses alike, resulting in significant market growth. By 2022, the market size is expected to reach around USD 490 billion, with annual growth predicted to be 15.7% from 2022 to 2030. This trajectory would result in a market size of USD 1500 billion by the end of the next decade. The rising demand for cloud services is fueled by digital transformation initiatives, the surge in remote work, and the increasing reliance on online services. However, potential factors such as technological breakthroughs, evolving regulations, economic fluctuations, and shifts in business approaches may influence the growth rate.

Cloud computing is essential in contemporary society, as it enables sizable enterprises to lease their hardware infrastructure to smaller businesses that lack the means to procure the hardware independently. This includes an array of services such as storage, servers, databases, networking, analytics, software, and intelligence, all delivered via the Internet. One of the most significant advantages of cloud computing is its inherent scalability, allowing businesses to expand seamlessly while accommodating evolving needs and reducing operational expenses.

In a broad context, cloud computing represents a forward-thinking abstraction of Internet-based computing resources and services, harnessed by system developers to implement intricate web-based systems [1]. In simpler terms, cloud computing leverages

a network of remote servers hosted on the Internet for data storage, management, and processing, supplanting the traditional local server approach. A finer understanding of cloud computing emerges when examining its deployment models and service categories. Four distinct deployment models shape the landscape: private, public, community, and hybrid. A public cloud is universally accessible, whereas a private cloud is owned by a specific entity. The community-based model is shared among multiple entities, while the hybrid model amalgamates elements of the other three. Furthermore, three core cloud service models categorize its functionalities. Software as a Service (SaaS) encapsulates complete software applications complete with user interfaces. Infrastructure as a Service (IaaS) supplies machines, storage, and networking resources that developers manage by incorporating their supportive assets. Platform as a Service (PaaS) furnishes a platform where developers deploy their applications [2].

The cloud computing industry faces several challenges, including data breaches, compliance with regulatory mandates, a lack of IT expertise, cloud migration, unsecured APIs, insider threats, and many others. These challenges usually fall into three main categories: security and privacy, data protection, and vendor lock-in.

1.1. Security and Privacy

Ensuring the security of cloud computing requires a comprehensive approach that takes into account the persistent growth of challenges that come with technological advancements. As technology evolves, it becomes increasingly important to provide robust protection. Protecting digital environments requires mitigating potential threats and preempting attacks to build user confidence in using services, platforms, and software. The level of security provided by a cloud service is directly linked to user adoption and acceptance. Unlike traditional systems where physical access often leads to vulnerabilities, the remote nature of cloud computing changes this dynamic. However, the increasing sophistication of hackers remains a significant challenge. To enhance cloud computing security, a combination of strategies is essential. This includes implementing strict authentication mechanisms, robust encryption protocols, and vigilant intrusion detection systems. In addition, proactive monitoring, the timely patching of vulnerabilities, and continuous security audits are critical. Emphasizing user education and awareness regarding best security practices can also play a crucial role in fortifying the overall cloud environment. By addressing security concerns comprehensively, cloud providers and users can contribute to a safer ecosystem that safeguards against threats and creates a climate of trust, enabling cloud services to thrive despite evolving cyber threats [3].

1.2. Data Protection

Organizations must ensure that their data comply with regulations such as GDPR, HIPAA, and industry-specific standards. The location of data storage and processing can impact compliance [4]. Given that computing takes place over the Internet, the eventuality of a data breach is a concern that necessitates proactive measures to safeguard both businesses' and their customers' data. While striving to strike a balance amidst various challenges can be complex, prioritizing prevention is advisable. Mitigating the occurrence of breaches is key to pre-empting a potential fallout. Furthermore, the onus often falls on the cloud computing provider to furnish a significant portion of the security framework. However, this dynamic can limit the extent of control businesses possess over the security protocols. Hence, a strategic approach involves collaborative efforts between stakeholders to ensure a robust security posture while navigating the inherent complexities of cloud computing. By adopting a foresighted and preventative approach, businesses can mitigate risks, enhance security, and establish a resilient cloud environment that instills trust and safeguards data integrity. As of 2023, the Cost of a Data Breach report jointly conducted by IBM and the Ponemon Institute reveals that the average expense incurred due to a data breach has surged to an all-time high of USD 4.45 million [5].

Our paper focuses on addressing the challenge of vendor lock-in in cloud computing. Our goal is to develop a framework that goes beyond current considerations and predicts future instances of vendor lock-in. This forward-looking approach will provide stakeholders with anticipatory insights and enable proactive strategies to mitigate the risks associated with vendor lock-in. By predicting future occurrences, our research seeks to empower organizations with the knowledge needed to make informed decisions, ultimately fostering a more resilient and adaptable cloud computing environment. The main contributions of our work are as follows:

- The CVL offers a detailed analysis of the cost structures associated with different cloud service providers. This includes pricing models, charges for resource consumption, and any additional fees related to data transfer, storage, and computing resources. By quantifying the financial impact of each provider, our framework provides service consumers with a valuable tool to effectively compare and contrast the economic aspects of available offerings, thereby improving decision-making processes.
- The CVL assesses the degree of reliance on each cloud service provider. Higher reliance levels indicate potential challenges in transitioning to alternative providers in the future. This provides critical insights for strategic planning and risk mitigation.
- The CVL has introduced a new methodology that aims to assign rankings to cloud service providers. This ranking system is based on the data collected regarding the cost of services and the level of dependency on these services. Providers with lower service costs and reduced dependency levels will receive higher rankings. This new system will make it easier for consumers to make informed decisions when selecting optimal cloud service solutions.
- The CVL enables service consumers to make informed decisions that align with their organizational goals. Choosing a cloud service provider with lower dependency levels not only minimizes the risk of vendor lock-in but also provides financial benefits by potentially reducing overall service costs.
- The CVL enables service consumers to manage and mitigate challenges associated with vendor lock-in. By guiding the selection of providers with lower dependency levels, organizations gain flexibility, simplifying the process of adapting to evolving business needs.

This paper explores the challenges associated with vendor lock-in in the realm of cloud computing, as discussed in Section 2. Section 3 provides an overview of related work in the field. Moving forward, Section 4 introduces a proposed method, while Section 5 outlines the experimental framework utilized to assess the effectiveness of this approach. The conclusions drawn in the final section emphasize the importance of mitigating vendor lock-in to enhance the widespread adoption of cloud computing solutions across various corporate applications and services.

2. Vendor Lock-In

Vendor lock-in is a situation in cloud computing where a customer becomes heavily dependent on a specific cloud provider due to various reasons such as technical aspects, contract terms, or other factors. This dependency can result in limited options for the customer to switch to another provider, even if the current provider is not performing well. This creates difficulties for customers who want to switch to a different provider, as it involves significant obstacles, expenses, and disruptions. Vendor lock-in can occur when a cloud provider uses proprietary technologies, formats, or interfaces that are not easily interoperable with other providers, making it difficult for the customer to migrate their applications and data. It can also result from exclusive licensing terms, complex integration requirements, or custom configurations that tie the customer to the specific provider's ecosystem. This lack of portability and flexibility can limit the customer's ability to adapt to changing needs or take advantage of competitive offerings, potentially impacting cost-effectiveness and innovation.

2.1. Vendor Lock-In Scenario

Company TechWay, a rapidly growing Software as a Service (SaaS) provider, initially selected Cloud Provider XYZ to host their application and oversee their cloud infrastructure. They were drawn by competitive rates, dependable services, and most features aligned with their needs.

Over time, TechWay integrated its application with Cloud Provider XYZ's offerings, relying on databases, serverless functions, and AI services for data processing. However, challenges arose when their evolving needs exceeded Cloud Provider XYZ's AI services. They turned to Cloud Provider ABC, which offered advanced machine learning options like natural language processing and image recognition (see Figure 1).

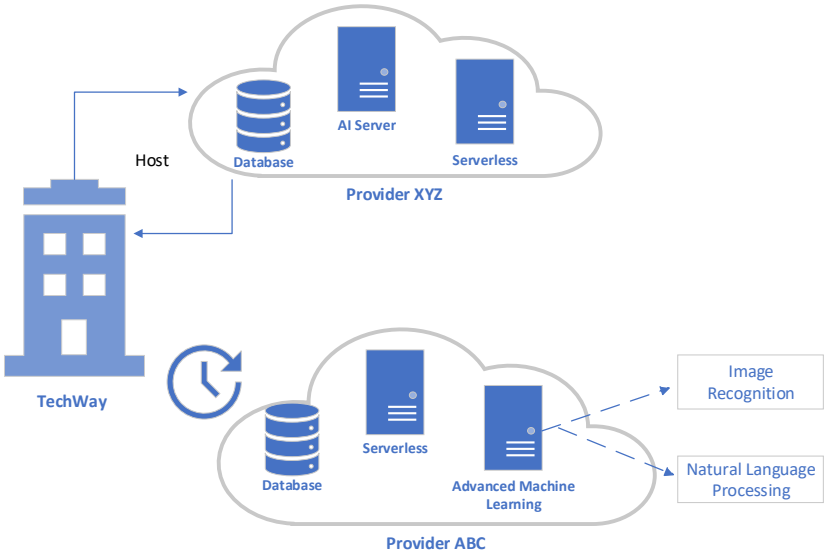

Figure 1. TechWay company scenario.

At this point, the vendor lock-in issue emerged. Shifting from Cloud Provider XYZ to Cloud Provider ABC would be intricate and time-intensive. They would need to modify their code, adapt data formats, and restructure workflows to align with Cloud Provider ABC. Furthermore, high data egress charges by Cloud Provider XYZ added to the migration expenses.

TechWay also faced the hurdle of their proficient developers specializing in Cloud Provider XYZ's tools. Adapting to Cloud Provider ABC would require training or hiring new talent familiar with their ecosystem.

Cloud computing is marked by the arduous and costly process of transferring applications and data to alternate providers. Cloud software vendors employ various strategies to bind customers, such as creating systems that are incompatible with other vendor software, utilizing closed architectures or proprietary standards that lack interoperability with other applications, and imposing exclusive licensing conditions [6]. This practice dissuades organizations from embracing cloud technology, posing a formidable obstacle that demands substantial efforts to surmount [7].

As elucidated by [7], the mounting market demand and the quest for increased customer engagement exert pressure on cloud providers to prioritize interoperability—a direct advantage that circumvents the pitfalls of vendor lock-in. Prior studies have largely explored interoperability issues and the concerns tied to vendor lock-in. Numerous standardization solutions have emerged to enhance interoperability [8,9]. However, limited research has been devoted to a comprehensive investigation of vendor lock-in review and its implications for cloud computing adoption.

2.2. Vendor Lock-In Types

In cloud computing, vendor lock-in appears in different ways, such as technical lock-in, data lock-in, service lock-in, certification lock-in, contract lock-in, economic lock-in, and

network lock-in (Figure 2). The technical lock-in occurs when a cloud service provider offers proprietary technologies, APIs, or data formats that are not easily transferable to other providers. Customers become dependent on these unique features, making it challenging to migrate to alternative platforms without significant redevelopment effort. For example, when the service consumer is using database solutions with cloud-specific functions.

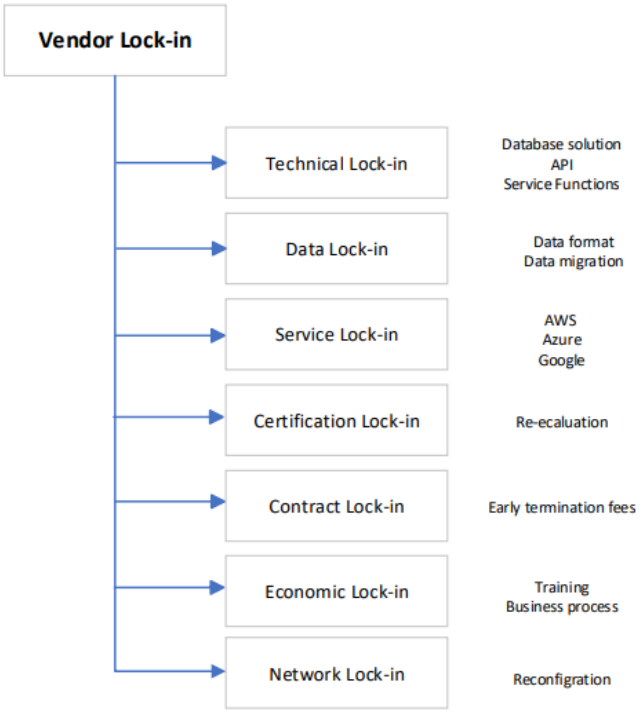

Figure 2. Vendor lock-in Ttypes.

In data lock-in, the data storage formats and structures are owned by the cloud providers. Cloud consumers should store their data in the specified formats, which makes it difficult to move these structured data to another service provider or to a different infrastructure. The migration of service consumer data will need transformation, which will increase the data migration complexities. Data transformation will lead to high costs and time-consuming processes. Service lock-in occurs when a service provider offers a wide range of services, such as analytical services, serverless computing, and machine learning. Service consumers who integrate these services into their applications will find it challenging to switch to another cloud provider because of the tight coupling with the provider's system. For example, AWS Lambda functions are not easily transferable to Azure Functions or other cloud providers, such as Google. In certification lock-in, some industries require specific certifications or standards. The service consumer should follow the roles and standards listed in the cloud provider certifications. Switching service providers requires a re-evaluation of the roles and standards of the certification, which will be costly and time-consuming. Contract lock-in appears in many cases especially when it is a long-term contract with cloud providers. Most of the contracts have penalties and/or termination fees for early termination, making it financially challenging for service consumers to switch to another service provider. Over time, service consumers may invest heavily in a specific cloud provider's ecosystem, which can lead to economic lock-in. This includes the direct costs associated with cloud services, the training of staff, the development of cloud applications, and the adoption of specific tools. The last type is network lock-in, which appears when the cloud providers offer a variety of networking services, and service consumers configure their network infrastructure to work with a particular provider's services. Migrating to a new provider will require network re-configuring.

Mitigating cloud vendor lock-in involves careful planning, the use of open standards and technologies when possible, and adopting strategies like multi-cloud or hybrid cloud architectures. It is crucial for organizations to weigh the benefits of cloud services against the potential lock-in risks when making cloud-related decisions.

This paper's contribution establishes a cornerstone for prospective analyses and evaluations concerning the influence of vendor lock-in on the uptake of corporate cloud computing applications and services.

3. Related Work

As we discussed before, vendor lock-in is a situation where a customer becomes dependent on a particular vendor's products or services to the extent that it becomes challenging to switch to an alternative vendor without extra costs or disruption. In this section, we discussed the related work about vendor lock-in by studying the types of vendor lock-in, the risks and costs of vendor lock-in, and vendor lock-in in specific industries [8–10]. Table 1 illustrates the features and challenges of different cloud services.

In [11], the authors discussed the issues of vendor lock-in in cloud computing and offered solutions. They also addressed concerns about moving cloud services, emphasizing security. Vendor lock-in arises due to differences in vendor technologies and a lack of interoperability. They conclude that preventing vendor lock-in or reducing its effects aligns with maintaining interoperability across various cloud computing systems and services.

The authors of [6] claimed that to build trust in cloud adoption and avoid vendor lock-in, it is vital to make it easy for users to switch cloud providers with their data. Just having a copy of the data is not enough if they are in a proprietary format. So, not only should users have access to their data, but they should also be in a format with a publicly available specification. Giving organizations control over their data and enabling smooth transitions between cloud providers is key to establishing trust and a vendor-neutral cloud environment.

The study in [7] explored the solutions related to interoperability, security, portability, and governance in the cloud. Concerning interoperability, they categorized the solutions into three groups: standardization bodies, industry solutions, and brokering and management.

While cloud providers typically ensure data's high availability and reliability, there are scenarios where a cloud provider's client may encounter difficulties accessing their data. One such scenario arises when the issue of vendor lock-in manifests itself. In [12], Razavian et al. studied how the vendor lock-in problem prevents organizations from migrating toward cloud storage by investigating how system parameters (i.e., failure rate, redundancy ratio, and the number of blocks) impact data accessibility. In this paper, they examined the dispersion of a large file across multiple cloud providers as a solution to address the issue of vendor lock-in.

The goal of the paper in [13] was to tackle vendor lock-in by using a unique approach. The authors' approach combined a versatile infrastructure setup using the Constructs Programming Model with the creation of abstraction layers that manage the integration with specific service providers. The proposed framework simplified the complexities of various serverless platforms, which allows developers to focus on their specific functions. The framework evaluation involved deploying a benchmark application on various cloud providers to showcase how easy and flexible the framework is.

There are many proposed techniques to solve the lock-in problem. In [14], the authors proposed a multi-cloud architecture. A multi-cloud architecture involves utilizing the services of multiple cloud service providers simultaneously to host different aspects of an organization's IT infrastructure. This approach offers several advantages, including reducing vendor lock-in, enhancing redundancy and resilience, optimizing costs, ensuring compliance with data regulations, improving performance, fostering innovation, and simplifying disaster recovery and business continuity planning. It also enables organizations

to deliver specialized services by leveraging the strengths of different cloud providers. In essence, multi-cloud architecture provides flexibility, autonomy, and adaptability in cloud strategy, making it an appealing approach in today's dynamic cloud computing landscape. Multi-cloud architectures allow for greater flexibility when it comes to delivering specialized services. Numerous cloud service providers each excel in a distinct set of specialized services. Organizations can select the most suitable provider for each individual use case, thereby capitalizing on distinctive advantages. For instance, one service provider may offer sophisticated AI capabilities, whereas another may focus on Internet of Things (IoT) services. Because of this flexibility, businesses can tailor a cloud environment that is high-performing, efficient, and cost-effective to meet the varied requirements of their operations. A multi-cloud architecture gives businesses the ability to capitalize on the benefits of cloud computing while minimizing their dependence on a single vendor and the risks that come along with it. As a result, it is becoming an increasingly appealing strategy for effectively navigating the constantly changing landscape of the cloud computing industry. Implementing a multi-cloud architecture introduces several challenges: managing resources across multiple providers requires specialized orchestration, monitoring, and governance tools and skills; ensuring smooth data integration and interoperability can be complex, especially when moving data between various environments; while cost optimization offers benefits, it demands meticulous cost monitoring and management to prevent overspending and; to maintain data integrity and regulatory compliance, organizations must establish consistent security and compliance practices across multiple cloud providers [15].

In the work in [9], containerization and Kubernetes were instrumental in addressing the challenge of cloud vendor lock-in in modern cloud computing. Containerization, exemplified by Docker, encapsulates applications and their dependencies into portable, self-contained units called containers. This abstraction reduces reliance on the underlying infrastructure, enabling applications to run consistently across multiple cloud providers. Developers can create and test containers locally, promoting portability and mitigating vendor lock-in by facilitating easy migration between providers or between cloud and on-premises environments. Kubernetes, an open-source container orchestration platform, complements containerization by simplifying the management and scaling of containerized applications. It abstracts infrastructure complexities and offers automated load balancing, scalability, and self-healing, ensuring consistent performance across diverse cloud environments. The combination of containerization and Kubernetes offers several advantages, including vendor-agnostic deployments, streamlined migration processes, choice and flexibility in cloud service selection, improved scalability, and cost optimization through the continuous evaluation of cloud pricing.

Table 1. Features and challenges of different cloud services.

Reference	Features	Challenges
[14,16,17]	Reduces vendor reliance by using multiple cloud providers	Management complexity due to multiple vendors
[18–20]	Enhances redundancy and ensures business continuity	Data integration challenges when moving between clouds
[21,22]	Improves robustness, as workloads can migrate during outages	Resource and cost optimization across providers
[23,24]	Enables cost optimization by selecting the best provider	Security and compliance management across vendors
[9,25]	Facilitates compliance with regulations by choosing locations	Operations and administration complexities
[26,27]	based on data residency requirements	Migration of applications to different clouds
[16,28,29]	Promotes innovation by driving competition among providers	Training and capability development for multi-cloud
[30–32]	Facilitates disaster recovery through data replication	Continual assessment and adaptation of cloud strategy
[33–35]	Portability of applications using containers	Data storage and database management in multi-cloud

The authors of [26] proposed that serverless computing is a powerful strategy for addressing cloud vendor lock-in concerns, offering organizations a flexible and vendor-agnostic approach to cloud application development and deployment. It abstracts much of the underlying infrastructure management, freeing developers from vendor-specific constraints. Serverless platforms, like AWS Lambda, Azure Functions, and Google Cloud Functions, enable developers to focus solely on writing code, promoting portability and compatibility with a multi-cloud strategy. Serverless applications typically follow an event-driven architecture, responding to triggers or events from various sources, making them cloud-agnostic and interoperable. Minimal vendor-specific code, often written in popular programming languages, facilitates migration between cloud providers and deployment on an on-premises infrastructure. Cost-effectiveness is another key benefit of serverless computing, with organizations billed based on actual execution time and resource consumption, reducing costs and minimizing financial impact when switching providers. Automatic scalability and load balancing ensure efficient traffic adaptation, regardless of the hosting cloud. The serverless ecosystem includes cross-cloud development and deployment tools that abstract cloud-specific details, enhancing portability. Serverless architectures are compatible with multi-cloud strategies, offering redundancy and resilience by deploying functions across multiple providers or an on-premises infrastructure.

Serverless computing platforms provide event bridges and connectors for integration with various cloud services, APIs, and data sources, simplifying vendor-specific integrations and promoting interoperability. Despite its portability and autonomy, serverless functions can still interact with cloud-specific services as needed, providing flexibility in selecting cloud services. Rapid deployment and updates are a primary advantage of serverless computing, allowing organizations to iterate on applications and respond to changing requirements, irrespective of the hosting cloud provider. Careful planning is essential for factors like data storage and databases to ensure complete portability. Additionally, prioritizing monitoring, security, and compliance is crucial when adopting serverless architectures across multiple cloud providers or environments for an effective cloud strategy. The last method is the one proposed by [28]. They were establishing common standards, utilizing containerization technologies, employing cloud-neutral management tools, and adopting hybrid or multi-cloud architectures as key strategies for addressing vendor lock-in concerns in the cloud computing industry. Notable standards like OpenStack and Kubernetes promote interoperability, while containerization technologies like Docker and Kubernetes enhance portability across various cloud environments. Cloud-neutral tools like Terraform and Ansible abstract cloud provider specifics, ensuring infrastructure consistency. Multi-cloud APIs and platforms simplify interaction with multiple cloud services through unified interfaces, reducing adaptation efforts. Hybrid and multi-cloud architectures increase flexibility and resilience by distributing workloads. Cloud brokerage services act as intermediaries, optimizing multi-cloud deployments. Open-source databases offer vendor-independent alternatives, and data portability using open standards is crucial. Contractual agreements should include clear data extraction and ownership terms, and the continuous evaluation and monitoring of cloud strategies is essential to minimize vendor lock-in risks. In summary, most of the related work tried to solve one specific issue that comes as a result of vendor lock-in (see Table 2).

Table 2. Different vendor lock-in techniques.

Technique	Benefits
Multi-Cloud Architecture	• Reduces reliance on a single vendor • Enhances redundancy • Increases robustness • Selects cost-effective providers • Uses multi-cloud to meet data residency • Simplifies disaster recovery planning • Has global reach
Containerization and Kubernetes	• Multiple cloud platforms • Consistency • Scale applications • Cut costs
Standardization and Interoperability	• Standards ensure compatibility • Cloud-neutral management tools • Cloud-independent connectors • Hybrid architectures • Data transfer
Serverless Computing	• Inherently portable • Pay-per-use pricing • Rapid deployment

4. Cloud Vendor Lock-In Prediction Framework (CVL)

In this section, we will introduce our solution to tackle the issue of vendor lock-in. We have developed a method called the cloud vendor lock-in prediction framework (CVL). The approach taken by the CVL involves evaluating the level of dependence between the service provider and the service consumer. This is followed by predicting the risks associated with entering into a contract with that particular provider. Additionally, the CVL conducts an extensive cost analysis for each provider, which is customized to meet the specific requirements of the service consumer.

The CVL addresses several critical factors associated with assessing vendor lock-in challenges in the cloud computing domain. Firstly, it evaluates the comprehensiveness of a cloud service provider's service offerings and the degree of dependency an organization may develop on specific features or technologies. The framework assesses the interoperability of services across different cloud platforms, examining how easily data and applications can be migrated or integrated with alternative providers. Moreover, the CVL considers the pricing models and contractual terms. It also assesses the level of customization and adaptability of applications, ensuring that organizations can maintain flexibility and control over their IT infrastructure. The security and compliance concerns are integral components. In general, the CVL comprehensively addresses factors such as service dependency, interoperability, pricing models, customization, security, compliance, and performance. By evaluating these critical elements, organizations can make informed decisions and proactively mitigate the risks associated with vendor lock-in in the dynamic landscape of cloud computing.

The figure shown in reference to Figure 3 demonstrates an important step in evaluating and handling dependencies on cloud service providers. During this stage, service consumers evaluate several cloud service providers to determine how much they depend on each provider. This evaluation involves inputting weighted factors into a dedicated Dependency Module, which then produces a comprehensive cost analysis and Dependency Degree for each service consumer.

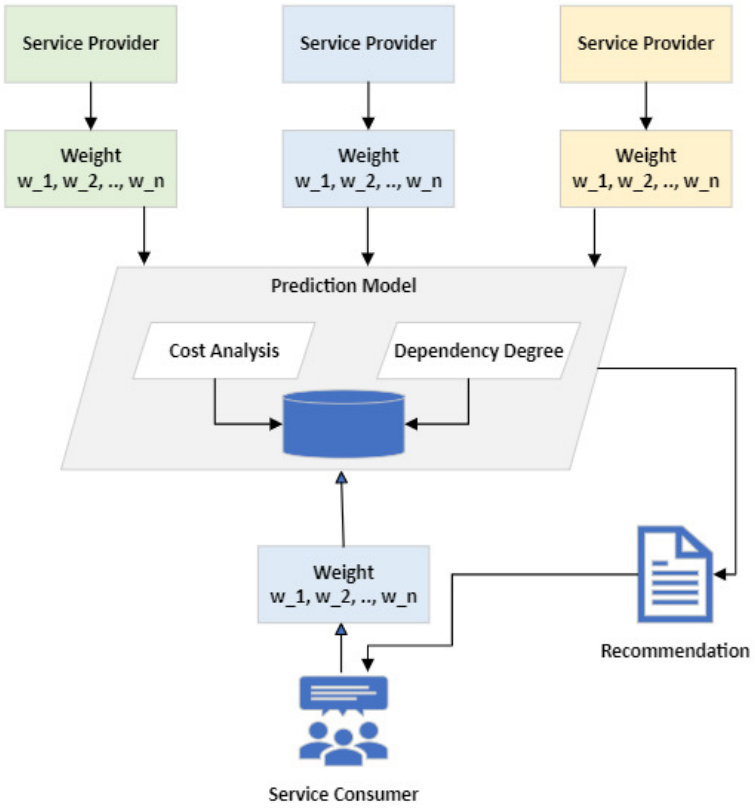

Figure 3. Cloud vendor lock-in framework.

The main components in the proposed framework (CVL) are as follows:

- Cloud service providers: Represented in the figure are multiple cloud service providers, each depicted with their respective logos or names. These providers offer various cloud services and solutions to service consumers.
- Service consumer input: Service consumers, which can be organizations or individuals, actively participate in the assessment process. They input specific weighted factors that are pertinent to their usage and reliance on each cloud service provider.
- Weighted factors: The weighted factors entered by service consumers reflect the critical aspects of their dependency on cloud service providers. These factors may include service costs, data transfer costs, service features, data security, compliance adherence, and more. Consumers assign relative weights to each factor based on their importance.
- Dependency Module: At the heart of the figure is the Dependency Module. This module receives the weighted factors input by service consumers and processes these data to perform a comprehensive analysis. It provides insights into how different services contribute to the overall expenses, allowing consumers to make informed decisions based on their budget constraints and cost optimization goals.
- Cost analysis: The module calculates the cost implications associated with each cloud service provider based on the weighted factors. It considers factors such as pricing models, data transfer costs, and service-specific charges to provide an accurate cost analysis.
- Dependency Degree: The Dependency Module computes the Dependency Degree for each service consumer–provider relationship. This metric quantifies the level of reliance or dependency of a service consumer on a particular cloud service provider. It takes into account the weighted factors and their impact on the overall dependency.

The Dependency Module of the CVL helps service consumers make informed decisions by evaluating critical factors such as service costs, data transfer expenses, security features, compliance adherence, scalability, and technical integrations. It assesses pricing structures, revealing both explicit and hidden costs, while also scrutinizing data transfer policies to

help users manage expenses effectively. The module examines security features, ensuring alignment with data protection needs and compliance standards. Additionally, it evaluates the scalability of services, which is crucial for organizations anticipating growth, and assesses technical integrations, including APIs and protocol support. By providing a comprehensive understanding of these dependencies, the CVL enables consumers to navigate and weigh these factors, facilitating optimal choices in selecting cloud service providers that align with their specific requirements and goals.

Cloud service providers will input their respective weight factors into a prediction model. The total weight added should be 100%. Afterward, users will specify how much weight they want to give to each factor. Then, the module will use the trained data to calculate the overall cost and predict the Dependency Degree between the service consumer and service providers. Finally, the recommended system will deliver a report to the user. The report will include a cost comparison and the Dependency Degree assessment for each of the service providers.

To reduce the risk of being locked into a particular vendor, the service consumer needs to follow the recommendations from the CVL. They should choose a service provider that best fits their specific requirements. This decision will also help them understand how dependent they are on the service provider, which in turn will help them predict the likelihood of vendor lock-in occurring in the future.

There are many benefits of using the CVL before selecting a service provider:

- By utilizing the dependency analysis, service consumers gain valuable insights into their reliance on multiple cloud service providers. This information enables them to make informed decisions regarding vendor lock-in, cost optimization, and risk management.
- The framework's cost analysis component assists service consumers in identifying potential cost-saving opportunities by comparing the cost implications of different cloud service providers.
- Service consumers can use the Dependency Degree to assess the level of risk associated with their cloud service provider relationships. This allows them to implement strategies to mitigate vendor lock-in risks and enhance operational flexibility.
- Armed with the insights from the framework, service consumers can develop tailored cloud strategies that align with their specific requirements, cost constraints, and desired levels of dependency on individual providers.

To put it simply, the cloud vendor lock-in (CVL) is a structure that helps in the examination and control of dependencies on several cloud service providers. It enables users to make informed decisions based on data, manage expenses, and efficiently maneuver the intricate terrain of cloud computing.

In the following sections, we define the main factors in the dependency calculation and cost analysis.

4.1. Dependency Degree

Measuring the Dependency Degree between a cloud provider and a consumer is essential for assessing the potential risks and understanding the level of reliance on the cloud provider, which will lead to vendor lock-in. In the CVL, we are considering many factors to calculate the dependency level as follows:

– Service-level agreements (SLAs): SLAs specify the level of service, availability, and performance guarantees. Analyze how these SLAs align with an organization's business needs. The tighter the SLAs, the higher the dependency.
– Data storage and ownership: Evaluate where the data are stored and who owns them. If the critical data resides primarily within the cloud provider's infrastructure, this implies a high degree of dependency.
– Resource scaling: Examine how easily an organization can scale its resources up or down within the cloud provider's environment. A lack of flexibility in resource scaling can indicate a higher degree of dependency.

- Service integration: Analyze the extent to which the organization's services and applications are tightly integrated with the cloud provider's services. The more tightly integrated, the greater the dependency.
- Exit strategy: Develop an exit strategy, which includes a plan for migrating away from the cloud provider if necessary. The complexity and cost of the exit strategy can give organizations an idea of their dependency level.
- Downtime and outages: Assess the historical downtime and outages experienced with the cloud provider's services. Frequent and prolonged outages can be a sign of dependency risk.
- Cost analysis: Evaluate the cost structure of the organization's cloud services. Understand how changes in usage and pricing can affect the organization's financial stability. Rapid cost increases can indicate a high level of dependency.
- Technical dependencies: Document the technical dependencies the organization's applications and services have on specific cloud services or APIs. Analyze how difficult it would be to replace or adapt these dependencies.
- Skills and training: Assess the skills and training of the organization's IT staff. If the team is heavily trained in a particular cloud provider's services, it can increase dependency.
- Monitoring and performance: Continuously monitor the performance and availability of the cloud resources. Sudden drops in performance or frequent downtime can indicate dependency issues.
- Third-party tools: Evaluate any third-party tools or services the organization uses to manage or enhance its cloud infrastructure. These tools may introduce additional dependencies.

4.2. Cost Analysis

Cost analysis data for vendor lock-in is highly specific to individual organizations, their cloud usage patterns, and the cloud providers they are currently using or considering. Here are some general cost analysis considerations related to vendor lock-in:

- Pricing models: Different cloud providers offer various pricing models, such as pay-as-you-go, reserved instances, or spot instances. Evaluate how these pricing models align with the organization's workload and budget. Consider the potential cost savings of switching to a different provider or adopting a multi-cloud strategy.
- Data egress costs: Some cloud providers charge fees for transferring data out of their ecosystems. Calculate the potential data egress costs if the organization needs to migrate its data to a different provider. This cost can be a significant factor in vendor lock-in scenarios.
- Service costs: Analyze the costs of the specific services and features the organization is using with its current provider. Compare these costs with equivalent services offered by other providers to assess potential cost savings.
- Licensing and support costs: Take into account any licensing or support agreements the organization has with its current provider. Consider how these costs may change if they switch to another provider or adopt a multi-cloud strategy.
- Exit costs: Calculate the potential costs associated with migrating the organization's applications, data, and workloads away from its current provider. This includes factors like re-configuration, re-testing, and potential downtime during the migration process.
- Resource scaling: Assess how well the current provider accommodates resource scaling and whether there are any associated costs. Evaluate whether a different provider or multi-cloud approach offers more cost-effective scalability options.
- Idle resource costs: Consider the costs of idle or underutilized resources. Some providers offer tools for optimizing resource allocation and cost management. Assess whether a different provider can help minimize idle resource costs.

- Cost optimization tools: Evaluate the cost optimization tools and features provided by the current provider and compare them with those offered by other providers. Look for tools that help identify cost-saving opportunities.
- Cost of compliance: Factor in the cost of maintaining compliance with regulatory requirements in different cloud environments. Some industries may have specific compliance obligations that can affect costs.
- Projected growth: Consider the organization's projected growth and how it may impact costs with the current provider. Assess whether alternative providers or multi-cloud strategies can accommodate future scalability needs more cost-effectively.

To conduct a thorough cost analysis for vendor lock-in, organizations must gather detailed financial data about their cloud usage, contracts, and anticipated changes. This analysis should be a part of a broader evaluation of cloud strategies, which should consider the financial repercussions of vendor lock-in in the short and long term, as well as potential strategies to minimize its impact.

5. Experiment and Results

There are numerous studies, outlined in the related work section [17,29,32,34,35], that utilize various equations for computing cost. This paper introduces a cost analysis module that will employ Equation (1):

$$Annual\ Costs = ((N \times C) \times (S + M)), \tag{1}$$

where:

N = Number of users;
C = A complexity multiplier from 1.00–2.00;
S = Security requirements rated on a 100–300 scale;
M = Monitoring requirements rated on a 100–300 scale.

To calculate the total cost of using a service provider, an equation is used. The equation takes into account several factors including the base cost, compute cost, storage cost, and data transfer cost. A service consumer must input the number of users and the complexity of the resources, which ranges from 1.00 to 2.00. The maximum complexity multiplier is two. Additionally, the security and monitoring provided by the service providers are rated on a scale from 100 to 300. The question about costs arises from a careful examination of previous studies that delved into cost calculations, pinpointing the key factors that can sway annual expenditures. These factors include the number of users (N), where a larger user base often translates to increased infrastructure demands, higher licensing costs, and potentially elevated expenses for security and monitoring. The complexity multiplier (C) acknowledges the diverse nature of systems, recognizing that not all are equally intricate; it provides a means to adjust costs based on a system's complexity arising from configurations, integrations, or customization. Security requirements (S) take center stage as a critical consideration in any IT system accommodating varied security levels and factoring in associated costs for advanced security measures. Likewise, monitoring requirements (M) are essential for system health and performance, with the scale providing flexibility to address diverse monitoring needs. Systems with heightened complexity or stringent performance demands may incur higher monitoring expenses.

We have analyzed three cloud service providers, namely Provider A, Provider B, and Provider C, based on our dataset. For each provider, we calculated the cost of using their services at various scales, taking into account the number of users, complexity, and security requirements. In Figure 4, we compared the cost of the three providers with the same number of users but different complexity scales. Our analysis shows that Provider A has the least complexity, followed by Provider B, and Provider C with the highest complexity. In terms of monitoring requirements, Provider C has the highest requirements, while Providers A and B have the same scale. Furthermore, we compared the cost of the three providers with the same number of users but different security scales as shown in Figure 5.

Our findings indicate that Provider A has the lowest security scale, followed by Provider C, and then Provider B with the highest security scale.

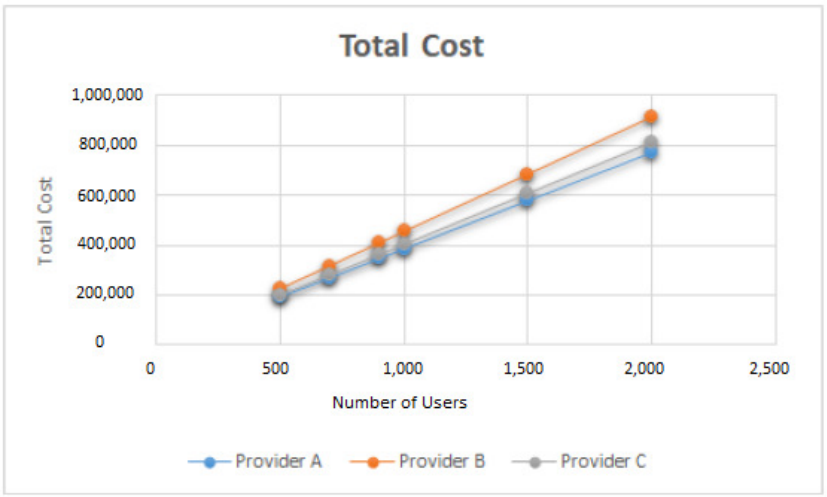

Figure 4. Total cost for three providers with different complexity scales.

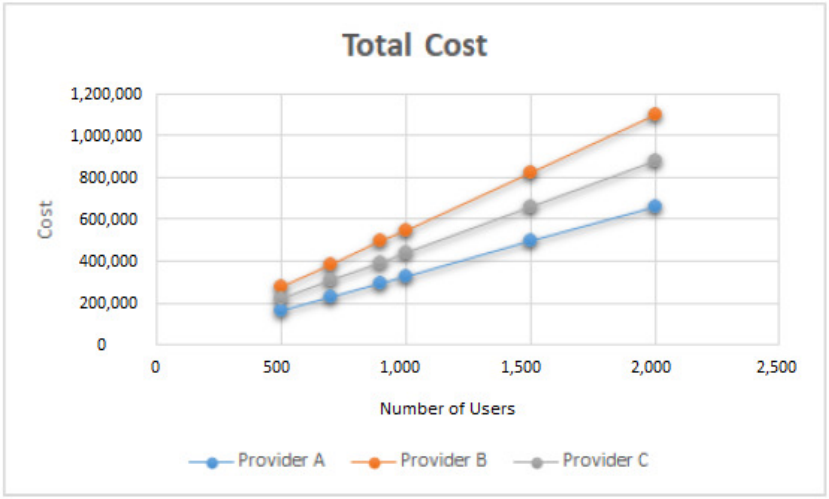

Figure 5. Total cost for three providers with different security scales.

In the second step of the CVL, the degree of dependency is calculated. The service consumer assigns weight to technical (TL), data (DL), service (SL), certification (CL), contract (OL), economic (EL), and network (NL) factors. The Dependency Score is then calculated using Equation (2):

$$Dependency\ Score = \sum_{i=1}^{n} (w_i \times f_i), \tag{2}$$

where:

n is the number of factors considered.
w_i is the weight assigned to factor i (a value between 0 and 1, representing its importance).
f_i is the normalized score for factor i (a value between 0 and 1, representing the level of dependency for that factor).

We collected data from three providers and fifteen consumers. Figure 6 shows their dependency levels. Figure 7 demonstrates how varying weights impact the dependency levels.

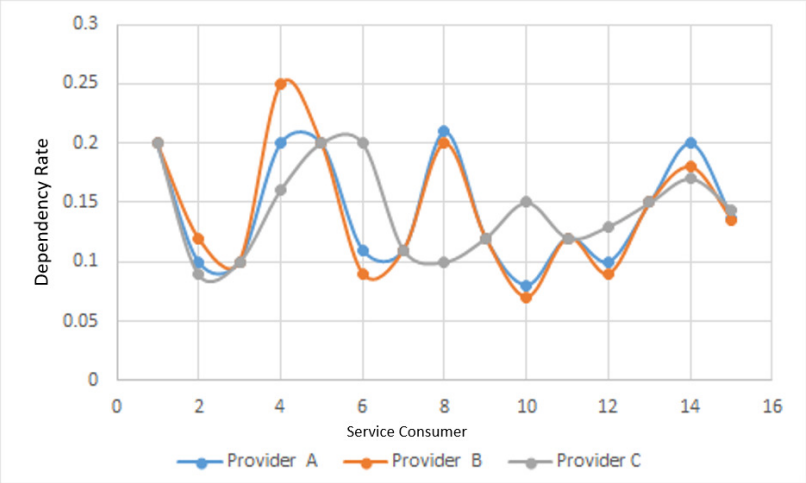

Figure 6. Dependency Degree for three different service providers and fifteen service consumers.

Figure 7. Dependency level for three different service providers and one service consumer.

The cloud vendor lock-in framework (CVL) involves generating a comprehensive report that assesses and ranks different cloud service providers. This approach aims to give service consumers valuable insights, helping them make informed decisions and mitigate the challenges associated with vendor lock-in. The CVL provides clear insights into how much an organization relies on specific cloud services, including the financial impact of these dependencies. By analyzing factors like service cost and level of dependency, the CVL helps decision-makers understand potential risks and hidden costs. This information enables them to make smart choices, avoiding excessive reliance on a single cloud vendor and ensuring alignment with organizational goals and budgets. In summary, the CVL simplifies the complex landscape of cloud vendor dependencies, allowing for more strategic decision-making.

The research conducted under the CVL has significant implications for both managerial decision-making and academic discourse. By comparing the results of various studies with the literature review, potential alignments or divergences can be observed based on the specific focus and methodologies used in each study. From a managerial perspective, the findings of the CVL offer valuable guidance to decision-makers who are facing challenges related to cloud vendor lock-in. The research provides insights into the levels of dependency, financial implications, and comparative analyses of service providers, which can help organizations optimize their cloud infrastructure. This information empowers managers to

mitigate risks, make informed vendor selections, and strategically plan cloud adoption in line with organizational objectives.

The CVL's research contributes to the existing literature by presenting a systematic framework for evaluating dependencies and financial implications in the context of cloud vendor lock-in. This research enriches academic discussions surrounding cloud computing, vendor management, and strategic decision-making. It not only advances our understanding of these dynamics but also provides a practical foundation for future studies to build upon.

It is important to recognize certain limitations in the proposed methodology. One notable limitation is the dependence on historical data, which may not fully reflect the rapidly changing landscape of cloud services. Moreover, the sensitivity of the methodology to variations in organizational structures, sizes, and industry regulations should be acknowledged. Along with these challenges, a significant consideration involves determining the weights assigned to factors from the perspective of the service consumer. Selecting these weights and deciding their application in the equation represent key challenges in the methodology. These limitations highlight the need for cautious interpretation and application of the research findings in practical settings.

6. Conclusions

The cloud vendor lock-in prediction framework (CVL) is a solution that addresses the issue of vendor lock-in in cloud computing. This paper presents an innovative approach that evaluates the relationship between cloud service providers and consumers, enabling predictive risk analysis and detailed cost assessments for informed decision-making. The Dependency Module, which is at the core of the CVL, integrates consumer input with factors such as service cost, security, compliance, scalability, and technical integrations, producing comprehensive reports. These reports empower users with the insights needed to make educated decisions, mitigate vendor lock-in risks, optimize costs, and tailor cloud strategies to organizational needs. This paper provides in-depth insights into varying Dependency Degrees and associated monetary implications, with a focus on key factors in dependency calculation and cost analysis. The experimental results demonstrate the effectiveness of the CVL, showcasing diverse dependency levels between service providers and customers and highlighting its data-driven approach to strategic decision-making.

In essence, the CVL proves to be an invaluable tool, empowering users to proactively address vendor lock-in challenges and make informed decisions for strategically tailored cloud strategies in the dynamic cloud computing landscape. In future developments, the intention is to extend the application of the method to large-scale datasets and explore additional factors that can contribute to the calculation of the Dependency Degree. This expansion aims to enhance the robustness and applicability of the approach, allowing for a more comprehensive and nuanced evaluation of the relationship between service providers and consumers in the dynamic landscape of cloud computing.

Author Contributions: Methodology, A.A.; Validation, A.A.; Formal analysis, S.P.; Investigation, A.A., S.P. and K.K.; Resources, S.P. and K.K.; Writing—original draft, A.A.; Writing—review & editing, A.A.; Supervision, A.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Almorsy, M.; Grundy, J.; Müller, I. An analysis of the cloud computing security problem. *arXiv* **2016**, arXiv:1609.01107.
2. Vinoth, S.; Vemula, H.L.; Haralayya, B.; Mamgain, P.; Hasan, M.F.; Naved, M. Application of cloud computing in banking and e-commerce and related security threats. *Mater. Today Proc.* **2022**, *51*, 2172–2175. [CrossRef]
3. Subashini, S.; Kavitha, V. A survey on security issues in service delivery models of cloud computing. *J. Netw. Comput. Appl.* **2011**, *34*, 1–11. [CrossRef]

4. Pearson, S. Taking account of privacy when designing cloud computing services. *Int. J. Inf. Manag.* **2011**, *31*, 353–357.
5. Kang, Y. Development of Large-Scale Farming Based on Explainable Machine Learning for a Sustainable Rural Economy: The Case of Cyber Risk Analysis to Prevent Costly Data Breaches. *Appl. Artif. Intell.* **2023**, *37*, 2223862. [CrossRef]
6. Fitzpatrick, B.W.; Lueck, J. The Case Against Data Lock-in: Want to keep your users? Just make it easy for them to leave. *Queue* **2010**, *8*, 20–26. [CrossRef]
7. Govindarajan, A.; Lakshmanan. Overview of cloud standards. *Cloud Comput. Princ. Syst. Appl.* **2010**, 77–89.
8. Armbrust, M.; Fox, A.; Griffith, R.; Joseph, A.D.; Katz, R.H.; Konwinski, A.; Lee, G.; Patterson, D.A.; Rabkin, A.; Stoica, I.; et al. A view of cloud computing. *Commun. ACM* **2010**, *53*, 50–58. [CrossRef]
9. Opara-Martins, J.; Sahandi, R.; Tian, F. Critical review of vendor lock-in and its impact on adoption of cloud computing. In Proceedings of the International Conference on Information Society (i-Society 2014), London, UK, 10–12 November 2014; pp. 92–97.
10. Stravoskoufos, K.; Preventis, A.; Sotiriadis, S.; Petrakis, E.G. A Survey on Approaches for Interoperability and Portability of Cloud Computing Services. In Proceedings of the CLOSER 2014—4th International Conference on Cloud Computing and Services Science, Barcelona, Spain, 3–5 April 2014; pp. 112–117.
11. Bhavya, K.; Yamini, K.; Sreenivas, V. Cloud Services Portability for secure migration. *Int. J. Comput. Trends Technol.* **2013**, *4*.
12. Razavian, S.M.; Khani, H.; Yazdani, N.; Ghassemi, F. An analysis of vendor lock-in problem in cloud storage. In Proceedings of the ICCKE 2013, Mashhad, Iran, 16 December 2013; pp. 331–335.
13. Giacomini, M.; Ullah, A. *YASF: A Vendor-Agnostic Framework for Serverless Computing*; Scitepress Digital Library: Setúbal, Portugal, 2023.
14. Quint, P.C.; Kratzke, N. Overcome vendor lock-in by integrating already available container technologies towards transferability in cloud computing for smes. *Cloud Comput.* **2016**, *50*.
15. Cattaneo, G. The demand of Cloud Computing in Europe: Drivers, barriers, market estimates. In Proceedings of the IDC, Research in Future Cloud Computing Workshop, Bruxelles, Belgium, 2 May 2012.
16. Bhardwaj, S.; Jain, L.; Jain, S. Cloud computing: A study of infrastructure as a service (IAAS). *Int. J. Eng. Inf. Technol.* **2010**, *2*, 60–63.
17. Berriman, G.B.; Juve, G.; Deelman, E.; Regelson, M.; Plavchan, P. The application of cloud computing to astronomy: A study of cost and performance. In Proceedings of the 2010 Sixth IEEE International Conference on e-Science Workshops, Brisbane, Australia, 7–10 December 2010; pp. 1–7.
18. Armbrust, M.; Fox, A.; Griffith, R.; Joseph, A.D.; Katz, R.H.; Konwinski, A.; Lee, G.; Patterson, D.A.; Rabkin, A.; Stoica, I.; et al. *Above the Clouds: A Berkeley View of Cloud Computing*; Technical Report, Technical Report UCB/EECS-2009-28; EECS Department, University of California: Berkeley, CA, USA, 2009.
19. Karidis, J.; Moreira, J.E.; Moreno, J. True value: Assessing and optimizing the cost of computing at the data center level. In Proceedings of the 6th ACM Conference on Computing Frontiers, Ischia, Italy, 18–20 May 2009; pp. 185–192.
20. Abadi, D.J. Data management in the cloud: Limitations and opportunities. *IEEE Data Eng. Bull.* **2009**, *32*, 3–12.
21. Buyya, R.; Yeo, C.S.; Venugopal, S.; Broberg, J.; Brandic, I. Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility. *Future Gener. Comput. Syst.* **2009**, *25*, 599–616. [CrossRef]
22. Khalid, A. Cloud computing: Applying issues in small business. In Proceedings of the 2010 International Conference on Signal Acquisition and Processing, Bangalore, India, 9–10 February 2010; pp. 278–281.
23. Gonzalez, N.; Miers, C.; Redigolo, F.; Simplicio, M.; Carvalho, T.; Näslund, M.; Pourzandi, M. A quantitative analysis of current security concerns and solutions for cloud computing. *J. Cloud Comput. Adv. Syst. Appl.* **2012**, *1*, 11. [CrossRef]
24. Li, A.; Yang, X.; Kandula, S.; Zhang, M. CloudCmp: Comparing public cloud providers. In Proceedings of the 10th ACM SIGCOMM Conference on Internet Measurement, Melbourne, Australia, 1–30 November 2010; pp. 1–14.
25. Cloud, H. The nist definition of cloud computing. *Natl. Inst. Sci. Technol. Spec. Publ.* **2011**, *800*, 145.
26. Badger, M.L.; Grance, T.; Patt-Corner, R.; Voas, J.M. *Cloud Computing Synopsis and Recommendations*; National Institute of Standards & Technology: Gaithersburg, MA, USA, 2012.
27. Satapathy, S.C.; Bhateja, V.; Das, S. Smart intelligent computing and applications. In Proceedings of the Second International Conference on SCI, Hohhot, China, 22–24 October 2018; Springer: Berlin/Heidelberg, Germany, 2018; Volume 1.
28. Ahronovitz, M. Cloud Computing Use Cases: Introducing Service Level Agreements. Use Cases Discussion Group, White Paper V4. 0. 2015.
29. Martens, B.; Walterbusch, M.; Teuteberg, F. Costing of cloud computing services: A total cost of ownership approach. In Proceedings of the 2012 45th Hawaii International Conference on System Sciences, Maui, HI, USA, 4–7 January 2012; pp. 1563–1572.
30. Säfsten, K.; Gustavsson, M. *Research Methodology: For Engineers and Other Problem-Solvers*; Studentlitteratur AB: Lund, Sweden, 2020.
31. Chaudry, R.; Guabtni, A.; Fekete, A.; Bass, L.; Liu, A. Consumer Monitoring of Infrastructure Performance in a Public Cloud. In Proceedings of the Web Information Systems Engineering—WISE 2014: 15th International Conference, Thessaloniki, Greece, 12–14 October 2014; Part II 15; Springer: Berlin/Heidelberg, Germany, 2014; pp. 425–434.
32. Mazrekaj, A.; Shabani, I.; Sejdiu, B. Pricing schemes in cloud computing: An overview. *Int. J. Adv. Comput. Sci. Appl.* **2016**, *7*, 80–86. [CrossRef]

33. Andrikopoulos, V.; Binz, T.; Leymann, F.; Strauch, S. How to adapt applications for the Cloud environment: Challenges and solutions in migrating applications to the Cloud. *Computing* **2013**, *95*, 493–535. [CrossRef]
34. Al-Roomi, M.; Al-Ebrahim, S.; Buqrais, S.; Ahmad, I. Cloud computing pricing models: A survey. *Int. J. Grid Distrib. Comput.* **2013**, *6*, 93–106. [CrossRef]
35. Ji, C.; Li, Y.; Qiu, W.; Awada, U.; Li, K. Big data processing in cloud computing environments. In Proceedings of the 2012 12th International Symposium on Pervasive Systems, Algorithms and Networks, San Marcos, TX, USA, 13–15 December 2012; pp. 17–23.