Document name	Code	Segment	Created by
Ahmad2020-Cloud_Computing_Trends_and_Cloud_Migration_Tuple	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	And looking at intra-cloud, these ser-vices are principally managed and administered through service-oriented architecture (SOA).	Ivon Miranda Santos
Ahmad2020-Cloud_Computing_Trends_and_Cloud_Migration_Tuple	CLOUD-NATIVE ARCHITECTURE > Scalability	Cloud computing also provides a new approach to application development and deployment known as cloud native. Cloud native approach among others ensures the stateless computing so that elasticity of cloud can be achieved in real time and hence user traffic can be dynamically directed to any server regardless of their state of sessions. Cloud-native application (CNA) characterizes a distributed, elastic, and horizontal scalable system consisted of (micro) services that segregates state in a minimum of stateful components [14]. CNA may be developed with combination of best	Ivon Miranda Santos
Ahmad2020-Cloud_Computing_Trends_and_Cloud_Migration_Tuple	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Cloud-native application (CNA) characterizes a distributed, elastic, and horizontal scalable system consisted of (micro) services that segregates state in a minimum of stateful components [14].	Ivon Miranda Santos
Ahmad2020-Cloud_Computing_Trends_and_Cloud_Migration_Tuple	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Dynamic: dynamic term is associated with real-time changes for instance cloud-native applications support dynamic resource allocation	Ivon Miranda Santos
Ahmad2020-Cloud_Computing_Trends_and_Cloud_Migration_Tuple	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Cloud computing technology is maturing day by day with hypervisors, containers, cloud-native applications, flog computing, edge computing, and cloudlets to become a robust platform to host complete range of information technology	Ivon Miranda Santos
Alouffi2021-A_systematic_literature_review_on_cloud_computing_s	CLOUD-NATIVE ARCHITECTURE > Scalability	The emerging paradigm of cloud computing ensures the reliability, availability, and scalability while users access the cloud computing services	Ivon Miranda Santos
Alouffi2021-A_systematic_literature_review_on_cloud_computing_s	CLOUD-NATIVE ARCHITECTURE > Scalability	Minimized latency is needed when communication and computational capabilities are transferred close to the sensor nodes. To ease the network scalability, mobility, and location awareness, minimized latency is required for the devices connected in a physical environment. Therefore, low latency is favorable to cloud computing users.	Ivon Miranda Santos
Asthana2021-Multi-cloud_Solution_Design_for_Migrating_a_Portfol	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	. Large client may have significant percentage of applications in this category. So, one has to evaluate cloud feasibility and understand whether there is a need to re-architect application based on what services providers are able to offer. Second, clients usually define multiple features, encryption/security level, and other service level requirements they expect in the providers they will migrate each of their applications to. Thus, choosing the right providers for different	Ivon Miranda Santos
Asthana2021-Multi-cloud_Solution_Design_for_Migrating_a_Portfol	CLOUD-NATIVE ARCHITECTURE > Flexibility	Moving an application to the cloud enables organization to make use of the advantages of the cloud like elasticity [2], lower costs, and accessibility of data. To further utilize these advantages, achieve maximum flexibility, and avoid "concentration risk" (putting too many application eggs in one cloud basket), organizations recently tend to spread their	Ivon Miranda Santos
Asthana2021-Multi-cloud_Solution_Design_for_Migrating_a_Portfol	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	However, there are many challenges in achieving this. First, there are usually a lot of legacy applications that lack proper documentation. This makes it hard to even assess whether it is feasible to migrate such applications to the cloud	Ivon Miranda Santos
Asthana2021-Multi-cloud_Solution_Design_for_Migrating_a_Portfol	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	Also, sometimes these applications require an overall architecture overhaul, which again, suffer from the lack of documentation of the current	Ivon Miranda Santos
Asthana2021-Multi-cloud_Solution_Design_for_Migrating_a_Portfol	CLOUD-NATIVE ARCHITECTURE > Architecture refactoring	Also, sometimes these applications require an overall architecture overhaul, which again, suffer from the lack of documentation of the current	Ivon Miranda Santos
Asthana2021-Multi-cloud_Solution_Design_for_Migrating_a_Portfol	CLOUD-NATIVE ARCHITECTURE > Scalability	Perform Text Mining on Data for Each Application. In this step, we first collect meta-data for each application. For each client application that the client wants to migrate to cloud, we screen applications and collect application metadata features like number of users, geography of users, investment on application, security and compliance agree-ment, geography of database where data is stored, software platforms, authentication server, server, hardware, network setup, reliability, scalability, etc. Also, we collect client request for features that should be in the	Ivon Miranda Santos
Asthana2021-Multi-cloud_Solution_Design_for_Migrating_a_Portfol	CLOUD-NATIVE ARCHITECTURE > Flexibility	capacity • Applications that have architectural flexibility • Applications that require global scale e.g.,	Ivon Miranda Santos

Asthana2021-Multi-cloud_Solution_Design_for_Migrating	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	• Based on applications with similar architectures including microservices, cloud native, mobile etc.	Ivon Miranda Santos
Asthana2021-Multi-cloud_Solution_Design_for_Migrating	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	• Based on applications with similar architectures including microservices, cloud native, mobile etc.	Ivon Miranda Santos
Asthana2021-Multi-cloud_Solution_Design_for_Migrating	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	• Based on applications with similar architectures including microservices, cloud native, mobile etc.	Ivon Miranda Santos
Asthana2021-Multi-cloud_Solution_Design_for_Migrating_a_Portfol	CLOUD-NATIVE ARCHITECTURE > Flexibility	– limited scalable storage up to 2 TBPS • Does it have architectural flexibility? – Not much, it requires more storage capacity and more in-stances to make it scalable and reliable.	Ivon Miranda Santos
Aydin2021-A_Study_of_Cloud_Computing_Adoption_in_Universities_a	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Cloud architecture status. Nearly half of the universities (43.5%) stated that they did not have a cloud architecture (Figure 5). Results showed that 21.2% of the participants had a private cloud architecture, 12.9% had a hybrid cloud archi-tecture, 11.8% had a community cloud architecture, and 10.6% had a public cloud architecture. In addition, the data obtained from	Ivon Miranda Santos
Aydin2021-A_Study_of_Cloud_Computing_Adoption_in_Universities_a	CLOUD-NATIVE ARCHITECTURE > Flexibility	Participants were asked to identify the benefits of cloud computing for universities (Figure 9). As the first ben-efit of cloud computing, 76.5% of the participants stated that "service continuity will be ensured in natural disasters and unexpected developments." This is followed by "service flexibility" (70.6%), "service variety" (69.4%), "paying as much as you use in information services" (65.9%), "decrease in ICT operation and maintenance expenditures" (65.1%), "execution of information services with simpler ICT tools" (62.4%), "less damage to the environment"	Ivon Miranda Santos
Aydin2021-A_Study_of_Cloud_Computing_Adoption_in_Universities_a	CLOUD-NATIVE ARCHITECTURE > Flexibility	Different meth-odologies and results have been used in the adoption of cloud computing. Some of the critical success factors (CSFs) for the effective implementation of cloud-based e-learning at universities and institutes of higher education are Cloud Data Security, Availability and Reliability, Customizable Service Level Agreement, Network Bandwidth, Compatibility, Technical Support, Management Support, Human and Resource Readiness, Complexity, Cost Flexibility, Ease of Use, and Relative Advantage (Naveed & Ahmad, 2019). Hybrid clouds contain a combination of two or	Ivon Miranda Santos
Aydin2021-A_Study_of_Cloud_Computing_Adoption_in_Universities_a	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	A proposed architec-ture based on the hybrid cloud model which uses both the pub-lic and private clouds is simulated using CloudSim.	Ivon Miranda Santos
Aydin2021-A_Study_of_Cloud_Computing_Adoption_in_Universities_a	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	In the context of current technologies such as Web 2.0, universities may develop cloud services within the service-oriented architecture (SOA) structure in accordance with the needs of their	Ivon Miranda Santos
Aydin2021-A_Study_of_Cloud_Computing_Adopt	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	The architecture of the proposed hybrid cloud includes SaaS, IaaS, and PaaS platform as	Ivon Miranda Santos
Aydin2021-A_Study_of_Cloud_Computing_Adoption_in_Universities_a	CLOUD-NATIVE ARCHITECTURE > Flexibility	In this architecture, data will be synchronized across PUC and CUC infrastructure. Universities can gain the flexibility and computing power of the CUC cloud for their basic and non-sensitive computing tasks, while they can keep their criti-cal applications and data safely behind a university firewall via CUC. Each of the PUC and CUC environments that make up this hybrid cloud	Ivon Miranda Santos
Aydin2021-A_Study_of_Cloud_Computing_Adoption_in_Universities_a	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Each of the PUC and CUC environments that make up this hybrid cloud architecture has its own benefits and uses.	Ivon Miranda Santos
Aydin2021-A_Study_of_Cloud_Computing_Adopt	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	These results support the fact that nearly half of the IT departments have a cloud architecture.	Ivon Miranda Santos
Baby2015-Multicloud_architecture_for_augmenting_security_in_clo	CLOUD-NATIVE ARCHITECTURE > Flexibility	It can be used to increase robustness of the system in the design of dependable systems [4]. In large business processes for managing supply chains, where single source suppliers are not considered to lower dependency on suppliers, thereby increasing the flexibility of the business process [5]. B. Partition of Application This architecture protects against undesired data	Ivon Miranda Santos
Bainomugisha2022-Crane_cloud_A_resilient_multicloud_service_abs	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Originally conceived as a monolithic system, the APDD is transitioning to a microservice architecture to optimize efficiency and work around the technology constraints (Fig.	Ivon Miranda Santos

Bainomugisha2022-Crane_cloud_A_resilient_multicloud_service_abs	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Many countries are coming up with new laws and guidelines to reg-ulate the location of data and access. Countries have recently enacted laws that enforce data sovereignty and prevent data from leaving the country's boundaries. This becomes difficult to implement in regions where public clouds do not have physical presence. For instance, on the African continent where public cloud data centers are still sparse, it becomes almost impossible to comply with such policies. The microservice architecture and cloud orchestration platforms such as Kubernetes (Burns et al., 2016) promises potential remedies to this challenge. In the above scenario, the APDD	Ivon Miranda Santos
Bainomugisha2022-Crane_cloud_A_resilient_multicloud_service_abs	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	The microservice architecture and cloud orchestration platforms such as Kubernetes (Burns et al., 2016) promises potential remedies to this challenge. In the above scenario, the APDD system can be broken into independent microservices each with different data jurisdiction	Ivon Miranda Santos
Bainomugisha2022-Crane_cloud_A_resilient_multicloud_service_abs	CLOUD-NATIVE ARCHITECTURE > Flexibility	, 2013). These user categories may not have the local computing resources to run their workloads and most often resort to cloud providers but flexibility to switch/shift between providers is one of their desirable properties. Other than vendor lock-in, there are other variations including product lock-in, version lock-in, architecture lock-in, platform lock-in, skills lock-in and mental lock-in	Ivon Miranda Santos
Bainomugisha2022-Crane_cloud_A_resilient_multicloud_service_abs	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Many organizations are recently adopting microservice architec-tures in place of traditional monolithic architectures so as to truly reap from the benefits of modern cloud services.	Ivon Miranda Santos
Bainomugisha2022-Crane_cloud_A_resilient_multicloud_service_abs	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Microservice architec-tures involve collaborations between different fine-grained and inde-pendently deployable modules usually without a centralized controller to achieve the desired overall functionality of the system (Nadareishvili et al., 2016; Knoche and Hasselbring, 2019).	Ivon Miranda Santos
Bainomugisha2022-Crane_cloud_A_resilient_multicloud_service_abs	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Driven by application features such as scalability, agility, performance and fault-tolerance, microservice architectures involve autonomous software development teams independently working to build loosely coupled application fea-tures and employing collaborative workflows and automation tools from version control systems to full scale production deployment (Has-selbring and	Ivon Miranda Santos
Bainomugisha2022-Crane_cloud_A_resilient_multicloud_service_abs	CLOUD-NATIVE ARCHITECTURE > Scalability	Driven by application features such as scalability, agility, performance and fault-tolerance, microservice architectures involve autonomous software development teams independently working to build loosely coupled application fea-tures and employing collaborative workflows and automation tools from version control systems to full scale production deployment (Has-selbring and	Ivon Miranda Santos
Bainomugisha2022-Crane_cloud_A_resilient_multicloud_service_abs	CLOUD-NATIVE ARCHITECTURE > Coupling	Driven by application features such as scalability, agility, performance and fault-tolerance, microservice architectures involve autonomous software development teams independently working to build loosely coupled application fea-tures and employing collaborative workflows and automation tools from version control systems to full scale production deployment (Has-selbring and	Ivon Miranda Santos
Bainomugisha2022-Crane_cloud_A_resilient_multicloud_service_abs	CLOUD-NATIVE ARCHITECTURE > Scalability	A number of popular technology com-panies such as Uber, Spotify, Netflix, Amazon and Ebay are now using microservices at the core of their business processes and have achieved differing levels of reliability and scalability in their services (Knoche and Hasselbring, 2019). As part of the inceptor team of the microservice terminology, Fowler and Lewis (2014) identified the following	Ivon Miranda Santos
Bainomugisha2022-Crane_cloud_A_resilient_multicloud_service_abs	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Microservice archi-tectures thrive on the notion of 'small size' with reductions in the scope of the problem, task completion time, feedback response time and the size of the deployment unit.	Ivon Miranda Santos
Bainomugisha2022-Crane_cloud_A_resilient_multicloud_	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	(2016) noted that the microservice architecture is an enabler for use of DevOps.	Ivon Miranda Santos
Bainomugisha2022-Crane_cloud_A_resilient_multicloud_service_abs	CLOUD-NATIVE ARCHITECTURE > Coupling	The lightweight nature of containerized applications coupled with func-tional and configuration encapsulation facilitates replication and porta-bility, are cost-efficient and have a reduced overhead on the operation and	Ivon Miranda Santos

Bainomugisha2022-Crane_cloud_A_resilient_multicloud_service_abs	CLOUD-NATIVE ARCHITECTURE > Scalability	Fully leveraging the elasticity of using container-based virtualiza-tion requires automated orchestration and cluster management tools such as Kubernetes,11 Nomad,12 Docker Swarm13 and DC/OS14 that provide an abstraction layer between computing resource pools and the appli-cations. These tools provide a number of attributes important to imple-ment service discovery, scalability, load balancing, service replication and provisioning of replicas across multiple compute	Ivon Miranda Santos
Bainomugisha2022-Crane_cloud_A_resilient_multicloud_service_abs	CLOUD-NATIVE ARCHITECTURE > Scalability	Despite its complexity such as in installation, Kubernetes is the most widely adopted and powerful container orchestration tool owing to its immense scalability, performance and advanced automation features. It has inbuilt mon-itoring and logging libraries and processes which are lacking	Ivon Miranda Santos
Bainomugisha2022-Crane_cloud_A_resilient_multicloud_service_abs	CLOUD-NATIVE ARCHITECTURE > Scalability	Application replication ensures that there is operational business continuity in the face of downtime as a result of comput-ing equipment failures, natural disasters, planned maintenance opera-tions (Levijarvi and Mitzev, 2015), power outages, unreliable network connectivity, limited bandwidth and utilization surges. Replication can further be used to realize scalability, availability and fault-tolerance of an application under scheduled or unplanned downtime periods. The replica count, the number of clones of an application, depends on the reliability assurance as a requirement for an application and also the popularity of the service in a cluster(node) region. Replication strategies fall into two broad categories: 1. Static replication strategy where the number of nodes and repli-cas is defined beforehand 2. Dynamic replication strategy where replicas are	Ivon Miranda Santos
Bainomugisha2022-Crane_cloud_A_resilient_multicloud_service_abs	CLOUD-NATIVE ARCHITECTURE > Scalability	As noted earlier, cloud computing offers significant benefits such as scalability, disaster recovery, mobility and cost reduction in operation of an organization's IT infrastructure. This is evidenced in the introduction of different cloud computing technologies and deployments to make it easy for organizations to embrace and adopt this new wave of handling com-pute, storage and	Ivon Miranda Santos
Bainomugisha2022-Crane_cloud_A_resilient_multicloud_service_abs	CLOUD-NATIVE ARCHITECTURE > Scalability	4.2.1. Container orchestration Container orchestration tools are used to automate the deployment, management, scaling, and networking of containerized applications. These tools provide an abstraction layer between pools of resources and the application containers that run on those resources. Kubernetes, HashiCorp Nomad (Sabharwal et al., 2021), Docker swarm and Apache Mesos are the most popular tools with the former taking a fair share of the cloud-native market. With an impressively large community and functionality, backed by Cloud Native Computing Foundation (CNCF) and its open source nature, Crane Cloud uses Kubernetes for container orchestration. We considered five factors in selecting the most viable tool for our setup: Community (Support), Open Source, Scalability & Flexibility, Fault tolerance and Monitoring support. As shown in the comparison Table 2, Kubernetes is an open source project that impressively commands the cloud-native market with an adoption rate of 50% in the past 6 months and 87% market penetration support-ing application scalability, fault tolerance and has inbuilt monitoring and logging tools and hence was the preferred choice for container	Ivon Miranda Santos
Bainomugisha2022-Crane_cloud_A_resilient_multicloud_service_abs	CLOUD-NATIVE ARCHITECTURE > Flexibility	We considered five factors in selecting the most viable tool for our setup: Community (Support), Open Source, Scalability & Flexibility, Fault tolerance and Monitoring support. As shown in the comparison Table 2, Kubernetes is an open source project that impressively commands the cloud-native market with an adoption rate of 50% in the past 6 months and 87% market penetration support-ing application scalability, fault tolerance and has inbuilt monitoring and logging tools and hence was the preferred choice for container	Ivon Miranda Santos

Bainomugisha2022-Crane_cloud_A_resilient_multicloud_service_abs	CLOUD-NATIVE ARCHITECTURE > Scalability	This enables organizations to work with data-driven and legacy applications while leveraging the portability, scalability and highly available features of containers. Traditionally, Kubernetes used to provide support for manual attachment of cloud-backed storage to applications limiting usage outside the cloud provider but cloud native	Ivon Miranda Santos
Bainomugisha2022-Crane_cloud_A_resilient_multicloud_service_abs	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Crane Cloud uses OpenEBS,36 an open source Container Attached Storage (CAS) solution developed using the microservice architecture.	Ivon Miranda Santos
Bainomugisha2022-Crane_cloud_A_resilient_multicloud_service_abs	CLOUD-NATIVE ARCHITECTURE > Scalability	IaaS (virtual machines) hybrid clouds using custom heuristics. Proper-ties such as high availability, scalability, fault-tolerance and monitoring are not discussed and the use of virtual machines may not be the most cost-effective approach to running application workloads. Filip et al. (2018) proposed a solution that considers a finite catalog of primitive microservices and designs a hybrid scheduling algorithm that matches tasks to resources based on task history and availability of resources. In addition to benefits of using a microservice architecture, the paper asserted that costs can further be reduced by placing data	Ivon Miranda Santos
Bainomugisha2022-Crane_cloud_A_resilient_multicloud_service_abs	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	In addition to benefits of using a microservice architecture, the paper asserted that costs can further be reduced by placing data closer to processing points based on user density.	Ivon Miranda Santos
Balalaie2016-Microservices_architecture_enables_devops_Migratio	CLOUD-NATIVE ARCHITECTURE > Flexibility	The microservices architecture is one of the first service-based architectural styles that has been introduced, applied in practice, and become popular when the DevOps practices gained momentum in the software industry. Migrating monolithic architectures to cloud-native architectures like microservices brings in many benefits such as flexibility to adapt to the technological changes and independent resource management for different system components. Here, we report our experiences and lessons learned during incremental migration and	Ivon Miranda Santos
Balalaie2016-Microservices_architecture_enables_devops_Migratio	CLOUD-NATIVE ARCHITECTURE > Architecture refactoring	Migrating monolithic architectures to cloud-native architectures like microservices brings in many benefits such as flexibility to adapt to the technological changes and independent resource management for different system components. Here, we report our experiences and lessons learned during incremental migration and architectural refactoring of a commercial Mobile Backend as a Service to microservices. We provide a detailed explanation of how we adopted DevOps and its practices, and how these	Ivon Miranda Santos
Balalaie2016-Microservices_architecture_enables_devops_Migratio	CLOUD-NATIVE ARCHITECTURE > Flexibility	In this setting, each service is a business capability that can utilize various programming languages and data stores and is developed by a small team [2]. Migrating monolithic architectures to microservices brings in many benefits including, but not limited to flexibility to adapt to the technological changes in order to avoid technology lock-in, and more importantly, reduced time-to-market, and better development team structuring around services [3]. Here we explain our experiences and lessons	Ivon Miranda Santos
Balalaie2016-Microservices_architecture_enables_devops_Migratio	CLOUD-NATIVE ARCHITECTURE > Scalability	//www.backtory.com/) platform, a commercial Mobile Backend as a Service (MBaaS), to microservices in the context of the DevOps practices. Microservices helped Backtory in a variety of ways, especially in shipping new features more frequently, and in providing scalability for the collective set of users coming from different mobile application developers. Furthermore, we report on a number of migration	Ivon Miranda Santos
Balalaie2016-Microservices_architecture_enables_devops_Migratio	CLOUD-NATIVE ARCHITECTURE > Scalability	What motivated us to perform a migration to a microservices architecture was an issue raised with a require-ment for a Chat as a Service. To implement this requirement, we chose ejabberd due to its known built-in scalability and its ability to run on clusters. To this end, we wrote a python script that enabled ejabberd to perform	Ivon Miranda Santos

Balalaie2016-Microservices_architecture_enables_devops_Migratio	CLOUD-NATIVE ARCHITECTURE > Scalability	As the number of services was growing, another problem was to automate the deployment process and to decouple the build life cycle of each service from others. • The need for built-in scalability: The vision of Backtory is to serve millions of users. By increasing the number of services, we needed a new approach for handling such kind of scalability because scaling services individually requires major efforts and can be error-prone if not handled properly. Therefore, our solution was to locate service instances dynamically through the Service	Ivon Miranda Santos
Balalaie2016-Microservices_architecture_enables_devops_Migratio	CLOUD-NATIVE ARCHITECTURE > Architecture refactoring	2.3 The Target Architecture of Backtory After the Migration To realize microservices architecture and to satisfy our new requirements, we transformed the Backtory's core architecture to a target architecture by performing some refactorings. These changes included introducing	Ivon Miranda Santos
Balalaie2016-Microservices_architecture_enables_devops_Migratio	CLOUD-NATIVE ARCHITECTURE > Architecture refactoring	3 The Migration Process The changes we made throughout the migration process included (i) architectural refactorings and (ii) some necessary changes to enable DevOps. 3.1 Architectural Refactoring Migrating the system towards the target architecture was not a one-step procedure and we performed it incrementally without affecting the	Ivon Miranda Santos
Balalaie2016-Microservices_architecture_enables_devops_Migratio	CLOUD-NATIVE ARCHITECTURE > Architecture refactoring	3.1.7 Introducing ChatServices and DeveloperInfoServices As the final step in the architectural refactoring, we introduced the following services (see the target archi-tecture in Figure 2(g)): • The DeveloperInfoServices by factoring out	Ivon Miranda Santos
Balalaie2016-Microservices_architecture_enables_devops_Migratio	CLOUD-NATIVE ARCHITECTURE > Architecture refactoring	Furthermore, as each team is focused on a particular service, the maintainability and comprehensibility of each service's code is much higher, and new members can be added to the teams with a lower learning curve. During the migration, as depicted in Figure 2(j), we gradually formed small cross-functional teams for each new service constructed as a result of architectural refactorings. Furthermore, we formed a core team consisted of representatives of each service's	Ivon Miranda Santos
Balalaie2016-Microservices_architecture_enables_devops_Migratio	CLOUD-NATIVE ARCHITECTURE > Architecture refactoring	Nevertheless, in practice, a radical interpretation of these promises could result in a chaos in the system and make it even unmaintainable. As a solution, from the beginning of architectural refactoring steps, we started to create service development templates. We have different templates for creating a microservice in Java using different data stores which includes a simple	Ivon Miranda Santos
Balalaie2016-Microservices_architecture_enables_devops_Migratio	CLOUD-NATIVE ARCHITECTURE > Flexibility	• Microservices is not a silver bullet: Microservices was beneficial for us because we needed that amount of flexibility in our system, and that we had the Spring Cloud and Netflix OSS that made our migra-tion and development a lot easier. However, as mentioned before, by adopting microservices several complexities would be introduced to the	Ivon Miranda Santos
Balalaie2016-Microservices_architecture_enables_devops_Migratio	CLOUD-NATIVE ARCHITECTURE > Architecture refactoring	3.2.1 Performance monitoring enables systematic collection of performance data and sharing to enhance decision mak-ing. For example, the Dev team can use such information to refactor the architecture if they find out there exists a performance anomaly in their system The architectural refactorings as the result of pattern applications cannot occur in an ad hoc manner. There exists some invariants that should	Ivon Miranda Santos
Belafia2021-From_monolithic_to_microservice_ar	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	From Mono-lithic to Microservice Architecture:	Ivon Miranda Santos
Belafia2021-From_monolithic_to_microservice_ar	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	From Monolithic to Microservice Architecture: The Case of Extensible and Domain-Specific IDEs	Ivon Miranda Santos
Belafia2021-From_monolithic_to_microservice_architecture_The_ca	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Abstract—Integrated Development Environments (IDEs) are evolving towards cloud-native applications with the aim to relo-cate the language services provided by an IDE on distant servers.	Ivon Miranda Santos
Belafia2021-From_monolithic_to_microservice_architecture_The_ca	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	For these reasons we explore a different software architecture more likely to meet these constraints: the Microservice architecture.	Ivon Miranda Santos

Belafia2021-From_monolithic_to_microservice_architecture_The_ca	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Microservices have been recently introduced by the Soft-ware Engineering community as a support for new distributed and dynamic software architectures. In contrast to traditional monolithic architectures, a microservice architecture consists of small self-contained lightweight services working together. This new architecture has been widely adopted in the case of complex and heterogeneous projects by companies such as Netflix [3], Spotify [4] or Uber [5]. Indeed, microservices provide many benefits for Cloud-	Ivon Miranda Santos
Belafia2021-From_monolithic_to_microservice_architecture_The_ca	CLOUD-NATIVE ARCHITECTURE > Scalability	Microservices have been recently introduced by the Soft-ware Engineering community as a support for new distributed and dynamic software architectures. In contrast to traditional monolithic architectures, a microservice architecture consists of small self-contained lightweight services working together. This new architecture has been widely adopted in the case of complex and heterogeneous projects by companies such as Netflix [3], Spotify [4] or Uber [5]. Indeed, microservices provide many benefits for Cloud-	Ivon Miranda Santos
Belafia2021-From_monolithic_to_microservice_architecture_The_ca	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	This new approach is based on a microservice architecture, which is a deployment organization that separates every functionality of the system as independent services.	Ivon Miranda Santos
Belafia2021-From_monolithic_to_microservice_architecture_The_ca	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	B. Microservice Architectures 1) Definition and Properties of Microservice Architectures: Microservice architecture is an approach of software service design, development and delivery, which has gained a lot of popularity in the recent years. It has been widely used in the industry, more particularly in the case of large-scale applications. For instance, companies such as Netflix [3], Spotify [4] and Uber [5] have been using it as the basis for	Ivon Miranda Santos
Belafia2021-From_monolithic_to_microservice_ar	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	The main characteristic of a microservice architecture is the decomposition of the system in	Ivon Miranda Santos
Belafia2021-From_monolithic_to_microservice_architecture_The_ca	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Such architectures are currently becoming a standard in soft-ware engineering. In [7], the authors emphasize the benefits that microservices architectures can bring to DevOps practices. For instance, microservices are known for their ability to deal with scalability [6], i.e. the ability to make the size of the system evolve as the project evolves. This ability is crucial in a DevOps-driven development, to quickly adapt a system to its usage. In addition to such technical benefits, microservices also allow for more process-related benefits, such as the distribution of the workforce into small and self-managed teams, each focused on a microservice. This fits very well DevOps	Ivon Miranda Santos
Belafia2021-From_monolithic_to_microservice_architecture_The_ca	CLOUD-NATIVE ARCHITECTURE > Scalability	Such architectures are currently becoming a standard in soft-ware engineering. In [7], the authors emphasize the benefits that microservices architectures can bring to DevOps practices. For instance, microservices are known for their ability to deal with scalability [6], i.e. the ability to make the size of the system evolve as the project evolves. This ability is crucial in a DevOps-driven development, to quickly adapt a system to its usage. In addition to such technical benefits, microservices also allow for more process-related benefits, such as the distribution of the workforce into small and self-managed teams, each focused on a microservice. This fits very well DevOps	Ivon Miranda Santos
Belafia2021-From_monolithic_to_microservice_ar	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	2) Transition Process from Legacy Systems Towards a Mi-croservice Architecture:	Ivon Miranda Santos
Belafia2021-From_monolithic_to_microservice_architecture_The_ca	CLOUD-NATIVE ARCHITECTURE > Scalability	a) Analysis of the Driving Forces: First, the analysis of the driving forces is considered as a crucial step of a modernization. Indeed, while the transition to microservice can bring significant improvements to the application in the long term, and ease future developments, transition from legacy systems towards microservice architecture is a long and risky process [9]. Thus, companies often establish cost-benefit analysis, based on the potential benefits such as increased	Ivon Miranda Santos
Belafia2021-From_monolithic_to_microservice_architecture_The_ca	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Indeed, while the transition to microservice can bring significant improvements to the application in the long term, and ease future developments, transition from legacy systems towards microservice architecture is a long and risky	Ivon Miranda Santos

Belafia2021-From_monolithic_to_microservice_architecture_The_ca	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Finally, the microservice architecture is properly defined, by describing the microservices behavior and the way they communicate.	Ivon Miranda Santos
Belafia2021-From_monolithic_to_microservice_architecture_The_ca	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Our work aims to explore the specific case of Cloud-Native IDEs, as a way to analyze the constraints linked to the manipulation of rich data structures, in the case of a migration towards	Ivon Miranda Santos
Belafia2021-From_monolithic_to_microservice_ar	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Figure 2 represents the legacy architecture of Sirius Web and the modern microservice	Ivon Miranda Santos
Belafia2021-From_monolithic_to_microservice_ar	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	2) Microservice Architecture:	Ivon Miranda Santos
Belafia2021-From_monolithic_to_microservice_architecture_The_ca	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Moreover, an AQL expression can be directly evaluated without the need for an initialization phase. However, this approach turns out to be very cumbersome for the rest of the application. Indeed, the Java services and additional packages must be carried between the components. If this can be handled in a monolithic	Ivon Miranda Santos
Belafia2021-From_monolithic_to_microservice_architecture_The_ca	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	However, the microservice architecture prompts the practitioners to develop the microservices to be deployable and usable independently of their	Ivon Miranda Santos
Belafia2021-From_monolithic_to_microservice_architecture_The_ca	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	This approach is based on Jolie31, a programming language designed to implement microservices and specify microservice	Ivon Miranda Santos
Brogi2014-Seaclouds_Seamless_adaptive_multi-cloud_management_of	CLOUD-NATIVE ARCHITECTURE > Scalability	Cloud computing is a model for enabling convenient and on-demand network access to a shared pool of configurable computing resources that can be rapidly provisioned and released with minimal management effort or service provider interaction [1]. The cloud assists to reduce time-to-market and provides on-demand scalability at a low cost for the users. Due to its prospective benefits and potential, cloud computing is a hot research area. Many private and public clouds have emerged during the last years, offering a range of different services at SaaS, PaaS and IaaS levels aimed at matching different user requirements. To take full benefit of the flexibility provided by the cloud, modules of a complex	Ivon Miranda Santos
Brogi2014-Seaclouds_Seamless_adaptive_multi-cloud_management_of	CLOUD-NATIVE ARCHITECTURE > Flexibility	Many private and public clouds have emerged during the last years, offering a range of different services at SaaS, PaaS and IaaS levels aimed at matching different user requirements. To take full benefit of the flexibility provided by the cloud, modules of a complex application should be deployed on multiple clouds depending on their characteristics and strong points. Current cloud technologies suffer from a lack of standardization, with different providers offering	Ivon Miranda Santos
Brogi2014-Seaclouds_Seamless_adaptive_multi-cloud_management_of	CLOUD-NATIVE ARCHITECTURE > Scalability	There is a need for integrating multiple heterogeneous clouds and to solve the problem of distributing services over several providers [4]. Thus, in a scenario where a complex application is distributed on different cloud service providers, a solution is needed in order to manage and orchestrate the distribution of modules in a sound and adaptive way. Such solution should determine the best cloud provider for each particular module based on client requirements (e.g., performance, cost, availability, scalability, etc.). Once the distribution has been decided, the solution should support operations such as managing the relationships between the different modules, maintaining all the specified properties and	Ivon Miranda Santos
Brogi2014-Seaclouds_Seamless_adaptive_multi-cloud_management_of	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	– Dynamic architecture reconfiguration might involve migrating some compo-nents of the application to other cloud providers at runtime.	Ivon Miranda Santos
Brogi2014-Seaclouds_Seamless_adaptive_multi-cloud_management_of	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	– Dynamic architecture reconfiguration might involve migrating some compo-nents of the application to other cloud providers at runtime.	Ivon Miranda Santos
Brogi2014-Seaclouds_Seamless_adaptive_multi-cloud_management_of	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Nonetheless, Broker@Cloud targets a brokering architecture, where the above mentioned services are available, and therefore cannot change the orchestration of the deployed services to adapt to	Ivon Miranda Santos
Brogi2014-Seaclouds_Seamless_adaptive_multi-	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Figure 2 shows the cloud architecture situation currently before SeaClouds (top), and after	Ivon Miranda Santos
Brogi2014-Seaclouds_Seamless_adaptive_multi-	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Cloud architecture before and after SeaClouds.	Ivon Miranda Santos

Brogi2014-Seaclouds_Seamless_adaptive_multi-cloud_management_of	CLOUD-NATIVE ARCHITECTURE > Flexibility	Increased availability and higher security. The usage of formal models to support the management of service-based applications over multi-clouds environments gives more flexibility to reconfigure the distribution as a SLA violation occurs.	Ivon Miranda Santos
Brogi2014-Seaclouds_Seamless_adaptive_multi-cloud_management_of	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Figure 3 shows SeaClouds initial architecture, including the various archi-tectural components and modules employed by the SeaClouds	Ivon Miranda Santos
Brogi2014-Seaclouds_Seamless_adaptive_multi-	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	SeaClouds Architecture.	Ivon Miranda Santos
Brogi2014-Seaclouds_Seamless_adaptive_multi-cloud_management_of	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	The Architecture and design of the SeaClouds platform is in charge of generating the final framework containing all the mechanisms and tools specified and developed during the project	Ivon Miranda Santos
Brogi2014-Seaclouds_Seamless_adaptive_multi-cloud_management_of	CLOUD-NATIVE ARCHITECTURE > Scalability	Dashboard for administration of services distributed between PaaS. This will constitute a graphical Web interface so to have a single-point where to visualize the current configuration of the deployed services and perform changes in their scalability parameters, resource consumption, migration and general management.	Ivon Miranda Santos
Brogi2015-Adaptive_management_of_applications_across_multiple_c	CLOUD-NATIVE ARCHITECTURE > Scalability	Cloud computing is a model for enabling convenient and on-demand network access to a shared pool of configurable computing resources that can be rapidly provisioned and released with minimal management effort or service provider interaction [1]. The cloud assists to reduce time-to-market and provides on-demand scalability at a low cost for the users. Due to its prospective benefits and potential, cloud computing is a hot research area. Many private and public clouds have emerged during the last years, offering a range of different services at SaaS, PaaS and IaaS levels aimed at matching different user requirements. To take full benefit of the flexibility provided by different clouds that offer different services, the modules of a complex application	Ivon Miranda Santos
Brogi2015-Adaptive_management_of_applications_across_multiple_c	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Cloud computing is a model for enabling convenient and on-demand network access to a shared pool of configurable computing resources that can be rapidly provisioned and released with minimal management effort or service provider interaction [1]. The cloud assists to reduce time-to-market and provides on-demand scalability at a low cost for the users. Due to its prospective benefits and potential, cloud computing is a hot research area. Many private and public clouds have emerged during the last years, offering a range of different services at SaaS, PaaS and IaaS levels aimed at matching different user requirements. To take full benefit of the flexibility provided by different clouds that offer different services, the modules of a complex application	Ivon Miranda Santos
Brogi2015-Adaptive_management_of_applications_across_multiple_c	CLOUD-NATIVE ARCHITECTURE > Flexibility	Many private and public clouds have emerged during the last years, offering a range of different services at SaaS, PaaS and IaaS levels aimed at matching different user requirements. To take full benefit of the flexibility provided by different clouds that offer different services, the modules of a complex application should be deployed on multiple clouds depending on their characteristics and strong points.	Ivon Miranda Santos
Brogi2015-Adaptive_management_of_applications_across_multiple_c	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	• Dynamic architecture reconfiguration might involve migrating some components of the application to	Ivon Miranda Santos
Brogi2015-Adaptive_management_of_applicatio	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Nonetheless, Broker@Cloud targets a brokering architecture,	Ivon Miranda Santos
Brogi2015-Adaptive_management_of_applications_across_multiple_c	CLOUD-NATIVE ARCHITECTURE > Flexibility	These platforms separate the PaaS and IaaS layers, thus giving to the users more decision capability on how to deploy their applications, allowing the IaaS selection and defining a set of common services for the communication and data storage. In the scope of commercial solutions, we can find some new platforms that are working on providing flexibility to users allowing the IaaS selection, and in some cases the migration over some specific PaaSes. On the one hand, one example is AppFog, a kind of Cloud Federation, whose system enables the user to select the	Ivon Miranda Santos
Brogi2015-Adaptive_management_of_applications_across_multiple_c	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	This section presents the overall strategy of SeaClouds, describes the reference architecture, and presents its novel aspects compared to	Ivon Miranda Santos

Brogi2015-Adaptive_management_of_applicatio	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Figure 4 shows the cloud architecture situation currently before SeaClouds (top), and after	Ivon Miranda Santos
Brogi2015-Adaptive_management_of_applications_across_multiple_c	CLOUD-NATIVE ARCHITECTURE > Flexibility	Increased availability and higher security. The usage of formal models to support the management of service-based applications over multi-clouds environments gives more flexibility to reconfigure the distribution as a SLA violation occurs.	Ivon Miranda Santos
Brogi2015-Adaptive_management_of_applicatio	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	4.2 SeaClouds architecture	Ivon Miranda Santos
Brogi2015-Adaptive_management_of_applications_across_multiple_c	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	We present the SeaClouds reference architecture, and discuss its novel aspects compared to previous initiatives and efforts.	Ivon Miranda Santos
Brogi2015-Adaptive_management_of_applications_across_multiple_c	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Before describing the core components of the architecture of the SeaClouds platform (Figure 5), it is worth observing that the platform features a graphical user interface (GUI) for two user roles	Ivon Miranda Santos
Brogi2015-Adaptive_management_of_applicatio	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Architecture of the SeaClouds Platform.	Ivon Miranda Santos
Brogi2015-Adaptive_management_of_applicatio	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	A distinguishing aspect of the SeaClouds architecture is that it builds on top of two OASIS	Ivon Miranda Santos
Brogi2015-Adaptive_management_of_applications_across_multiple_c	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	By solving the problems caused by the multiple-vendor scenario, the SeaClouds architecture would benefit not only application developers and cloud providers, but also the whole market, by reducing the adoption barrier for new players.	Ivon Miranda Santos
Brogi2015-Adaptive_management_of_applications_across_multiple_c	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	By solving the problems caused by the multiple-vendor scenario, the SeaClouds architecture would benefit not only application developers and cloud providers, but also the whole market, by reducing the adoption barrier for new players.	Ivon Miranda Santos
Brogi2015-Adaptive_management_of_applications_across_multiple_c	CLOUD-NATIVE ARCHITECTURE > Scalability	Moreover, if some new standard or language for cloud application deployment and management, which is widely accepted by industry, appears in the future, the SeaClouds platform can be scaled more easily to adapt to the new situation, by just modifying the deployer component or adding an interpreter into it for the new emerged language. This scalability and flexibility can also help SeaClouds to avoid the risks resulting from the possible delay or even failure of CAMP and TOSCA standardisation activities, or the case that they are not widely accepted by the industry. In addition, by this separation, the planner and deployer components can be more easily reused,	Ivon Miranda Santos
Brogi2015-Adaptive_management_of_applications_across_multiple_c	CLOUD-NATIVE ARCHITECTURE > Flexibility	Moreover, if some new standard or language for cloud application deployment and management, which is widely accepted by industry, appears in the future, the SeaClouds platform can be scaled more easily to adapt to the new situation, by just modifying the deployer component or adding an interpreter into it for the new emerged language. This scalability and flexibility can also help SeaClouds to avoid the risks resulting from the possible delay or even failure of CAMP and TOSCA standardisation activities, or the case that they are not widely accepted by the industry. In addition, by this separation, the planner and deployer components can be more easily reused,	Ivon Miranda Santos
Brogi2015-Adaptive_management_of_applications_across_multiple_c	CLOUD-NATIVE ARCHITECTURE > Flexibility	Furthermore, by leveraging CAMP, SeaClouds will attract a significant user base (as this standard has a lot of interest but no reference implementations, so far) and advance the standard, ensuring the long-term viability of the benefits implied in SeaClouds, i.e., management and monitoring of underlying providers, performance optimisation, low impact on the code, formal methods support, flexibility to include new services and react to problems at runtime. On the other hand, SeaClouds can provide valuable feedback and contribute to the standardisation effort of CAMP, both by implementing a CAMP-compliant interface towards PaaS providers for management, and by contributing review	Ivon Miranda Santos
Brogi2015-Adaptive_management_of_applications_across_multiple_c	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	The proposed architecture can well support this process, and also the exploitation of the best available offering for each application component at any time. Please note that, thanks to the seamless distribution over several different PaaS platforms, applications developed in SeaClouds will also take advantage of higher availability (via inter-PaaS redundancy), higher security (via inter-	Ivon Miranda Santos

Caceres2022-State-of-the-art_architectures_for_interoperability	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Keywords—cloud computing, cloud architecture, multi-cloud, IaaS, cloud interoperability	Ivon Miranda Santos
Caceres2022-State-of-the-art_architectures_for_interoperability	CLOUD-NATIVE ARCHITECTURE > Flexibility	This growth has shown enduring business change at scale and speed, and forecasts say that it will continue. This article describes emerging architectures of cloud infrastructure that give customers flexibility to use more services than a single cloud provider offers. Each architecture allows a certain degree of multi-cloud integrations, and each comes with own benefits and caveats.	Ivon Miranda Santos
Caceres2022-State-of-the-art_architectures_for_interoperability	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	This article describes emerging architectures of cloud infrastructure that give customers flexibility to use more services than a single cloud provider	Ivon Miranda Santos
Caceres2022-State-of-the-art_architectures_for_interoperability	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	We will discuss limitations of emerging cloud architectures, and present existing solutions to address them.	Ivon Miranda Santos
Caceres2022-State-of-the-art_architectures_for_interoperability	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	However, as providers have been investing more effort into development of niche-focused services, developers need to use new cloud architectures to combine the best of different	Ivon Miranda Santos
Caceres2022-State-of-the-art_architectures_for_interoperability	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	There are several cloud architectures presented in scientific works, in particular in [3].	Ivon Miranda Santos
Caceres2022-State-of-the-art_architectures_for_interoperability	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	In this research, we will pay attention specifically to architectures which involve multi-cloud	Ivon Miranda Santos
Caceres2022-State-of-the-art_architectures_for_interoperability	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	EMERGING MULTI-CLOUD ARCHITECTURES	Ivon Miranda Santos
Caceres2022-State-of-the-art_architectures_for_interoperability	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	In the current article, we will go over new multi-cloud architectures to showcase the need in changing the infrastructure of cloud systems.	Ivon Miranda Santos
Caceres2022-State-of-the-art_architectures_for_interoperability	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	The simplest form of multi-cloud architecture is a Hybrid Cloud, which represents a consolidation of private and public clouds or a composition of private and public IT infrastructure	Ivon Miranda Santos
Caceres2022-State-of-the-art_architectures_for_interoperability	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	After we presented an overview of most common architectures that involve multi-cloud integrations, we can summarize their common problems from the interoperability point of view.	Ivon Miranda Santos
Caceres2022-State-of-the-art_architectures_for_interoperability	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	We have reviewed modern cloud architectures involving more than one cloud and analyzed their up- and downsides in terms of multi-cloud integration possibilities.	Ivon Miranda Santos
daSilva2013-From_the_desktop_to_the_multi-clouds_The_case_of_mo	CLOUD-NATIVE ARCHITECTURE > Scalability	Therefore, the SaaS could allow the definition of high level services, removing the burden of configuring and managing servers from clients. In the point of view of the service providers, this transition is not easy. Concerns such as vendor neutral design, scalability, (self-)adaptation and monitoring of running applications need to be dealt	Ivon Miranda Santos
daSilva2013-From_the_desktop_to_the_multi-clouds_The_case_of_mo	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	Special attention also needs to be paid to the data structures used to design the application and to the architecture it uses internally.	Ivon Miranda Santos
daSilva2013-From_the_desktop_to_the_multi-	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	describes the architecture of MODAClouds.	Ivon Miranda Santos
daSilva2013-From_the_desktop_to_the_multi-clouds_The_case_of_mo	CLOUD-NATIVE ARCHITECTURE > Flexibility	As we are going to present in the following sections, the role of MODAClouds in ModelioSaaS is two-fold. At design time, MODAClouds should support its design and implementation in a cloud provider independent way, reducing development costs, and increasing its flexibility. At run time, it should support the monitoring and adaptation of the application to	Ivon Miranda Santos
daSilva2013-From_the_desktop_to_the_multi-clouds_The_case_of_mo	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	migration, the architecture of our cloud solution will rely on the implementation of the so-called Administration Service.	Ivon Miranda Santos
daSilva2013-From_the_desktop_to_the_multi-	CLOUD-NATIVE ARCHITECTURE > Architecture refactoring	This is even worse when the application's architecture needs to be refactored to a vendor's	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservic	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	A Flexible MicroServices based Architecture for Interoperability in Multi-Cloud Environments	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservices_based_arc	CLOUD-NATIVE ARCHITECTURE > Flexibility	Dealing with ven-dor lock-in in multiple clouds requires addressing two important challenges: interoperability and portability. Some solutions have been proposed to deal with both problems, but most of them fail to provide flexibility. Therefore, we propose PacificClouds, a novel architecture based on microservices for addressing interopera-bility in a multi-cloud environment. PacificClouds differs from previous works by providing greater flexibility due to the microservices architectural pattern. In this article, we also propose a definition of microservices and	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservices_based_arc	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Therefore, we propose PacificClouds, a novel architecture based on microservices for addressing interopera-bility in a multi-cloud	Ivon Miranda Santos

deCarvalho2018-Pacificclouds_A_flexible_microservices_based_arc	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Therefore, we propose PacificClouds, a novel architecture based on microservices for addressing interopera-bility in a multi-cloud	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservices_based_arc	CLOUD-NATIVE ARCHITECTURE > Scalability	According to (Opara-Martins et al., 2015), vendor lock-in is one of the barriers to the adoption of cloud computing. The research further points out that organizations' desire to adopt the cloud for their benefit is primarily related to capacity, scalability, and speed, but they consider urgent	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservic	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	A Flexible MicroServices based Architecture for Interoperability in Multi-Cloud Environments.	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservic	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	A Flexible MicroServices based Architecture for Interoperability in Multi-Cloud Environments.	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservices_based_arc	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	For that matter, the applications are built with the focus on integration with the cloud model, to obtain full cloud advantages; it, also ensures other fe-atures labeled as IDEAL (Isolated state, Distribution, Elasticity, Automated management,	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservices_based_arc	CLOUD-NATIVE ARCHITECTURE > Coupling	For that matter, the applications are built with the focus on integration with the cloud model, to obtain full cloud advantages; it, also ensures other fe-atures labeled as IDEAL (Isolated state, Distribution, Elasticity, Automated management,	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservices_based_arc	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	In this manner, the native cloud application can faci-litate the application deployment in multiple clouds, hence help treat interoperability and	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservices_based_arc	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	In this manner, the native cloud application can faci-litate the application deployment in multiple clouds, hence help treat interoperability and	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservices_based_arc	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Microservices can aid in obtaining the native cloud application's characteristics; therefore, they fo-cus on aspects as componentization of small and lig-htweight services, agile and DevOps practices, in-frastructure automation with continuous delivery fe-atures, decentralized data management, and decen-tralized governance among services. The microser-vices promise more agility, more delivery speed, and more scalability compared with traditional monolithic applications, resulting in less overall cost (Newman, 2015), (RV, 2016). In Section 3, we describe, present	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservices_based_arc	CLOUD-NATIVE ARCHITECTURE > Scalability	Microservices can aid in obtaining the native cloud application's characteristics; therefore, they fo-cus on aspects as componentization of small and lig-htweight services, agile and DevOps practices, in-frastructure automation with continuous delivery fe-atures, decentralized data management, and decen-tralized governance among services. The microser-vices promise more agility, more delivery speed, and more scalability compared with traditional monolithic applications, resulting in less overall cost (Newman, 2015), (RV, 2016). In Section 3, we describe, present	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservices_based_arc	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	In this work, we propose a novel architecture ba-sed on microservices to address interoperability for a multi-cloud environment, called PacificClouds, in or-der to mitigate vendor lock-in	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservices_based_arc	CLOUD-NATIVE ARCHITECTURE > Flexibility	In this work, we propose a novel architecture ba-sed on microservices to address interoperability for a multi-cloud environment, called PacificClouds, in or-der to mitigate vendor lock-in and aid to obtain full cloud advantages. PacificClouds promotes flexibility for user's decisions related to requirements and ap-plication architecture, and for choosing the clouds to execute each microservice of the application. There-fore, PacificClouds differentiate from the other works as it possesses features related to the use of the micro-service and native cloud	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservices_based_arc	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	PacificClouds promotes flexibility for user's decisions related to requirements and ap-plication architecture, and for choosing the clouds to execute each microservice of the application.	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservices_based_arc	CLOUD-NATIVE ARCHITECTURE > Coupling	It possesses decen-tralized governance of the application because each part of the application is independent of one another, i.e., the application modules possess loosely coupled.	Ivon Miranda Santos

deCarvalho2018-Pacificclouds_A_flexible_microservices_based_arc	CLOUD-NATIVE ARCHITECTURE > Flexibility	According to the context aforementioned, the con-tributions of this article are given below: • The proposition and description of PacificClouds, a novel architecture, which provides greater ease in the development of native cloud application, and more flexibility in the deployment and exe-cution of an application in multiple clouds envi-ronment because of microservice use. • A more comprehensive definition of microservice.	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservices_based_arc	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Last, (Dragoni and Lluch-lafuente, 2016) pro-posed two definitions related to microservices: one for microservice and another for microservice archi-	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservices_based_arc	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Therefore, microservice is a cohesive, inde-pendent process interacting via messages, and micro-service architecture is a distributed application where all its modules are	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservices_based_arc	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	Therefore, microservice is a cohesive, inde-pendent process interacting via messages, and micro-service architecture is a distributed application where all its modules are	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservices_based_arc	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Definition (Microservices): are a set of autonomous, independent, self-contained services, in which each service has a single goal, is loosely coupled, and inte-ract to build a	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservices_based_arc	CLOUD-NATIVE ARCHITECTURE > Coupling	Definition (Microservices): are a set of autonomous, independent, self-contained services, in which each service has a single goal, is loosely coupled, and inte-ract to build a	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservices_based_arc	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	PacificClouds is a novel architecture that addresses the interoperability of distributed applica-tion in multiple clouds via microservices.	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservices_based_arc	CLOUD-NATIVE ARCHITECTURE > Flexibility	PacificClouds is a novel architecture that addresses the interoperability of distributed applica-tion in multiple clouds via microservices. Thus, Pa-cificClouds intends to provide flexibility for the soft-ware architect both in making decisions related to ap-plication requirements and the use of the clouds. In addition, PacificClouds can migrate microservice in runtime to better meet application	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservic	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Next, we introduce and explain the PacificClouds architecture.	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservic	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	4.2 PacificClouds Architecture	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservices_based_arc	CLOUD-NATIVE ARCHITECTURE > Flexibility	According to the PacificClouds goals described in section 1, we adopt the multi-cloud delivery model, for it brings greater flexibility. In relation to porta-bility, we assume the three categories used by (Pe-tcu et al., 2013) and we report the IaaS and PaaS le-vels for the portability requirements. According to interoperability levels, we have considered the follo-wing criteria, (Nogueira et al., 2016): we have adop-ted both syntactic and semantic level	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservices_based_arc	CLOUD-NATIVE ARCHITECTURE > Flexibility	PacificClouds intends to address interoperability in the multi-cloud that focuses on the user's per-spective through microservices and native cloud ap-plications. In this manner, it also addresses an impor-tant issue related to the adoption of cloud computing, flexibility. Thus, in this subsection, we present and describe the architecture	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservic	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Thus, in this subsection, we present and describe the architecture PacificClouds.	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservic	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Therefore, PacificClouds ar-chitecture consists of three parts:	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservices_based_arc	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	The Figure 1 illustra-tes the PacificClouds architecture; after, we describe each PacificClouds architecture component.	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservic	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	PacificClouds Architecture.	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservic	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	PacificClouds Core is the central part of the ar-chitecture.	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservic	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	A Flexible MicroServices based Architecture for Interoperability in Multi-Cloud Environments	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservic	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	A Flexible MicroServices based Architecture for Interoperability in Multi-Cloud Environments	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservices_based_arc	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Thus, the PacificClouds will be able to help the software architecture in the process of de- CLOSER 2018 - 8th International Conference on Cloud Computing and Services Science	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservices_based_arc	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Cloud4SOA introduces a broker-based architec-ture whose primary goal is to address semantic in-teroperability challenges at the PaaS layer, based on SOA architecture (Dandria et al.,	Ivon Miranda Santos

deCarvalho2018-Pacificclouds_A_flexible_microservices_based_arc	CLOUD-NATIVE ARCHITECTURE > Flexibility	For this, we offer a summary of the features of all the soluti-ons described in this article, including PacificClouds, related to promoting interoperability in a multi-cloud environment. We consider 10 aspects described in this article, because through them we can observe the le-vel of flexibility of a given solution. In Table 1, there is the only aspect that all soluti-	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservices_based_arc	CLOUD-NATIVE ARCHITECTURE > Flexibility	The difference bet-ween ASMEMA and PacificClouds is that the former does not do decentralized governance by not treating either interoperability or portability. Therefore, Paci-ficClouds is the only proposed solution that promo-tes the decentralized governance of the applications, to the best of our knowledge, allowing the applica-tion modules to have a loosely coupled, in addition to generating less traffic in the network and therefore allowing greater flexibility. According to the analysis performed in this sub-section, we can observe that PacificClouds allows gre-ater flexibility regarding interoperability in a multi-cloud environment. Because of allowing an applica-tion to be distributed over several clouds according to the requirements of the application and the user re-gardless of the technology used,	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservices_based_arc	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	We show in this work a novel architecture, called Pa-cificClouds, that supports the interoperability across multiple clouds, allowing the software architect to choose the application requirements and architecture with no need to worry about the application de-ployment in the clouds.	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservices_based_arc	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	We show in this work a novel architecture, called Pa-cificClouds, that supports the interoperability across multiple clouds, allowing the software architect to choose the application requirements and architecture with no need to worry about the application de-ployment in the clouds.	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservices_based_arc	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	The architecture also al-lows the use of several independent clouds and diffe-rent technologies backgrounds.	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservices_based_arc	CLOUD-NATIVE ARCHITECTURE > Flexibility	The architecture also al-lows the use of several independent clouds and diffe-rent technologies backgrounds. Thus, PacificClouds achieves another goal of flexibility, which helps more cloud computing adoption. The PacificClouds archi-tecture is based on microservices to help achieve	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservic	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	The PacificClouds archi-tecture is based on microservices to help achieve their goals.	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservic	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	The PacificClouds archi-tecture is based on microservices to help achieve their goals.	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservices_based_arc	CLOUD-NATIVE ARCHITECTURE > Flexibility	In this manner, PacificClouds provides other contributions: a) lightweight, since each microservice has only one function; b) decentralized governance of the application, because each microservice is in-dependent of one another, which allows loosely cou-pled; c) the software architect can request at runtime a new SLA by analyzing the monitoring metrics; d) facilitates the use of native cloud application in appli-cation development. One of the contributions of PacificClouds is the flexibility; however, to achieve this, some challenges need to be overcome. One of the more significant challenges is to reach the interoperability of clouds that possess distinct	Ivon Miranda Santos
deCarvalho2018-Pacificclouds_A_flexible_microservices_based_arc	CLOUD-NATIVE ARCHITECTURE > Coupling	In this manner, PacificClouds provides other contributions: a) lightweight, since each microservice has only one function; b) decentralized governance of the application, because each microservice is in-dependent of one another, which allows loosely cou-pled; c) the software architect can request at runtime a new SLA by analyzing the monitoring metrics; d)	Ivon Miranda Santos
dePaula2016-A_systematic_literature_review_on_cloud_computing_a	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	During the conduction of this study, we found four systematic literature reviews (SLRs) focusing on the following themes: migration to the CC [10], service com-position [11], service evaluation [18] and challenges and concerns when building cloud-	Ivon Miranda Santos
dePaula2016-A_systematic_literature_review_on_cloud_computing_a	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	The authors discuss how a proposed model can support companies to analyze several characteristics of their own business as well as pre-existing IT resources to identify their favorability in the migration to the Cloud	Ivon Miranda Santos

dePaula2016-A_systematic_literature_review_on_cloud_computing_a	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	A spectrum of techniques and approaches has been identified that cope with vari-ous concerns, i.e., security and trustworthiness, elasticity, portability and inter-operability, and cloud resilience. In addition, many studies look into reference architectures and cloud-based	Ivon Miranda Santos
Elgedawy2015-Sultan_A_composite_data_consistency_approach_for_s	CLOUD-NATIVE ARCHITECTURE > Flexibility	I. INTRODUCTION Moving business services to global clouds as SaaS services has many benefits (such as flexibility, scalability, reducing capital expenditure on capacity investment), however many businesses are still reluctant to do the move due to its high business risks such as "cloud-vendor lock-in". The cloud-vendor lock-in risk makes businesses vulnerable to price increase and/or	Ivon Miranda Santos
Elgedawy2015-Sultan_A_composite_data_consistency_approach_for_s	CLOUD-NATIVE ARCHITECTURE > Scalability	Moving business services to global clouds as SaaS services has many benefits (such as flexibility, scalability, reducing capital expenditure on capacity investment), however many businesses are still reluctant to do the move due to its high business risks such as "cloud-vendor lock-in". The cloud-vendor lock-in risk makes businesses vulnerable to price increase and/or	Ivon Miranda Santos
Elmroth2011-Self-management_challenges_for_multi-	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Self-management Challenges for Multi-cloud Architectures (Invited Paper)	Ivon Miranda Santos
Elmroth2011-Self-management_challenges_for_multi-cloud_architec	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Addressing the management challenges for a multitude of distributed cloud architectures, we focus on the three complementary cloud management problems of predictive elasticity, admission control, and placement (or scheduling)	Ivon Miranda Santos
Elmroth2011-Self-management_challenges_for_multi-cloud_architec	CLOUD-NATIVE ARCHITECTURE > Flexibility	Despite these recent advances significant research challenges remain in terms of how to achieve e.g., flexibility, robustness, cost-efficiency, and sustainability of cloud infrastruc-tures [2,27,28,29]. Our ongoing efforts combines distributed systems and auto-nomic computing technologies with the overall aim of creating a self-	Ivon Miranda Santos
Elmroth2011-Self-management_challenges_for_multi-cloud_architec	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Figure 1 outlines a conceptual cloud architecture and illustrates the interac-tions between admission control, placement, and elasticity.	Ivon Miranda Santos
Elmroth2011-Self-management_challenges_for_multi-	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Conceptual cloud architecture showing interactions between key management	Ivon Miranda Santos
Elmroth2011-Self-management_challenges_for_multi-cloud_architec	CLOUD-NATIVE ARCHITECTURE > Flexibility	Availability SLAs enable some relaxation of various constraints, e.g., that all VMs need to be assigned physical hosts at all time, and thus increases flexibility in management. For the non-local placement problem, there is a need to handle both the ex-treme cases (1) where, e.g.,	Ivon Miranda Santos
Fowley2018-Cloud_migration_architecture_and_pricing_–_Mapping_a	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	2 Linking Cloud Architecture and Services to License Management and Billing The key concern for the ISV is the product monetisation. A number of technology requirement emerge for a successful SaaS monetisation [10]: – Platform as a service (PaaS) solutions provide development tools and support necessary to create and deliver software (as a service). PaaS generally includes support for service provisioning to the customer, which includes metering, monitoring and auditing tool to determine and analyse who accesses and utilizes the provided resources. – License management allows a service provider to enforce, analyse and man-age use parameters related to how the service is licensed or provisioned. This includes support for typical cloud and also more complex provisioning sce-narios (i.e., beyond per user/per month or unlimited enterprise usage for the latter category). – Billing takes over from monitoring and metering to implement the subscrip-tion model and automate the collection of fees. Billing includes the	Ivon Miranda Santos

Fowley2018-Cloud_migration_architecture_and_pricing_–_Mapping_a	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	4 Cloud Migration – Joint Architecture and Costing Concerns 4.1 Pre-migration – Licensing Model of ISVs The traditional monetisation approach for software vendors involves a licensing model, which needs to be off-set against costs for development and operation. An independent software vendor (ISV) is an organization specializing in mak-ing or selling software, designed for mass or niche markets. This is in contrast to software developed for in-house use only within an organization or software designed or adapted for a single, specific customer.	Ivon Miranda Santos
Fowley2018-Cloud_migration_architecture_and_pricing_–_Mapping_a	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	From a cloud architecture perspective the following concerns need to be considered: – Components: the components of the provided software product - componen-tisation is important to consider their mapping onto different cloud services that might be differently charged. – Usage metering and Billing: applies to the services	Ivon Miranda Santos
Fowley2018-Cloud_migration_architecture_and_pr	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	would be to move from VMs to cloud native applications at platform level.	Ivon Miranda Santos
Fowley2018-Cloud_migration_architecture_and_pricing_–_Mapping_a	CLOUD-NATIVE ARCHITECTURE > Scalability	This is assumed to have better scalability characteristics. It also, as already said, allows better licensing and cost management.	Ivon Miranda Santos
Fowley2018-Cloud_migration_architecture_and_pricing_–_Mapping_a	CLOUD-NATIVE ARCHITECTURE > Scalability	fees, where many providers offer monthly subscription fees, but extra fea-tures will incur extra cost: examples here are scalability, access (IP endpoint, network bandwidth) or monitoring	Ivon Miranda Santos
Fowley2018-Cloud_migration_architecture_and_pricing_–_Mapping_a	CLOUD-NATIVE ARCHITECTURE > Flexibility	This solution is migrated into the cloud to respond to changing business requirements, particularly: – flexibility, allowing different access forms – expansion, requiring to facilitate new customers in new markets The challenge is a high-volume data storage and processing need.	Ivon Miranda Santos
Fowley2018-Cloud_migration_architecture_and_pricing_–_Mapping_a	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	From a technical perspective, we also aim to investigate container technology and a microservices style architecture [23] to see their impact in the context of cloud-native architectures	Ivon Miranda Santos
Fowley2018-Cloud_migration_architecture_and_pricing_–_Mapping_a	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	From a technical perspective, we also aim to investigate container technology and a microservices style architecture [23] to see their impact in the context of cloud-native architectures	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey__evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Scalability	1 INTRODUCTION Many enterprise software applications that support IT services are characterised by the need for high computing capability, scalability, and resource consumption (Buyya et al., 2009, Armbrust et al.,	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey__evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	The complexity of migration is exacerbated by the fact that some legacy applications may have been developed without taking into account the unique requirements attributed to cloud environments such as elasticity, multi-tenancy, interoperability, and refactoring. Such requirements raise new challenges to the migration of applications to the cloud and hence needs improving conventional software development methodologies to address these specific requirements. Various projects and studies in cloud computing community define migration approaches in order to enable legacy applications to take benefit from cloud services	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey__evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Architecture refactoring	The complexity of migration is exacerbated by the fact that some legacy applications may have been developed without taking into account the unique requirements attributed to cloud environments such as elasticity, multi-tenancy, interoperability, and refactoring. Such requirements raise new challenges to the migration of applications to the cloud and hence needs improving conventional software development methodologies to address these specific requirements. Various projects and studies in cloud computing community define migration approaches in order to enable legacy applications to take benefit from cloud services	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey__evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	As will be elaborated in Section 2.3, there are several surveys on the cloud migration each focuses on different aspects of cloud migration such as interoperability, techniques and tools for migration, and cloud architecture design.	Ivon Miranda Santos

Gholami2016-Cloud_migration_process—a_survey__evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	— To help both researchers and practitioners in the cloud community if they want to capture key facets of existing approaches and select or discard one or collection of them that may suite their needs for a particular migration exercise, and — To give a broad view of research challenges, specifically concerned with process models for the legacy to cloud migration that need to be investigated by researchers. Hence, a gateway to new research	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey__evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Architecture refactoring	That is, migration to cloud is a kind of software reengineering where the target application will be able to interact or become integrated with cloud	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey__evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Moving applications to the cloud is similar to conventional legacy application re-engineering. However, cloud applications should also satisfy specific cloud environmental concerns. Drawing on the general literature on cloud computing (Fox et al., 2009, Brebner, 2012, Rimal et al., 2009, Guo et al., 2007, Nathuji et al., 2010, Toosi et al., 2014, Dalheimer and Pfreundt, 2009, Ristenpart et al., 2009), we identified six cloud intrinsic key concerns as follow: (i) resource elasticity, (ii) multi-tenancy, (iii) interoperability and migration over multiple-clouds, (iv) application licensing, (v) dynamicity and unpredictability, and (vi) legal issues. These concerns trigger considerations that an application owner should consider them in the migration process, though they might have been already automatically supported by cloud	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey__evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Moving applications to the cloud is similar to conventional legacy application re-engineering. However, cloud applications should also satisfy specific cloud environmental concerns. Drawing on the general literature on cloud computing (Fox et al., 2009, Brebner, 2012, Rimal et al., 2009, Guo et al., 2007, Nathuji et al., 2010, Toosi et al., 2014, Dalheimer and Pfreundt, 2009, Ristenpart et al., 2009), we identified six cloud intrinsic key concerns as follow: (i) resource elasticity, (ii) multi-tenancy, (iii) interoperability and migration over multiple-clouds, (iv) application licensing, (v) dynamicity and unpredictability, and (vi) legal issues. These concerns trigger considerations that an application owner should consider them in the migration process, though they might have been already automatically supported by cloud	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey__evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Architecture refactoring	Moving applications to the cloud is similar to conventional legacy application re-engineering. However, cloud applications should also satisfy specific cloud environmental concerns. Drawing on the general literature on cloud computing (Fox et al., 2009, Brebner, 2012, Rimal et al., 2009, Guo et al., 2007, Nathuji et al., 2010, Toosi et al., 2014, Dalheimer and Pfreundt, 2009, Ristenpart et al., 2009), we identified six cloud intrinsic key concerns as follow: (i) resource elasticity, (ii) multi-tenancy, (iii) interoperability and migration over multiple-clouds, (iv) application licensing, (v) dynamicity and unpredictability, and (vi) legal issues. These concerns trigger considerations that an application owner should consider them in the migration process, though they might have been already automatically supported by cloud	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey__evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	(i) Resource elasticity. Cloud environment can be viewed as an infinite pool of resources such as CPU, memory, storage, and network bandwidth in a way that resources can be acquired and released by applications based on demand (Fox et al., 2009, Brebner, 2012). Nevertheless, running an application on the cloud does not provide elasticity per se, rather applications need to optimise resource usage in the case of fluctuation in their workload in order to reduce infrastructure cost. Many legacy applications might not have been implemented with a support of dynamic scaling up/down of resources. They assume that elasticity is supported by providing more powerful physical servers. Inevitably, the architecture refactoring of these applications will not be easy and force many modifications in the application	Ivon Miranda Santos

Gholami2016-Cloud_migration_process—a_survey __evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Architecture refactoring	. They assume that elasticity is supported by providing more powerful physical servers. Inevitably, the architecture refactoring of these applications will not be easy and force many modifications in the application tiers.	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey __evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	However, migration from single-tenant to multi-tenant raises several issues, specifically for migration type II. As tenants are dedicated to the same instance of an application, there is often a concern that tenant's QoS is negatively affected	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey __evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Flexibility	APIs). Lack of a proper interoperability can also limit flexibility to use various cloud services where each service provider proposes proprietary interfaces to access services. (iv) Application licensing.	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey __evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Other studies such as (Singh and Chana, 2012) and (Taher et al., 2012) report general challenges of cloud-based application development in terms of cloud-based architecture, component-based development and reusability, quality, design, and	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey __evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Scalability	, 1988), (Ramsin and Paige, 2008), (Sturm and Shehory, 2004), and (Tran and Low, 2005). After analysing these sources, removing redundancy and overlapping between criteria, 11 distinct criteria were derived for the purpose of this study including (1) Process clarity, (2) Procedure and supportive techniques, (3) Tailorability, (4) Development roles, (5) Modelling language, (6) Traceability, (7) Work-products, (8) Formality, (9) Scalability, (10) Tool Support, and (11) Domain applicability. A detailed description of these criteria is presented in Appendix F. Section 5.3 motivates and elaborates each criterion by	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey __evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	This resulted in defining 17 cloud-specific criteria including (1) Analysing Context, (2) Understanding Legacy Application, (3) Analysing Migration Requirements, (4) Planning Migration, (5) Cloud Service/Platform Selection, (6) Training, Re-Architecting Legacy Application (including (7) Incompatibility Resolution, (8) Enabling Multi-Tenancy, (9) Enabling Elasticity, (10) Cloud Architecture Model Definition, (11) Applying Architecture Design Principles), (12) Training, (13) Test and Continuous Integration, (14) Environment Configuration, (15) Continuous Monitoring, (16)	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey __evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	Applied in cloud migration, it meant to transform the legacy application models (e.g. codes and architecture) into platform-independent models, configure them and then generate platform-specific cloud applications using model	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey __evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	As another example, a common concern in designing a new architecture for the legacy application is to define a proper deployment of	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey __evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Identifying potential cloud hosting solution, analysing technical incompatibilities between legacy application and cloud environments, design potential architecture solutions, evaluation QoS of cloud, and incremental architecture scoping and	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey __evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	Identifying potential cloud hosting solution, analysing technical incompatibilities between legacy application and cloud environments, design potential architecture solutions, evaluation QoS of cloud, and incremental architecture scoping and	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey __evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	(ii) SoaML for modelling individual service interfaces, service implementation, and architecture of target application.	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey __evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	[S17] Simple Block Diagram Representing a legacy architecture model and a cloud deployment model of the application.	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey __evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	[S17] Simple Block Diagram Representing a legacy architecture model and a cloud deployment model of the application.	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	[S20] Simple Block Diagram Representing cloud migration patterns to modify legacy application	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	[S20] Simple Block Diagram Representing cloud migration patterns to modify legacy application	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey __evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	[S21] Graph Modeling Legacy source code representation [S26] UML In general [S29] Simple Block Diagram Modelling application architecture before and after adding support for multi-tenancy and architecture decoupling.	Ivon Miranda Santos

Gholami2016-Cloud_migration_process—a_survey__evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Coupling	[S21] Graph Modeling Legacy source code representation [S26] UML In general [S29] Simple Block Diagram Modelling application architecture before and after adding support for multi-tenancy and architecture decoupling.	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey__evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	[S39] UML and OWL Representing the legacy application architecture using UML and transforming it to ontology	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	[S40] Simple Block Diagram Modelling legacy application architecture and dependency tree.	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey__evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Five approaches provide support for traceability between particular activities but not for their whole model: [S12] (Legacy code model → architecture representation → architecture redesign), [S21] (Requirement analysis → Migration plan, Legacy code → Code model → Legacy architecture → Cloud-service architecture model), [S41] (Legacy model → Target architecture → Mapping model → Constraint violation), [S43] (Source code →	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey__evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	From this table, the most frequently recommended work-products are Legacy Application Architecture Model and Cloud Architecture Model as stated in 14 and 12 approaches, respectively.	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey__evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	From this table, the most frequently recommended work-products are Legacy Application Architecture Model and Cloud Architecture Model as stated in 14 and 12 approaches, respectively.	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey__evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Moreover, Aakash and Ali Babar [S21] provide a set of mathematical operators for transforming legacy code and architecture models to cloud-	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey__evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Scalability	5.3.10. Scalability Scalability is the applicability of a methodology to be used for various migration project sizes. Some methodologies can be suitable for moving large and complex workloads from traditional data centers to cloud infrastructures whilst others might be best suited for relatively partial and light	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey__evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Scalability	A migration methodology should be examined if it is appropriate to handle the intended scale of migration. This is done by firstly investigating activities that explicitly support scalability and characteristics that concerned with scalability (such as workload size, degree of interconnectivity of legacy applications and number of applications) and secondly, if the scalability of the methodology has been appraised in real-world scenarios. None of the reviewed approaches supports scalability nor refers to it. Since limited attention has been given to the scalability, future approaches should define the project size that	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey__evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Since produced work-products are stored in a shared repository, they can be accessed and modified other tools; and (ii) REMICS methodology [S26] that includes a set of tools that can be classified in the areas such as requirement management, knowledge recovery from legacy applications, re-transformation of application components to cloud architecture, and model-	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey__evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	Since produced work-products are stored in a shared repository, they can be accessed and modified other tools; and (ii) REMICS methodology [S26] that includes a set of tools that can be classified in the areas such as requirement management, knowledge recovery from legacy applications, re-transformation of application components to cloud architecture, and model-	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey__evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	[S39] Legacy Application Understanding All Available Extracting an ontology of source code, data, and architecture of the legacy application.	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Design Cloud Solution Generating a platform specific implementation model from the cloud	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey__evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	The activity of legacy application understanding aims at recapturing an abstract As-Is representation of application architecture in terms of functionality, different types of dependencies to other applications, interaction points and message follows between application components, and the	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey__evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	Some approaches such as [S12], [S21], and [S41] use produced high-level architecture models of the legacy application as a mean for automatic transformation to target cloud platforms.	Ivon Miranda Santos

Gholami2016-Cloud_migration_process—a_survey __evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	For example, CloudMIG [S41] proposes OMG's Knowledge Discovery Meta-Model (KDM) for extracting a meta-model of application and transforming it to a target cloud architecture.	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	Reconstruct an architecture model of the legacy application and any business entities and	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey __evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Scalability	Common requirements engineering techniques (.e.g. interview, prototyping, and workshop) still are useable to elicit and analysis requirements from users, developers, and managers as it can be seen in approaches [S8] and [S26]. However, the requirement analysis in the context of cloud migration is with a specific focus on elasticity and scalability application requirements [S18], computing requirements [S19], inter-operability requirements for deployment in the cloud [S21], security and regulatory requirements [S23], and storage space requirements in the cloud [S33]. According to Table XVII, in 15 (35%) of reviewed	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey __evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Coupling	For example, Amazon methodology [S40] recommends that for the application that organisation is tightly coupled with complex third parties application, which have not been migrated to the cloud, developers should extend the legacy	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey __evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	(i) Define Cloud Architecture Model. One important aspect of re-architecting is to find suitable components for migration and re-arrangement of their deployment topology in the cloud environments. With respect to this, approaches vary in their support for this sub-activity. For example, Conway et. al. [S4] states this [component selection] will require an understanding of the current state, so that it can be compared to the desired future state. Andrikopoulos et al. [S35] numerate several factors are taken into account for component selection such as data privacy, expected workload profile, acceptable network latency and performance variability, availability zone of cloud providers, the affinity of components in the cloud, and the geographical location of cloud servers. Among the reviewed approaches, Leymann et al. [S11] proposes a concrete technique for component selection and distribution in the cloud via transforming application architecture into a	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey __evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	[S11] proposes a concrete technique for component selection and distribution in the cloud via transforming application architecture into a graph partitioning problem and using existing algorithms such as simulated annealing problem to optimise the distribution of components among	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey __evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Scalability	(ii) Enabling Application Elasticity. Running applications in the cloud does not help to resource efficiency and scalability issues by itself. Rather applications should be able to grow and shrink resources under dynamic workload and maximise resource efficiency (Vaquero et al.,	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey __evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Coupling	[S29], to handle elasticity it is required that application components be decoupled and communicate in a-synchronised manner.	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey __evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Coupling	(i) decoupling databases, (ii) adding tenant configuration databases, (iii) providing tenant configuration interface, (iv) dynamic feature selection, (v) managing tenant data, users, and	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey __evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Coupling	—Application decoupling. To take the advantage of dynamic deployment, one architectural requirement is to decouple application components so that they can interact in a transparent manner. Decoupling also enables independent elastic scaling of the components (i.e. dynamically adding/removing more instances of the same components), minimise time required for code refactoring and test in the case of changing cloud provider, handle a-synchronised communication,	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey __evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Scalability	— Stateless Programming. To support dynamic deployment and independent component scalability, components should be stateless and minimise storing contextual data. Moreover, the coexistence of multiple running instances of the same components in the cloud environment requires a new session management mechanisms in order to track tenant's behaviors and ensure their security when a number of	Ivon Miranda Santos

Gholami2016-Cloud_migration_process—a_survey __evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Scalability	At the managerial level, for example, there is a change from a traditional licensing model to pay-as-you-go or post-usage billing [S18]. On the other hand, developers need training on new programming concepts such as asynchronous interaction, distributed state and session management, caching, scale out across data centers and providers (scalability), multi-tenancy [S35]. Tran et. al. state that training activities should be incorporated into migration process since all cloud services may not support some	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey __evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Scalability	— Other problems. Beyond the above-mentioned problems, the criteria traceability, scalability, and formality have been weakly supported by existing approaches. Furthermore, automation support in performing migration activities has not been sufficiently supported.	Ivon Miranda Santos
Gholami2016-Cloud_migration_process—a_survey __evaluation_framew	CLOUD-NATIVE ARCHITECTURE > Scalability	The fact that each year a considerable number of research papers are published in the field cloud computing, where each reports different solutions, experience reports, and recommendations to move legacy assets to cloud environments, itself is an evidence that the field has reached a maturity point where the development of such a generic reference model is mandatory. Sixthly, some criteria such as traceability, scalability, formality, and automation have been not been properly supported by existing approaches. The current state definitely calls for further enhancement of	Ivon Miranda Santos
Gourisaria2020-An_Extensive_Review_on_Cloud_Computing	CLOUD-NATIVE ARCHITECTURE > Scalability	All the resources offered by the company are at the disposal of the end user which can be used by them as per their necessity [5]. 2.4 Rapid Elasticity and Scalability Cloud technologies are scalable, as they possess the potential to flex with as per the interest of	Ivon Miranda Santos
Gourisaria2020-An_Extensive_Review_on_Cloud_Computing	CLOUD-NATIVE ARCHITECTURE > Scalability	Google App Engine and Amazon Elastic Compute Cloud (EC2) are important examples of public cloud [13]. The advantages of public cloud are inexpensive setup, scalability, seamless uptime and no wastage of resources. The principal concerns pertaining to public cloud are security	Ivon Miranda Santos
Gourisaria2020-An_Extensive_Review_on_Cloud_Computing	CLOUD-NATIVE ARCHITECTURE > Scalability	It is valued in places of fluctuating workloads. The scalability that hybrid cloud brings to the company workforce poses a significant reduction in its infras-tructural and managemental expenditure [15]. 4.4 Community Cloud	Ivon Miranda Santos
Gourisaria2020-An_Extensive_Review_on_Cloud_Computing	CLOUD-NATIVE ARCHITECTURE > Scalability	On the one hand where traditional hosting services rent out the physical servers or a part of it on basis of a subscription model, IaaS providers rent virtual machines based on a utility model. Due to its scalability, this scheme is highly flexible and efficient. It offers a high level of responsibility and control to end consumer as the particular user can issue administrative control over the VM, thereby controlling when it runs, the	Ivon Miranda Santos
Gourisaria2020-An_Extensive_Review_on_Cloud_Computing	CLOUD-NATIVE ARCHITECTURE > Scalability	Developers will no longer have to painstaking acquire the main hardware and get associated with the com-plexities of the infrastructure setup, configuration and management. PaaS offers facilities for application development, application design, testing and deployment, web services, and database integration, security, persistence, storage, scalability, state management, application versioning and application instrumentation. The most common PaaS providers are Force.com,	Ivon Miranda Santos
Gourisaria2020-An_Extensive_Review_on_Cloud_Computing	CLOUD-NATIVE ARCHITECTURE > Flexibility	Some cons are • The usage of an inelastic programming model leads to inflexibility in man-agement of frequency and cost of VMMs to guest changeovers. • Inflated guest to hypervisor transition expenses.	Ivon Miranda Santos
Gourisaria2020-An_Extensive_Review_on_Cloud_Computing	CLOUD-NATIVE ARCHITECTURE > Scalability	Direct-attached storage (DAS), storage area network (SAN) and network-attached storage (NAS) are used in vir-tualization. However, scalability is an important factor to be looked after in virtual storage solutions and a wide range of functionalities have to be offered by them to surpass the great variety offered by normal storage arrays [32].	Ivon Miranda Santos
Gourisaria2020-An_Extensive_Review_on_Cloud_Computing	CLOUD-NATIVE ARCHITECTURE > Scalability	• Fault tolerant • High scalability • Minimal time to migrate workloads from one node to another	Ivon Miranda Santos

Hajjat2010-Cloudward_bound_Planning_for_beneficial_migration_of	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Such hy-brid architectures enable enterprises to benefit from cloud-based ar-chitectures, while honoring application performance requirements, and privacy restrictions on what services may be	Ivon Miranda Santos
Hajjat2010-Cloudward_bound_Planning_for_beneficial_migration_of	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	Such hy-brid architectures enable enterprises to benefit from cloud-based ar-chitectures, while honoring application performance requirements, and privacy restrictions on what services may be	Ivon Miranda Santos
Hajjat2010-Cloudward_bound_Planning_for_beneficial_migration_of	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	In response to these concerns, there has been significant interest in the industry in hybrid architectures where enterprise applications are partly hosted on-premise, and partly in the cloud	Ivon Miranda Santos
Hajjat2010-Cloudward_bound_Planning_for_beneficial_migration_of	CLOUD-NATIVE ARCHITECTURE > Flexibility	Enterprise applications are typically composed of multiple compo-nents, and hybrid architectures allow for individual components to be migrated, or kept local. Hybrid architectures offer enterprises flexibility in decision making that can enable them to find the right balance between privacy considerations, performance and cost sav-ings.	Ivon Miranda Santos
Hajjat2010-Cloudward_bound_Planning_for_beneficial_migration_of	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	Enterprise applications are typically composed of multiple compo-nents, and hybrid architectures allow for individual components to be migrated, or	Ivon Miranda Santos
Hajjat2010-Cloudward_bound_Planning_for_ben	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	Such applications are implemented using multi-tier architectures.	Ivon Miranda Santos
Hajjat2010-Cloudward_bound_Planning_for_beneficial_migration_of	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	While a 3-tiered design is the conventional archi-tecture used in most applications, in practice applications are much more complex.	Ivon Miranda Santos
Hajjat2010-Cloudward_bound_Planning_for_beneficial_migration_of	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	For instance, following the tiered applica-tion architecture, typical security policies may only allow front-end servers to access business logic	Ivon Miranda Santos
Hajjat2010-Cloudward_bound_Planning_for_beneficial_migration_of	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	2.3 Issues in migrating enterprise applications In this paper, we focus on hybrid cloud architectures, where in-dividual application components may be placed locally or in the cloud.	Ivon Miranda Santos
Hajjat2010-Cloudward_bound_Planning_for_beneficial_migration_of	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	2.3 Issues in migrating enterprise applications In this paper, we focus on hybrid cloud architectures, where in-dividual application components may be placed locally or in the cloud.	Ivon Miranda Santos
Hajjat2010-Cloudward_bound_Planning_for_ben	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	In this paper, we address two key challenges associated with re-alizing hybrid cloud	Ivon Miranda Santos
Hajjat2010-Cloudward_bound_Planning_for_beneficial_migration_of	CLOUD-NATIVE ARCHITECTURE > Scalability	Subset of ACLs inheriting rules from a3 after migration. 5.3.3 Performance and scalability To evaluate the scalability of our ACL migration algorithm, we ran it on the reachability policies of the entire campus network, which is a distinct rule-set from that described in § 5.3.1 corresponding to policies in the campus data-center. The network	Ivon Miranda Santos
Haugeland2021-Migrating_monoliths_to_microservices-based_custom	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	Monolithic applications have been the prevailing archi-tecture for enterprise applications after the emergence of frameworks like J2EE around 2000.	Ivon Miranda Santos
Haugeland2021-Migrating_monoliths_to_microservices-based_custom	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	Many companies today still have applications following monolithic architecture where all functions are coupled and built together as a	Ivon Miranda Santos
Haugeland2021-Migrating_monoliths_to_microservices-based_custom	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	Monoliths have many drawbacks. They are large and complicated, making them difficult to change, add new features, or adopt new technologies. Their large sizes also make them slower to move around networks, start or restart on failure and also inhibits scaling with unclear resource requirements. Reliability is impacted as even simple bug fixes cause the entire application to be	Ivon Miranda Santos
Haugeland2021-Migrating_monoliths_to_microservices-based_custom	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	In this paper we explore the current approaches used in the industry when migrating enterprise applications from a monolithic architecture to the microservices architecture and the different approaches used when transitioning an application	Ivon Miranda Santos
Haugeland2021-Migrating_monoliths_to_microservices-based_custom	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Both microservices architecture and multi-tenancy offer additional benefits to the end-users of the application and the developers.	Ivon Miranda Santos
Haugeland2021-Migrating_monoliths_to_microservices-based_custom	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	In the end, we aim to suggest an migration approach for cases where the target application follows a microservice architecture while also allowing tenants to customize the business logic to better fit their needs in a multi-tenant context.	Ivon Miranda Santos
Haugeland2021-Migrating_monoliths_to_microservices-based_custom	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	In the end, we aim to suggest an migration approach for cases where the target application follows a microservice architecture while also allowing tenants to customize the business logic to better fit their needs in a multi-tenant context.	Ivon Miranda Santos
Haugeland2021-Migrating_monoliths_to_microservice	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Migrating to microservices architecture (MSA) is the right way forward for legacy systems to be	Ivon Miranda Santos

Haugeland2021-Migrating_monoliths_to_microservices-based_custom	CLOUD-NATIVE ARCHITECTURE > Scalability	Migrating to microservices architecture (MSA) is the right way forward for legacy systems to be modernized [3], [4]. There are huge benefits for migrating to MSA such as maintainability and scalability in the long run [5], e.g., by adopting DevOps and benefiting from Cloud-native elasticity	Ivon Miranda Santos
Haugeland2021-Migrating_monoliths_to_microservices-based_custom	CLOUD-NATIVE ARCHITECTURE > Architecture refactoring	Our approach focuses on three stages during the migration, analyzing and breaking down the appli-cation into small bounded contexts, transforming the existing infrastructure to fit the new architecture and implementing functionality from the contexts as separate microservice, and finally adding the necessary components to support	Ivon Miranda Santos
Haugeland2021-Migrating_monoliths_to_microservices-based_custom	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	A. Monolithic Applications Monolithic application architecture is a common pattern that software applications follow. The pattern contains all the different layers of an application, including presentation, logic and persistence. All of which are contained within a single deployable package. The monolithic pattern is simple to deploy, scale and develop initially, but as the application grows and becomes more complex and developers encounter some new drawbacks with this pattern. Over time the single code-base for projects grows, and getting a complete understanding of all the internal complexities can become overwhelming. The frequency of changes in the application can potentially be an issue when the size of the	Ivon Miranda Santos
Haugeland2021-Migrating_monoliths_to_microservices-based_custom	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	If the existing infrastructure can not be transformed, or if more effort needed to transform the infras-tructure, we implement additional infrastructure to support the new application	Ivon Miranda Santos
Haugeland2021-Migrating_monoliths_to_microservices-based_custom	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	To validate the approach, we applied it to the SportStore application [22], whose monolithic architecture is simplified in Fig.	Ivon Miranda Santos
Haugeland2021-Migrating_monoliths_to_microservices-based_custom	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	Once the structure and architecture of the existing appli-cation are analyzed and mapped out, we start decomposing the application into separate	Ivon Miranda Santos
Haugeland2021-Migrating_monoliths_to_microservice	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	The API gateway is an integral part of the microservice architecture.	Ivon Miranda Santos
Haugeland2021-Migrating_monoliths_to_microservice	CLOUD-NATIVE ARCHITECTURE > Coupling	Each of these stages aimed to further decouple the microservices from each other.	Ivon Miranda Santos
Haugeland2021-Migrating_monoliths_to_microservices-based_custom	CLOUD-NATIVE ARCHITECTURE > Coupling	The second iteration involves adding a message broker to the application to decouple the services further and adding an asynchronous way for them to orchestrate events af-fecting multiple services.	Ivon Miranda Santos
Haugeland2021-Migrating_monoliths_to_microservices-based_custom	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	Final architecture of the application after migrating all the different services, and implementing the different additional infrastructure components needed for the the new architecture	Ivon Miranda Santos
Haugeland2021-Migrating_monoliths_to_microservices-based_custom	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	C. Implementation In this section, we go through the process following our migration approach and applying it to the migration of the SportsStore application, first to the microservice architecture, and then implementing multi-tenancy for the application.	Ivon Miranda Santos
Haugeland2021-Migrating_monoliths_to_microservices-based_custom	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	C. Implementation In this section, we go through the process following our migration approach and applying it to the migration of the SportsStore application, first to the microservice architecture, and then implementing multi-tenancy for the application.	Ivon Miranda Santos
Haugeland2021-Migrating_monoliths_to_microservices-based_custom	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	In Figure 6 we see the final architecture of the application after all the different services and infrastructure components have been	Ivon Miranda Santos
Haugeland2021-Migrating_monoliths_to_microservices-based_custom	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	We add the additional infrastructure pieces associated with a microservice architecture before we migrate the service.	Ivon Miranda Santos
Haugeland2021-Migrating_monoliths_to_microservices-based_custom	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Once the product service retrieves the message from the queue, it aggregates all the products from the cart into a list before returning it to the cart service After the fourth phase of the migration, the	Ivon Miranda Santos
Haugeland2021-Migrating_monoliths_to_microservice	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	One of the main benefits of the microservice architecture is the decoupled nature of the	Ivon Miranda Santos
Haugeland2021-Migrating_monoliths_to_microservices-based_custom	CLOUD-NATIVE ARCHITECTURE > Coupling	While in [4], the authors present an incremental approach of re-engineering a mission critical banking system that led to reduced complexity, lower coupling, higher cohesion, and a simplified	Ivon Miranda Santos

Hwang2015-Computing_resource_transformation__consolidation_and_	CLOUD-NATIVE ARCHITECTURE > Scalability	However this often brings troubles because applications running on the same server can use the same resource simultaneously, and this can result in slowing down each other. Not only the performance issues, the scalability is an another issue because, for example, the database can not easily be scaled out when it is running in the localhost, together with a consumer application. When an administrator wants to redesign application de-ployment to increase performance and support an elastic scalability in the target cloud, a server that runs multiple networked applications needs to be decomposed into separate components. For example, a high-end physical server that runs many applications can be decomposed into many virtual machines, each	Ivon Miranda Santos
Hwang2015-Computing_resource_transformation__consolidation_and_	CLOUD-NATIVE ARCHITECTURE > Scalability	As shown in Fig-ure 4 that illustrates savings for itemized resources, resources are often overprovisioned for the readiness of the worst case scenarios. However since the cloud computing is meant to provide elastic scalability when the request rate increases, a traditional provisioning strategy that overprovisions resources only charges more operational	Ivon Miranda Santos
Hwang2015-Computing_resource_transformation__consolidation_and_	CLOUD-NATIVE ARCHITECTURE > Scalability	As shown in Figure 8, the total cloudability for virtual servers is limited by the CPU. This tells us that if users want to migrate into the virtual server environment, they need to redesign the application deployment using elastic scalability. One simple example is that a load-balancer can	Ivon Miranda Santos
Hwang2015-Computing_resource_transformation__consolidation_and_	CLOUD-NATIVE ARCHITECTURE > Scalability	E. Server Decomposition As explained in Section III-E, a server can be decom-posed into multiple server instances in order to provide the distributed deployment. The main reason we may consider the decomposition is that the cloud-like application deployment requires modularity to achieve an elastic scalability when the ratio of service requests changes over time. So the condition for the server decomposition is that applications or services	Ivon Miranda Santos
Jambunathan2018-Architecture_decision_on_using_micr	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Architecture Decision on using Microservices or Serverless Functions with Containers	Ivon Miranda Santos
Jambunathan2018-Architecture_decision_on_using_microservices_or	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Cloud adoption is gaining lots of momentum across the globe and enterprise are focussing not only migration on to cloud but also on developing cloud native application.	Ivon Miranda Santos
Jambunathan2018-Architecture_decision_on_using_microservices_or	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Application hosting on Cloud is becoming more popular and becoming inevitable these days and organization are either forced to migrate the existing application onto cloud or developing cloud native application using services provided	Ivon Miranda Santos
Jambunathan2018-Architecture_decision_on_using_microservices_or	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	In the process they are trying to re-architect, re-host, re-platform their application to the new platform or architecture and would like to exploit the features provided by the service providers and make it effective in the cloud environment.	Ivon Miranda Santos
Jambunathan2018-Architecture_decision_on_using_microservices_or	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Application development in cloud becomes more modular, interoperable and distributive in nature and many new evolving technologies like docker[1] containers, microservices[2] and serverless[3] architecture are helping to meet this objective and achieve the desired result.	Ivon Miranda Santos
Jambunathan2018-Architecture_decision_on_using_microservices_or	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	PaaS vs FaaS: Platform-as-a-Service greatly simplifies deploying applications. It allows us to deploy the app and the cloud service provider worries about how to deploy the servers to run it.	Ivon Miranda Santos
Jambunathan2018-Architecture_decision_on_using_microservices_or	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	Serverless architectures refer to applications that significantly depend on third-party services (knows as Backend as a Service or "BaaS") or on custom code that's run in ephemeral containers (Function as a Service or "FaaS").	Ivon Miranda Santos
Jambunathan2018-Architecture_decision_on_using_micr	CLOUD-NATIVE ARCHITECTURE > Scalability	Decreased time to marke Enhanced scalability Improved latency	Ivon Miranda Santos
Jambunathan2018-Architecture_decision_on_using_microservices_or	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Every cloud service provider supports Microservices architecture and has its own API and services to support this development model and enable developers to build a truly distributed	Ivon Miranda Santos

Jambunathan2018-Architecture_decision_on_using_microservices_or	CLOUD-NATIVE ARCHITECTURE > Scalability	As microservices enable smaller, faster releases, they allow new features to be released to only a subset of users initially, and then to the entire user base once the feature meets quality expectations. Even though microservices architecture is more complex, especially at the start, it brings much-needed speed, agility, reliability, and scalability, which are critical to today's DevOps teams.	Ivon Miranda Santos
Jambunathan2018-Architecture_decision_on_using_microservices_or	CLOUD-NATIVE ARCHITECTURE > Scalability	Conversion of Monolithic to Functions Serverless computing is the concept of abstracting away the infrastructure server layer, so that developers can focus on building an app without worrying about scalability, availability, and security of hardware servers. From a usage point of view, the application utilizes as much	Ivon Miranda Santos
Jambunathan2018-Architecture_decision_on_using_microservices_or	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	As for the relationship between serverless computing and microservices architecture, microservices architecture allows small teams of developers to focus on separate services, each providing a specific task, and serverless computing helps them get these microservices up	Ivon Miranda Santos
Jambunathan2018-Architecture_decision_on_using_microservices_or	CLOUD-NATIVE ARCHITECTURE > Scalability	In one sense, it doubles down on the efficiencies that docker containers offer. Docker containers reduce the management burden associated with virtual machines and also provide more scalability. Serverless functions take things a step	Ivon Miranda Santos
Jambunathan2018-Architecture_decision_on_using_microservices_or	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Hence containers provide ideal platform to build a distributed, portable, scalable, secured applications and would be our first choice irrespective of whether developer chose	Ivon Miranda Santos
Jambunathan2018-Architecture_decision_on_using_microservices_or	CLOUD-NATIVE ARCHITECTURE > Scalability	Because of its modular in nature, one can easily containerize the application, dockerize, and deploy it in both public and private cloud environment. With help of the native cloud services, one can avail the scalability, security, fault tolerant features and that makes microservices more appropriate technique to use in the containerized cloud environment. It is not necessary that one need to use container or cloud or both, but with container services in cloud, microservices can be fully utilized and exploited and can be made as a de-facto standard to build modern applications	Ivon Miranda Santos
Jamshidi2013-Cloud_migration_research_A_systematic_review	CLOUD-NATIVE ARCHITECTURE > Flexibility	Larger enterprises are trying to exploit the benefits of this platform [4] by taking business continuity strategies into account [1]. For the former category, the attraction comes from the costing flexibility favoring pay-per-use models rather than upfront purchase of an overprovisioned infrastructure. Scalability, reliability, and	Ivon Miranda Santos
Jamshidi2013-Cloud_migration_research_A_systematic_review	CLOUD-NATIVE ARCHITECTURE > Scalability	For the former category, the attraction comes from the costing flexibility favoring pay-per-use models rather than upfront purchase of an overprovisioned infrastructure. Scalability, reliability, and interoperability of cloud environments are also appealing [5], [4]. Many organizations still rely on so-called legacy systems—software developed over the lifetime of	Ivon Miranda Santos
Jamshidi2013-Cloud_migration_research_A_system	CLOUD-NATIVE ARCHITECTURE > Scalability	. Application scalability (26 percent of the studies). .	Ivon Miranda Santos
Jamshidi2013-Cloud_migration_research_A_systematic_review	CLOUD-NATIVE ARCHITECTURE > Flexibility	However, in the recent version in 2013, in a departure from previous research, cost saving for the first time becomes the primary reason for cloud adoption. A significant percentage (initially 53 percent and in the newer version 46 percent) refer to flexibility as the reason that the cloud brings to the organization. In another survey performed by the same forum, but for industries based in the US, 28 percent of the participants voted for cost saving as opposed to 31 percent for flexibility [31]. Application scalability is reported as a third important driver for cloud adoption. The findings clearly highlight that cost saving, scalabil-ity, and efficient utilization of resources as well as flexibility are key drivers to migrate	Ivon Miranda Santos

Jamshidi2013-Cloud_migration_research_A_systematic_review	CLOUD-NATIVE ARCHITECTURE > Scalability	In another survey performed by the same forum, but for industries based in the US, 28 percent of the participants voted for cost saving as opposed to 31 percent for flexibility [31]. Application scalability is reported as a third important driver for cloud adoption. The findings clearly highlight that cost saving, scalabil-ity, and efficient utilization of resources as well as flexibility are key drivers to migrate	Ivon Miranda Santos
Jamshidi2013-Cloud_migration_research_A_systematic_review	CLOUD-NATIVE ARCHITECTURE > Scalability	4), adding more instances of applications will only influence if the application is engineered for load balancing between the resources. This capability rarely holds for type III while the other types of migration can benefit from this scalability. For a more detailed association of adaptations and cross-cutting concerns to each type refer to [3].	Ivon Miranda Santos
Jamshidi2013-Cloud_migration_research_A_systematic_review	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	As an example, the logical separation between the cloud service providers and the applications, the agreed contract between them is one of the distinguishing aspect of cloud applications that need to be supported by architecture description—see [34] as an initial attempt toward	Ivon Miranda Santos
Jamshidi2013-Cloud_migration_research_A_systematic_review	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	As an example, the logical separation between the cloud service providers and the applications, the agreed contract between them is one of the distinguishing aspect of cloud applications that need to be supported by architecture description—see [34] as an initial attempt toward	Ivon Miranda Santos
Jamshidi2013-Cloud_migration_research_A_systematic_review	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Multi-tenancy refers to a design principle of cloud architecture, where a single instance of the application running on a computing facility on the cloud serves multiple tenants that reside in	Ivon Miranda Santos
Jamshidi2013-Cloud_migration_research_A_systematic_review	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	This might impose a higher risk for business critical systems and a higher cost for enterprise systems than replacing them entirely with a cloud-	Ivon Miranda Santos
Jamshidi2015-Cloud_migration_patterns_A_multi-cloud_service_arc	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	fine-grained service-based cloud architecture migration patterns that target multi-cloud settings and are specified with architectural notations.	Ivon Miranda Santos
Jamshidi2015-Cloud_migration_patterns_A_multi-cloud_service_arc	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	The proposed migration patterns are based on empirical evidence from a number of migration projects, best practices for cloud architectures and a systematic literature review of existing research.	Ivon Miranda Santos
Jamshidi2015-Cloud_migration_patterns_A_multi-	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Cloud architecture Cloud migration Migration pattern Multi-cloud	Ivon Miranda Santos
Jamshidi2015-Cloud_migration_patterns_A_multi-cloud_service_arc	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Mixing cloud architecture with private data centers adds operational efficiency for workload bursts while legacy systems [3] on-premise still support core business services.	Ivon Miranda Santos
Jamshidi2015-Cloud_migration_patterns_A_multi-	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	re-hosted from on-premise to possibly multiple cloud architectures, either private or public ones.	Ivon Miranda Santos
Jamshidi2015-Cloud_migration_patterns_A_multi-	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	We are concerned with the migration of legacy on-premise software to multi-cloud architectures.	Ivon Miranda Santos
Jamshidi2015-Cloud_migration_patterns_A_multi-cloud_service_arc	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	fined-grained core and 6 variant cloud-specific architecture migration patterns, extracted based on empirical evidence from a number of migration projects [5], best practice for cloud architectures [4, 6] and a systematic literature review [1].	Ivon Miranda Santos
Jamshidi2015-Cloud_migration_patterns_A_multi-cloud_service_arc	CLOUD-NATIVE ARCHITECTURE > Flexibility	To account for the situational context of applications, e.g., security, performance, availability needs, existing approaches [1] suggest a trade-off between flexibility and ease of migration using a fixed set of migration strategies. We propose an assembly-based approach based on our experience in situational method engineering [8] where a method is constructed	Ivon Miranda Santos
Jamshidi2015-Cloud_migration_patterns_A_multi-	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	We first introduce architecture migration patterns and the multi-cloud deployment setting.	Ivon Miranda Santos
Jamshidi2015-Cloud_migration_patterns_A_multi-cloud_service_arc	CLOUD-NATIVE ARCHITECTURE > Scalability	Thus, the application component is re-deployed as-is on a cloud platform. The current archi-tecture is mirrored in the cloud, but can take advantage of virtualization to not only reduce operational expenditure, but also to create multiple instances of the application to improve scalability and failover without increasing capital expenditure. The key risk is that underlying architecture issues are not addressed. A monolithic legacy application in the cloud is still monolithic with limitations such as lack of scalability. Scalability is coarse-grained and cannot easily be achieved if, e.g., the	Ivon Miranda Santos

Jamshidi2015-Cloud_migration_patterns_A_multi-cloud_service_arc	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	The current archi-tecture is mirrored in the cloud, but can take advantage of virtualization to not only reduce operational expenditure, but also to create multiple instances of the application to improve scalability and failover without increasing capital	Ivon Miranda Santos
Jamshidi2015-Cloud_migration_patterns_A_multi-cloud_service_arc	CLOUD-NATIVE ARCHITECTURE > Scalability	Use runtime information to monitor and support on-the-fly changes. • Scalability. Scale out to meet bursts in demand and scale in when demand decreases.	Ivon Miranda Santos
Jamshidi2015-Cloud_migration_patterns_A_multi-cloud_service_arc	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Thus, we generalize the architectural elements of these cloud architectures with general concepts of software architecture, i.e.,	Ivon Miranda Santos
Jamshidi2015-Cloud_migration_patterns_A_multi-cloud_service_arc	CLOUD-NATIVE ARCHITECTURE > Scalability	In the cloud, the deployment of high-usage components can be optimized independently of low-usage ones. Re-architecting into independent components reduces dependencies and enables optimization for scalability and performance.	Ivon Miranda Santos
Jamshidi2015-Cloud_migration_patterns_A_multi-	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	(1) on-premise application modernized in isolation, not part of a consistent architecture,	Ivon Miranda Santos
Jamshidi2015-Cloud_migration_patterns_A_multi-	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	A Multi-cloud Service Architecture 9	Ivon Miranda Santos
Jamshidi2015-Cloud_migration_patterns_A_multi-	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	4 Cloud Architecture Migration Patterns	Ivon Miranda Santos
Jamshidi2015-Cloud_migration_patterns_A_multi-	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	The core patterns highlight the different construction principles for the cloud architecture:	Ivon Miranda Santos
Jamshidi2015-Cloud_migration_patterns_A_multi-cloud_service_arc	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	On-premise application is modernized in isolation; Modernization is performed primarily for technical reasons, Com-ponent architecture is only determined bottom-up may need to be re-evaluated because of multi-cloud environment.	Ivon Miranda Santos
Jamshidi2015-Cloud_migration_patterns_A_multi-cloud_service_arc	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	On-premise application is modernized in isolation; Modernization is performed primarily for technical reasons, Com-ponent architecture is only determined bottom-up may need to be re-evaluated because of multi-cloud environment.	Ivon Miranda Santos
Jamshidi2015-Cloud_migration_patterns_A_multi-cloud_service_arc	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Component re-deployed (relocated) on cloud platform is cloudified but without evolution in the application architecture.	Ivon Miranda Santos
Jamshidi2015-Cloud_migration_patterns_A_multi-cloud_service_arc	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	Component re-deployed (relocated) on cloud platform is cloudified but without evolution in the application architecture.	Ivon Miranda Santos
Jamshidi2015-Cloud_migration_patterns_A_multi-	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	A Multi-cloud Service Architecture 11	Ivon Miranda Santos
Jamshidi2015-Cloud_migration_patterns_A_multi-	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	There are initial and target architectures, but also intermediate application architectures.	Ivon Miranda Santos
Jamshidi2015-Cloud_migration_patterns_A_multi-cloud_service_arc	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Note that one path from the source configuration (current on-premise application architecture) to the target (multi-cloud application architecture) will be	Ivon Miranda Santos
Jamshidi2015-Cloud_migration_patterns_A_multi-cloud_service_arc	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	Note that one path from the source configuration (current on-premise application architecture) to the target (multi-cloud application architecture) will be	Ivon Miranda Santos
Jamshidi2015-Cloud_migration_patterns_A_multi-cloud_service_arc	CLOUD-NATIVE ARCHITECTURE > Flexibility	The selected patterns can be integrated based on the presence/absence of overlaps between patterns. The flexibility of this approach is restricted only by the set of available migration patterns. The patterns can be extended over time,	Ivon Miranda Santos
Jamshidi2015-Cloud_migration_patterns_A_multi-	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	A Multi-cloud Service Architecture 13	Ivon Miranda Santos
Jamshidi2015-Cloud_migration_patterns_A_multi-cloud_service_arc	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Applications in a public cloud platform can take advantage of economies of scale and have automated processes for managing.	Ivon Miranda Santos
Jamshidi2015-Cloud_migration_patterns_A_multi-	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Application architecture before migration to the cloud.	Ivon Miranda Santos
Jamshidi2015-Cloud_migration_patterns_A_multi-	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	Application architecture before migration to the cloud.	Ivon Miranda Santos
Jamshidi2015-Cloud_migration_patterns_A_multi-	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Application architecture after migration to the cloud.	Ivon Miranda Santos
Jamshidi2015-Cloud_migration_patterns_A_multi-	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	Application architecture after migration to the cloud.	Ivon Miranda Santos
Jamshidi2015-Cloud_migration_patterns_A_multi-cloud_service_arc	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	We conducted a review [1] aiming to identify, taxonomically classify, and systemat-ically compare the existing research focused on planning, executing, and validating migration of legacy systems towards cloud-based software based on earlier architecture evolution work [13].	Ivon Miranda Santos
Jamshidi2015-Cloud_migration_patterns_A_multi-	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	A Multi-cloud Service Architecture 17	Ivon Miranda Santos

Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	This paper presents Variability-based, Pattern-driven Architecture Migration .V-PAM), a migration method based on (i) a catalogue of fine-grained service-based cloud architecture migration patterns that target multi-cloud, (ii) a situational migration process framework to guide pattern selection and composi-tion, and (iii) a variability model to structure system migration into a	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	KEY WORDS: cloud architecture; microservice architecture; cloud migration; migration pattern; multi-cloud; situational method engineering;	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	KEY WORDS: cloud architecture; microservice architecture; cloud migration; migration pattern; multi-cloud; situational method engineering;	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Mixing cloud architecture with private data centers adds operational efficiency for workload bursts while legacy systems on-premise still support core business services [3].	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Instead of re-architecting applications, they can be re-hosted from on-premise to possibly multiple cloud offerings, either private or public ones. We are con-cerned with the migration of legacy on-premise software to multi-cloud architectures. According to a Gartner report [4], multicloud strategies will become common for 70 percent of organizations by 2019. Multi-cloud deployment is particularly effective in dealing with the following	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	In this paper, we present V-PAM (Variability-based, Pattern-driven Architecture Migration), a cloud architecture migration method, see Figure 1.	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Flexibility	V-PAM defines activities to plan and execute cloud migration [7] based on the concept of patterns or templates, here describing the entities involved in the process. To account for the situational context of applications, for example, security, performance, availability needs, existing approaches suggest a trade-off between flexibility, and ease of migration using a fixed set of migration strategies [1]. We propose an assembly-based approach based on our experience in situational method engineering [8] where a method	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	We present nine core and six variant cloud-specific architecture migration patterns, extracted based on empirical evidence from a number of migration projects [10], best practice for cloud archi-tectures [5, 11] and a systematic literature	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Scalability	Use runtime information to monitor and support on-the-fly changes. Scalability. Scale out to meet bursts in demand and scale in when demand	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	A Systematic Literature Review (SLR) has been used to identify documented patterns. We recorded existing cloud design and architecture patterns [5, 11]. A major role in this process was played by a SLR on cloud migration [1]. We detected shortcomings associated with these design patterns when we applied them in migration planning. The patterns were either limited to specific platforms [5] or fine-grained at a very technical level [11]. To redesign an on-premise application with these patterns requires a deep knowledge of vendor-specific services as well as a fair understanding of detailed design documents. Thus, a migration plan based on these patters cannot be communicated with non-technical stakeholders. Thus, we generalize the architectural elements of these cloud architectures with general concepts of software architecture, as we	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Scalability	In the cloud, the deployment of high-usage components can be optimized independently of low-usage ones. Re-architecting into independent components reduces depen-dencies and enables optimization for scalability and performance.	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	However, challenges remain: (i) on-premise application modernized in isolation, not part of a consistent architecture; (ii) modernization performed primarily for technical reasons resulting in sub-optimal response to business change; (iii) architectures determined bottom-up from existing APIs and transactions may need re-evaluation for	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Central activities of the overall migration process are the guided identification and analysis of fac-tors that might influence the selection of the cloud architecture and the planning of the migration	Ivon Miranda Santos

Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	These factors relate to the characteristics of the main entities in the migration decision – which are the interested organization, the on-premise legacy architecture, and the possible cloud provider	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	At the core of the method, which we detail over the next three sections for a multi-cloud setting, is a collection of architecture-oriented migrations patterns that help to manage and find solutions for the technical constraints in particular.	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Architecture refactoring	The selected patterns are composed to construct a migration plan. The migration plan is then executed by performing sequential architectural refactorings while in each step of the execution the identified constraints need to be satisfied. 4.	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migrat	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	Application and deployment architecture of a video processing system.	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Architecture refactoring	Migration patterns in multi-cloud setting Our migration patterns are sequences of architectural changes (refactoring) in the application deployment setting, through which the current application is gradually modernized. For each migra-tion pattern, an architectural migration	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Scalability	Thus, the application component is re-deployed as-is on a cloud platform. The current architecture is mirrored in the cloud, but can take advantage of vir-tualization to not only reduce operational expenditure, but also to create multiple instances of the application to improve scalability and failover without increasing capital expenditure. The key risk is that underlying architecture issues are not addressed. A monolithic legacy application in the cloud is still monolithic with limitations such as lack of scalability. Scalability is coarse-grained and can-not easily be achieved if, for example, the architecture does not allow the database to be updated by multiple instances.	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	The current architecture is mirrored in the cloud, but can take advantage of vir-tualization to not only reduce operational expenditure, but also to create multiple instances of the application to improve scalability and failover without increasing capital	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	The key risk is that underlying architecture issues are not addressed. A monolithic legacy application in the cloud is still monolithic with limitations such as lack of scalability. Scalability is coarse-grained and can-not easily be achieved if, for example, the architecture does not allow the database to be	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Cloud architecture migration patterns To obtain unambiguous pattern descriptions and to ground pattern-based migration planning in the description, process and variability models, we use a template-based definition of migration pat-terns.	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	The core patterns highlight the dif-ferent construction principles for the cloud architecture: re-deployment, cloudification, relocation,	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migrat	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	There are initial and target architectures, but also intermediate application archi-tectures.	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Note that one path from the source configuration (current on-premise application architecture) to the target (multi-cloud application architecture) will be	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	Note that one path from the source configuration (current on-premise application architecture) to the target (multi-cloud application architecture) will be	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Flexibility	The selected patterns can be integrated based on the presence/absence of overlaps between patterns. The flexibility of this approach is restricted only by the set of available migration patterns. The patterns can be extended over time, for example, by integrating a new solution to new	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migrat	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	Application architecture before migration to the cloud.	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	a financial services application with a hybrid on-premise/cloud architecture and the need for integrated security management,	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	a financial services application with a hybrid on-premise/cloud architecture and the need for integrated security management,	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migrat	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	Application architecture after migration to the cloud.	Ivon Miranda Santos

Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Flexibility	Here a structured, systematic approach proved to be a valuable solution that has helped to deliver cloud solutions that meet the expectations and to keep the projects on track and avoid unnecessary delays. A common problem during migration is the need to refactor the architecture if the aim is to fully benefit from cloud performance and flexibility promises. For instance, the storage refactor-ing options relating to relational, table and blob storage, that we investigated and documented in	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Architecture refactoring	A common problem during migration is the need to refactor the architecture if the aim is to fully benefit from cloud performance and flexibility	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Scalability	For instance, the storage refactor-ing options relating to relational, table and blob storage, that we investigated and documented in [30], are particularly addressed by patterns MP1 and MP3. There, we highlighted the re-architecting options that advanced PaaS clouds offer, but also showed that while quality concerns such as scal-ability or availability are covered, their quantification and a trade-off analysis with cost aspects is not covered. Often, which specific paths are chosen is	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	We conducted a literature review [1] aiming to identify, taxonomically classify, and systematically compare the existing research focused on planning, executing, and validating migration of legacy systems towards cloud computing platforms based on earlier architecture evolution	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	They extend an existing ontology for design pattern description aiming at representing both classical design and cloud architecture patterns together with their architectural implementations.	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migrat	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	However, their pattern notion is only applied to the architecture of cloud-based system.	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Our solution could be extended by applying these cloud patterns, and associated design patterns, to the source (i.e., non-cloud based) and target (i.e., cloud-based) architectures of our architectural	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Differently from Di Martino et al. [42], Bruneliere et al. [43] suggest the TOSCA standard for Topology and Orchestration Specification for Cloud Applications to define cloud architectures.	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	[43] suggest the TOSCA standard for Topology and Orchestration Specification for Cloud Applications to define cloud architectures.	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migrat	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	A language like CloudML [44] could also be utilised for the purpose of architecture description.	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	a way to facilitate the description and deployment of our target cloud architectures through the specification of TOSCA orchestration plans.	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Scalability	Also, based on a model core is the CloudMIG framework proposed by Frey and Hassel-bring [46]. Like our solution, the need to address technical quality concerns like scalability is recognised. At the center is a hierarchy-structured model that guides decision processes and	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migrat	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	In [32], we have reported our experience on migration to microservices	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Based on a monolithic source architecture, an incremental stepwise migration approach to a microservices architecture was implemented.	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Using a Situational Method Engineering migration approach proposed in [49], a monolithic source architecture can be migrated to a target microservices archi-tecture through reusable migration patterns (see our initial catalogue of microservices migration patterns in [49]).	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	We have presented a cloud migration method – V-PAM for Variability-based, Pattern-driven Archi-tecture Migration – built around architecture change patterns, which allows planning the migration of applications for multiple cloud	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migrat	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Architecture-oriented patterns for multi-cloud settings are important for two reasons.	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Firstly, architectures are often refactored to adapt an application to the cloud platform, to benefit more from cloud characteristics such as elasticity or simply to modernize a legacy application.	Ivon Miranda Santos

Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	Firstly, architectures are often refactored to adapt an application to the cloud platform, to benefit more from cloud characteristics such as elasticity or simply to modernize a legacy application.	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Architecture refactoring	Firstly, architectures are often refactored to adapt an application to the cloud platform, to benefit more from cloud characteristics such as elasticity or simply to modernize a legacy application.	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Our implicit assumptions here included the possibility to compo-nentise legacy applications and also to target a cloud native architecture.	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Our implicit assumptions here included the possibility to compo-nentise legacy applications and also to target a cloud native architecture.	Ivon Miranda Santos
Jamshidi2017-Pattern-based_multicloud_architecture_migration	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	Our implicit assumptions here included the possibility to compo-nentise legacy applications and also to target a cloud native architecture.	Ivon Miranda Santos
Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	It is common sense that cloud-native applications (CNA) are intentionally designed for the cloud.	Ivon Miranda Santos
Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Al-though this understanding can be broadly used it does not guide and explain what a cloud-native appli-cation exactly is.	Ivon Miranda Santos
Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	The term “cloud-native” was used quite frequently in birthday times of cloud computing (2006) which seems somehow obvious nowadays.	Ivon Miranda Santos
Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	This paper summarizes the outcomes of a systematic mapping study analyzing research papers covering “cloud-native” topics, research questions and engineering methodologies. We summarize research focuses and trends dealing with cloud-native application engineering	Ivon Miranda Santos
Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Furthermore, we provide a def-inition for the term “cloud-native application” which takes all findings, insights of analyzed publications and already existing and well-defined terminology into account.	Ivon Miranda Santos
Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Cloud-native applications may be a logical continuation of microservice and container-based approaches from an industry point of view	Ivon Miranda Santos
Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	However, some contributions providing valuable insights what additional properties an application must fulfill to be called “cloud-native” compared with classical legacy distributed applications.	Ivon Miranda Santos
Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Because we identified enough studies dealing with cloud-native application specifics, we were able to propose a definition for the term “cloud-native” in Section 5 taking all identified literature	Ivon Miranda Santos
Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	The same is true for selecting relevant papers to derive a definition proposal for cloud-native applications (see Section 5). This process combination is visualized in Fig. 2 and explained	Ivon Miranda Santos
Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	This would have become problematic especially in case of Fehling et al. (2014), which turned out to be one of the most valuable sources in understanding cloud-native applications	Ivon Miranda Santos
Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	In further steps we unified author provided keywords into a unified keyword scheme by aggregating similar keywords into a unified keyword category. All selected papers dealt with 21 detailed research topics of cloud-native applications. We decided to group these detailed research topics into the following major CNA	Ivon Miranda Santos
Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	our focus on cloud-native application literature (we did not search for use cases relevant for cloud applications in general). Other keywords dealt with special aspects like forensics of cloud-native artifacts and cloud costs and accompanying decision making models. However, it seems that there is no clear focus on these aspects	Ivon Miranda Santos
Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	The theoretical validity might be incomplete due to not pub-lished new and controversial views on cloud-native applications. CNAs looking back on 10 years of cloud computing history.	Ivon Miranda Santos

Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	3.1. Origin of the term “cloud-native” [RQ1] According to Fig. 5 a the first contributions using the term "cloud-native” have been published in 2012 (Andrikopoulos et al., 2012; Garca-Gmez et al., 2012). Both conference papers proposed solutions for cloud-native application engineering. Andrikopoulos et al. (2012) proposes a pattern-based solution for a cloud-native application design methodology based on systematically identifying consistency, availability and network partitionability requirements (according to the CAP theorem presented by Fox et al. (1997), Fox and Brewer (1999) and Brewer (2000)). The authors of this contribution are all from the research group headed by Frank Leymann (Institute of Architec-ture and Application Systems, University of Stuttgart, Germany). The paper itself references several other papers dealing with pattern-based approaches. But none	Ivon Miranda Santos
Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Fig. 7 shows the correlation on research topics. The reader sees that the research topic automation platform is intensively corre-lated to cloud-native architecture topics and cloud-native application properties like scalability, resilience	Ivon Miranda Santos
Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Scalability	7 shows the correlation on research topics. The reader sees that the research topic automation platform is intensively corre-lated to cloud-native architecture topics and cloud-native application properties like scalability, resilience and elasticity. The same is true for patterns which are intensively correlated to cloud-native appli-cation design	Ivon Miranda Santos
Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	The same is true for patterns which are intensively correlated to cloud-native appli-cation design topics. So, both papers can be seen as one of the first research papers dealing with and introduced cloud-native topics. In fact, we found Fehling et al. (2014) as one of the most helpful and complete sources systematizing a lot of cloud-native applica-tion aspects. This contribution is derived by and integrates several other publications (beginning with Andrikopoulos et al. (2012)) of the Leymann	Ivon Miranda Santos
Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	3.2. Meaning of the term “cloud-native” [RQ2] Although there exist no precise definition what a cloud-native application exactly is, we can conclude that there is a common but unconscious understanding across all analyzed papers. According to Section 2.4 and Fig. 7, cloud-native applications (CNA) should be build according to corresponding CNA principles (being operated on automation platforms; using softwarization of infrastructure and network; having migration and	Ivon Miranda Santos
Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Scalability	7, cloud-native applications (CNA) should be build according to corresponding CNA principles (being operated on automation platforms; using softwarization of infrastructure and network; having migration and interoperability across different cloud infrastructures and platforms in mind). Following these principles enable to build CNA architectures (mainly service-based and often with self-contained deployment units involved) with spe-cific and wishful CNA properties (horizontal scalability, elasticity, resiliency, isolated strict consistent or eventual consistent state) of resulting applications. To realize CNAs,	Ivon Miranda Santos
Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	According to Fig. 5b, most research focus lays on principles how to design cloud-native applications. The most focused detail topic is to operate cloud-native applications on top of automation platforms. These kind of platforms are named elastic platforms by Fehling et al. (2014). In industry we see this trend as well. There is currently a lot of interest in platforms like Apache Mesos,	Ivon Miranda Santos
Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Another important focus can be seen in appropriate architec-tures of cloud-native	Ivon Miranda Santos
Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	with the trend to use microservice based approaches in cloud-native application engineering (Pahl and Jamshidi, 2016; Balalaie et al., 2015).	Ivon Miranda Santos

Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Scalability	Automation platforms are often mentioned beside architectural topics like mi-croservices, service composition, and deployment units. Furthermore, CNA properties like scalability, resiliency and elasticity are men-tioned very often in the same papers. So, automation platforms are seen by a lot of authors as an enabler to realize sustainable	Ivon Miranda Santos
Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Scalability	The design of sustainable CNA architectures seems to be influenced deeply by pattern based approaches and relies on microservices being build on self-contained deployment units (containers). Furthermore, microservices and CNA properties like scalability, resiliency and elasticity are mentioned very often in the same papers. So, automation platforms, microservices and pattern based approaches seem to be seen as key	Ivon Miranda Santos
Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Fig. 6 shows, that there is a clear focus on solution proposals in research dealing with cloud-native application engineering. The proposals cover mainly principles and architectures how to build	Ivon Miranda Santos
Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Solution proposals would be reflected by a lot of validation/evaluation studies. The complete research would be reflected by a lot of survey studies which might end in new insights. Looking at Fig. 6, we see that this maturity seems not to be reached so far in cloud-native ap-plication	Ivon Miranda Santos
Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	All trends (see Table 7) can be seen as isolated trends. How-ever, according to practitioners they fit very well together and support in a “natural way” to build massively scalable and large scale systems which are more and more often called	Ivon Miranda Santos
Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Scalability	Research implications of CNA properties According to our survey, a CNA shall have wishful CNA property combination of horizontal scalability (Cockcroft, 2015; Goldschmidt et al., 2014), elasticity (Moldovan et al.,	Ivon Miranda Santos
Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	A lot of research about cloud-native applications is reported as solution proposals and experience reports. However, systematic evaluation and validation of postulated effects seems to be rare.	Ivon Miranda Santos
Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Previous Section 4 showed that there are open questions re-garding CNA engineering and research. Furthermore, our literature review did not turn up a common definition that explains what a cloud-native application exactly is.4 Nevertheless, our survey identified enough studies which can be used to derive a definition proposal for CNA because there is a common but	Ivon Miranda Santos
Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Fehling et al. propose that a cloud-native application should be IDEAL, so it should have an [i]solated state, is [d]istributed in its nature, is [e]lastic in a horizontal scaling way, operated via an [a]utomated management system and its components should be [l]oosely coupled (Fehling	Ivon Miranda Santos
Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	According to Stine (Stine, 2015) there are common motivations for cloud-native appli-cation architectures like to deliver software-based solutions more quickly (speed), in a more fault isolating, fault tolerating, and au-tomatic recovering way (safety), to enable horizontal (instead of vertical) application scaling (scale), and finally to handle a huge di-versity of (mobile)	Ivon Miranda Santos
Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Microservices represent the decomposition of monolithic (busi-ness) systems into independently deployable services that do “one thing well” (Newman, 2015; Namiot and Sneps-Sneppe,	Ivon Miranda Santos
Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	The main mode of interaction between services in a cloud-native application architecture is via published and versioned APIs (API-based collaboration). These APIs are often HTTP based and follow a REST-style with JSON serialization, but other protocols and serialization formats can	Ivon Miranda Santos
Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Single deployment units of the architecture are designed and interconnected according to a collection of cloud-focused patterns like the twelve-factor app collection (Adam Wig-gins, 2014), the circuit breaker pattern (Martin Fowler, 2014a) or cloud computing patterns (Fehling et al., 2014; Erl et al.,	Ivon Miranda Santos

Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	And finally, self-service agile infrastructure platforms are used to deploy and operate these microservices via self-contained deployment units (containers). These platforms pro-vide additional operational capabilities on top of IaaS infras-tructures like automated and on-demand scaling of applica-tion instances, application health management, dynamic rout-ing, load balancing	Ivon Miranda Santos
Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	A cloud-native application (CNA) is a distributed, elastic and horizontal scalable system composed of (micro)services which isolates state in a minimum of stateful components. The application and each self-contained deployment unit of that application is designed according to cloud-focused de-sign patterns and operated on a self-service	Ivon Miranda Santos
Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Scalability	(2013). • Scalability can be differentiated to “structural scalability and load scalability. Structural scalability is the ability of a system to expand in a chosen dimension5 without major modifications to its architecture. Load scalability is the ability of a system to perform gracefully as the offered traffic increases” (Bondi,	Ivon Miranda Santos
Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	If sev-eral of these trends are applied in parallel resulting software sys-tems are rated more and more often as “cloud-native applications”.	Ivon Miranda Santos
Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Such cloud-native applications are not just distributed systems.	Ivon Miranda Santos
Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	However, our literature survey showed that there is a common but unconscious understanding of cloud-native applications. A CNA should be build according to corresponding CNA principles (being	Ivon Miranda Santos
Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	CNA architectures are more and more often microservice based. Proposed CNA development methodologies are often pattern-based (relying on comprehensive cloud computing pattern catalogs (Fehling et al., 2014; Erl et al., 2015)) and take DevOps principles into consideration. We could even derive a definition proposal for the term “cloud-native application” and we explained its relationship to well established terms in the context of distributed systems. We think our proposal contributes to a more precise understanding of the term “cloud-native”. Especially if this definition proposal is compared with the often heard but	Ivon Miranda Santos
Kratzke2017-Understanding_cloud-native_applications_after_10_ye	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Acknowledgments This study was funded by German Federal Ministry of Edu-cation and Research (Project Cloud TRANSIT, 03FH021PX4). We would like to thank our reviewers who contributed valuable and very constructive feedback. Furthermore, we thank Dr. Adersberger from QAWARE GmbH. He inspired us with his MEKUNS approach (Mesos, Kubernetes Cloud-native Stack) and his considerations about cloud-native applications from a practitioner point of view. And we thank	Ivon Miranda Santos
Lahmar2018-Multicloud_service_composition_A_s	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	2 CLOUD-COMPUTING ARCHITECTURE	Ivon Miranda Santos
Lahmar2018-Multicloud_service_composition_A_survey_of_current_a	CLOUD-NATIVE ARCHITECTURE > Cloud architecture		Ivon Miranda Santos
Lahmar2018-Multicloud_service_composition_A_survey_of_current_a	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	Practice and Experience Others 2014-2016 5 The computer journal J. New Computer Architectures and Their Applications INFORMS Journal on Computing American Conf. on Software Engineering	Ivon Miranda Santos
Lahmar2018-Multicloud_service_composition_A_survey_of_current_a	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	However, this approach is not always adequate and, in many cases, is unrealistic due to many reasons such as the cloud architecture, various service models, and large-scale cloud environment.	Ivon Miranda Santos

Lahmar2018-Multicloud_service_composition_A_survey_of_current_a	CLOUD-NATIVE ARCHITECTURE > Scalability	However, QoS may also depend on the cloud service model, taking as example the QoS attributes specific to IaaS, such as memory, CPU capacity, accuracy, VM response, throughput, storage, stable time, and failure rate. Quality-of-service attributes for PaaS services include availability, accessibility, security, and reliability, whereas those specific to SaaS include security, reliability, performance, interoperability, scalability, and availability.27 Quality-of-service attributes are generally expressed in the user request, depending on the demanded cloud service model. The user request is the set of requirements expected by the consumer, to be satisfied using a composite service from the multicloud environ-	Ivon Miranda Santos
Lahmar2018-Multicloud_service_composition_A_survey_of_current_a	CLOUD-NATIVE ARCHITECTURE > Scalability	6.1.4 Approaches based on service partitioning and clustering The service partitioning is a technique used to simplify the services treatment for later reuse. It is recommended in order to improve the scalability and the performance of the selected services by regrouping similar ones. From the used service-partitioning methods, we could mention the default method of MapReduce parallel programming model.56 In Dou et al53 and Mezni and Sellami,50k-means and FCA, respectively, are used as clustering	Ivon Miranda Santos
Lahmar2018-Multicloud_service_composition_A_survey_of_current_a	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	From the above few discussed approaches, it is clear that MCSC is a recent and emerging topic. Each of the existing solutions has its own advantages and drawbacks. In this section, we compare the discussed approaches based on the following criteria: • Goal: specifies the main objectives of the approach (eg, reducing the number of clouds, ensuring the privacy of services composition, and decreasing information leakage). • Composition criteria: indicates the criteria that must be satisfied during the composition of services or process fragments. • Service model: denotes the type of composed services in the approach (eg, Web service, process fragment, and mashup). • Composition unit: indicates whether the authors used simple services in the composition or they reused existing process fragments. • Used technique: presents the technique used for the composition (eg, particle swarm optimization, FCA, ACO, clustering, and MapReduce). • Context: indicates whether the composition is realized in a multicloud environment, a cross-cloud environment, an intercloud, etc. • Dimension: indicates whether the service composition is realized in a vertical or horizontal	Ivon Miranda Santos
Lahmar2018-Multicloud_service_composition_A_survey_of_current_a	CLOUD-NATIVE ARCHITECTURE > Flexibility	Therefore, the recent approaches tend to select services from several clouds, such as intercloud in Zhang et al56 and crosscloud in Dou et al53 (see Figure 9). For most researchers, the main objective behind composing services in a multicloud environment was ensuring a high QoS, reliability, flexibility, etc. Despite the advantages of the multicloud environment, we could remark from the above related works that most of the approaches that reuse process fragments are not proposed for reusing SPFs from the cloud environment, such as Yang et al58 and Zemni et al.59,60 Only 1 approach proposed in Nacer et al19 considers reusing fragments from the existing business processes in the multicloud	Ivon Miranda Santos

Lahmar2018-Multicloud_service_composition_A_survey_of_current_a	CLOUD-NATIVE ARCHITECTURE > Flexibility	However, when several clouds are used, the workload becomes difficult to manage by a single manager, such as the cloud combiner in Kurdi et al,49 which is responsible for organizing all the computation and communication flows. To remedy with such problem, Ismail and Cardellini74 made use of so-called decentralized or distributed approaches given their multipros such as flexibility, fault tolerance, and performance, taking as example, the adoption of an agent-based paradigm in Gutierrez-Garcia and Sim.26 The latter exploits the contract net protocol, which is a distributed technique used by the consumers and the providers searching for an agreement. The distributed approaches are efficient in reducing the	Ivon Miranda Santos
Lahmar2018-Multicloud_service_composition_A_survey_of_current_a	CLOUD-NATIVE ARCHITECTURE > Scalability	The complexity of cloud infrastructure translates into more effort needed for the monitoring and adaptation of services and resources in such envi-ronment. The greater scalability and larger size of multicloud environments, compared to traditional cloud or Web service hosting infrastructures, involve more complex monitoring and adaptation systems, which have therefore to be more scalable, robust, and fast. The applications existing in the	Ivon Miranda Santos
Lahmar2018-Multicloud_service_composition_A_survey_of_current_a	CLOUD-NATIVE ARCHITECTURE > Flexibility	Mul-ticloud tools are expected to accelerate services deployment and scaling and to automate monitoring, governance, and configuration tasks. They also can be seen as an abstraction layer that keeps the multicloud complexity under control by hiding the heterogeneity of clouds and providing the needed flexibility to users. That is why leading	Ivon Miranda Santos
Lichtenthaler2019-Requirements_for_a_model-driven_cloud-native_	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	– describes explicitly a model-driven approach to cloud migration, – describes a migration project to a microservices architec-ture or cloud-native architecture – or describes an architecture transformation	Ivon Miranda Santos
Lichtenthaler2019-Requirements_for_a_model-driven_cloud-native_	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	With the emergence of cloud-native applications, the question arises how existing, often monolithic, applications can be migrated to this new	Ivon Miranda Santos
Lichtenthaler2019-Requirements_for_a_model-	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Cloud-native is the predominant paradigm for building web-based applications today [1].	Ivon Miranda Santos
Lichtenthaler2019-Requirements_for_a_model-driven_cloud-native_	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Many new applications are built as a cloud-native application (CNA) right from the start, but there is also a need to migrate existing applications to a CNA style in order to benefit from the advantages	Ivon Miranda Santos
Lichtenthaler2019-Requirements_for_a_model-driven_cloud-native_	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	A key characteristic of CNAs is their fine-grained architecture, meaning that the application is composed of individual components which can be managed and evolved independently.	Ivon Miranda Santos
Lichtenthaler2019-Requirements_for_a_model-driven_cloud-native_	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	“A cloud-native application (CNA) is a dis-tributed, elastic and horizontal scalable system composed of (micro)services [..].	Ivon Miranda Santos
Lichtenthaler2019-Requirements_for_a_model-	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	In these architectures, applications are built through a composition of independent	Ivon Miranda Santos
Lichtenthaler2019-Requirements_for_a_model-	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	In this paper, the most important ones are the fine-grained architectures and the cloud computing	Ivon Miranda Santos
Lichtenthaler2019-Requirements_for_a_model-driven_cloud-native_	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Francesco et al. have investi-gated the approaches used and challenges met in the software industry when migrating to a	Ivon Miranda Santos
Lichtenthaler2019-Requirements_for_a_model-	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	[2] adapted to cloud-native applications	Ivon Miranda Santos
Lichtenthaler2019-Requirements_for_a_model-driven_cloud-native_	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	– describes explicitly a model-driven approach to cloud migration, – describes a migration project to a microservices architec-ture or cloud-native	Ivon Miranda Santos
Lichtenthaler2019-Requirements_for_a_model-driven_cloud-native_	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	2017 Cloud-based architectures for auto-scalable web geoportals towards the cloudification of the GeoVITe swiss academic geoportal [6]	Ivon Miranda Santos
Lichtenthaler2019-Requirements_for_a_model-driven_cloud-native_	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	2017 Extraction of microservices from monolithic software architectures [19] Spillner and Dorodko 2017 Java code analysis and transformation into AWS lambda functions [20] Bucchiarone et al.	Ivon Miranda Santos
Lichtenthaler2019-Requirements_for_a_model-driven_cloud-native_	CLOUD-NATIVE ARCHITECTURE > Coupling	These nanoentities are then connected on the basis of so-called coupling criteria to derive groups of nanoentities which form services.	Ivon Miranda Santos
Lichtenthaler2019-Requirements_for_a_model-driven_cloud-native_	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Nevertheless, it is an impor-tant part of the application and because of the recommended distributed data governance for microservice architectures, it should also be considered for	Ivon Miranda Santos

Lichtenthaler2019-Requirements_for_a_model-driven_cloud-native_	CLOUD-NATIVE ARCHITECTURE > Scalability	The modeling approach should differentiate between synchronous and asynchronous communication to enable a shift towards asynchronous communication, where possible. Asynchronous communication via message passing is preferable in CNAs to ensure loose coupling of components [5] leading to better scalability and resilience to failures. Finally,	Ivon Miranda Santos
Lichtenthaler2019-Requirements_for_a_model-driven_cloud-native_	CLOUD-NATIVE ARCHITECTURE > Coupling	Asynchronous communication via message passing is preferable in CNAs to ensure loose coupling of components [5] leading to better scalability and resilience to failures.	Ivon Miranda Santos
Lichtenthaler2019-Requirements_for_a_model-driven_cloud-native_	CLOUD-NATIVE ARCHITECTURE > Coupling	Finally, the differentia-tion between synchronous and asynchronous communication between components becomes more important and a shift to asynchronous communication should be considered during the migration to promote loose	Ivon Miranda Santos
Lichtenthaler2019-Requirements_for_a_model-driven_cloud-native_	CLOUD-NATIVE ARCHITECTURE > Scalability	Especially Kubernetes is increasingly chosen as one of the primary technologies for running CNAs [24]. The information in the PIM is rather abstract, for example consid-ering the scalability requirements of a component. The PSM should therefore include the available technological choices from a cloud provider, as also mentioned	Ivon Miranda Santos
Lichtenthaler2019-Requirements_for_a_model-driven_cloud-native_	CLOUD-NATIVE ARCHITECTURE > Coupling	The UML diagram also gives us a basic dependency graph for the classes and methods. This dependency graph highlights the tight coupling within the application. For a meaningful decomposition, however, the information that one component depends on another is not sufficient. Infor-mation about the strength of the relationship	Ivon Miranda Santos
Lichtenthaler2019-Requirements_for_a_model-driven_cloud-native_	CLOUD-NATIVE ARCHITECTURE > Coupling	The coupling of a system is therefore not only present in the appli-cation logic, but also in the databases and for a comprehensive migration, also the database needs to be addressed.	Ivon Miranda Santos
Lichtenthaler2019-Requirements_for_a_model-driven_cloud-native_	CLOUD-NATIVE ARCHITECTURE > Coupling	The argument that the coupling of an applica-tion is visible not only in the application logic, but also in the database, can be extended to the UI as	Ivon Miranda Santos
Lichtenthaler2019-Requirements_for_a_model-driven_cloud-native_	CLOUD-NATIVE ARCHITECTURE > Coupling	When an application is split up on the server-side by decoupling entities from each other, the server-side cannot serve an aggregate of different entities	Ivon Miranda Santos
Lichtenthaler2019-Requirements_for_a_model-driven_cloud-native_	CLOUD-NATIVE ARCHITECTURE > Flexibility	Therefore, we state that this phase can only be semi-automated and the ideal solution would be that a human expert can transform the architecture supported by a tool. The tool provides a frame for the envisioned CNA including constraints from the system, but also provides the flexibility to architect a fitting solution for the migration. A potential extension could be to include common blueprints or patterns of CNAs which	Ivon Miranda Santos
Lichtenthaler2019-Requirements_for_a_model-driven_cloud-native_	CLOUD-NATIVE ARCHITECTURE > Flexibility	For the final step of generating code out of the PSM of the cloud-native system, a reasonable approach is to only generate a skeleton of the envisioned system. The actual implementation is still left to the software engineers to ensure the correct functionality and enabling flexibility for the fin-ishing touches.	Ivon Miranda Santos
Lichtenthaler2019-Requirements_for_a_model-driven_cloud-native_	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Finally, there are approaches which aim to refine the architecture of an existing, but already microservices based, application.	Ivon Miranda Santos
Maniah2022-A_systematic_literature_review_Risk_analysis_in_clou	CLOUD-NATIVE ARCHITECTURE > Scalability	, 2011). In addition, users of cloud services can optimize server utilization, dynamic scalability, and minimize the development of new appli-cation life cycles (Al-Ruithe et al., 2019).	Ivon Miranda Santos
Maniah2022-A_systematic_literature_review_Risk_analysis_in_clou	CLOUD-NATIVE ARCHITECTURE > Scalability	Several other reasons for companies migrating to the cloud are: (1) because cloud comput-ing services have scalability, which means that they can meet the needs of information technology resources according to company needs; (2) because the cloud provider has provided settings for both hardware configuration and software updates or server set-tings and others, so that companies as cloud service users are more focused on developing better innovative products; (3) because the cloud provider has a data center that provides fast and efficient computing services, so this will have an effect on high performance in the cloud compared to the data center owned by the company. Based on data from cisco.com, it is estimated that in 2020 cloud	Ivon Miranda Santos

Mateen2021-A_dynamic_decision_support_system_for_selection_of_c	CLOUD-NATIVE ARCHITECTURE > Scalability	2021, 11, 11296 3 of 32 • Scalability Limits on the services can be increased or decreased with respect to (w.r.t) users' or	Ivon Miranda Santos
Mateen2021-A_dynamic_decision_support_system_for_selection_of_c	CLOUD-NATIVE ARCHITECTURE > Scalability	, a data center. Scalability Limits on the services can be increased or decreased with respect to (w.r.t) users' or	Ivon Miranda Santos
Mateen2021-A_dynamic_decision_support_system_for_selection_of_c	CLOUD-NATIVE ARCHITECTURE > Scalability	This flexible sharing concept attracts different organizations, which are interested in moving their applications, websites, and online stores to cloud-based solutions. IT-based organizations are migrating their data to the cloud because of several issues such as storage space, hardware availability, scalability, billing expenses, and data recovery [4]. Decades ago, data were being accessed and stored on the local system, which	Ivon Miranda Santos
Mateen2021-A_dynamic_decision_support_system_for_selection_of_c	CLOUD-NATIVE ARCHITECTURE > Scalability	It is free for organizations for a trial period with limited features, but the payment is necessary for each plan afterwards. Organizations are interested in migrating their legacy local data storage to cloud-based storage to get maximum benefits with less cost, sharing, consistency and scalability. The number of cloud storage providers	Ivon Miranda Santos
Mateen2021-A_dynamic_decision_support_system_for_selection_of_c	CLOUD-NATIVE ARCHITECTURE > Scalability	An example is assurances from KPIs, like availability and stability of service [9]. The SMI model summarizes the most significant attributes for the service quality such as assurances of usability, cost, agility, performance, scalability, security, accountability, and privacy. The metamodeling technique [10] was adopted to	Ivon Miranda Santos
Mateen2021-A_dynamic_decision_support_system_for_selection_of_c	CLOUD-NATIVE ARCHITECTURE > Scalability	SMICloud-Monitoring identifies all those cloud services that can accomplish the customer's essential QoS requests. In order to identify an appropriate cloud service, it validates user requirements based on attributes such as the number of CPU cores, storage size, scalability, memory, bandwidth, and synchronization. The SMICloud-Service Catalogue module is used for	Ivon Miranda Santos
Mateen2021-A_dynamic_decision_support_system_for_selection_of_c	CLOUD-NATIVE ARCHITECTURE > Scalability	Users enter their requirements as numerical or non-numerical values, where numerical values include the number of CPU cores, storage capacities, trial period availability, etc., whereas non-numerical values consist of attributes like file synchronization and scalability. Output from the user interface are the parameters entered by the	Ivon Miranda Santos
Mateen2021-A_dynamic_decision_support_system_for_selection_of_c	CLOUD-NATIVE ARCHITECTURE > Scalability	Cloud computing has substantial benefits that are typically associated with its acceptance, like decreasing functional costs. The computing resource seems unlimited, and scalability is another beauty of cloud computing, where resources can be increased or decreased at	Ivon Miranda Santos
Mateen2021-A_dynamic_decision_support_system_for_selection_of_c	CLOUD-NATIVE ARCHITECTURE > Scalability	The decision to shift data to the cloud environment is not a simple task as it involves multiple contradictory aspects. Some of these complex factors include performance, pri-vacy, data security, legal concerns, scalability, service	Ivon Miranda Santos
Mateen2021-A_dynamic_decision_support_system_for_selection_of_c	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	The proposed DSS architecture includes a web data scraper, web data parser, cloud provider ranking function, comparison function, and user	Ivon Miranda Santos
Mateen2021-A_dynamic_decision_support_system_for_selection_of_c	CLOUD-NATIVE ARCHITECTURE > Scalability	It is clearly shown from the results that Google Drive and IBM Cloud are leading other cloud storage vendors due to their features. The IBM cloud provides dynamic scalability, monitoring tools and technology integration while Google Drive attracts users with free storage, offline features and attractive user interface. On the other hand, iCloud is at the bottom due to access from	Ivon Miranda Santos
Mateen2021-A_dynamic_decision_support_system_for_selection_of_c	CLOUD-NATIVE ARCHITECTURE > Scalability	, user requirements) that can enhance the comparison and ranking modules, e.g., internet bandwidth, geographical location, organized versus unorganized scalability, and comparison of SLAs.	Ivon Miranda Santos
Monrat2019-A_Survey_of_Blockchain_From_the_Perspectives_of_Appl	CLOUD-NATIVE ARCHITECTURE > Scalability	A number of research studies focus on the opportunity that blockchain provides in various application domains. This paper presents a comparative study of the tradeoffs of blockchain and also explains the taxonomy and architecture of blockchain, provides a comparison among different consensus mechanisms and discusses challenges, including scalability, privacy, interoperability, energy consumption and regulatory issues. In addition, this paper also	Ivon Miranda Santos

Monrat2019-A_Survey_of_Blockchain_From_the_Perspectives_of_Appl	CLOUD-NATIVE ARCHITECTURE > Scalability	Despite the fact that blockchain technology shows great potential that may replace many of the current digital plat-forms, it has some technical constraints. Scalability is a huge concern for blockchain based platforms [9]. In Bitcoin, the limited size and frequency of the blocks along with the number of transactions the network can process can be con-sidered a scalability problem [10]. The average block creation time in Bitcoin is 10 minutes, and the block size is limited to 1	Ivon Miranda Santos
Monrat2019-A_Survey_of_Blockchain_From_the_Perspectives_of_Appl	CLOUD-NATIVE ARCHITECTURE > Flexibility	Survey of Blockchain From the Perspectives of Applications, Challenges, and Opportunities ensure the programmatic descriptions of anticipated power flexibility degrees, the validation and tractability of demand response agreements, and the balance between power needs and generation. Furthermore, blockchain can be used to enable energy trading in the Industrial Internet	Ivon Miranda Santos
Monrat2019-A_Survey_of_Blockchain_From_the_Perspectives_of_Appl	CLOUD-NATIVE ARCHITECTURE > Scalability	However, similar to other emerg-ing technologies, blockchain has its limitations and is not feasible for many all types of business model. This section describes the issues and challenges of blockchain technology as the following: performance & scalability in Section V-A, privacy in Section V-B, interoper-ability in Section V-C, energy consumptions in Section V-D, selfish mining in Section V-E and current regulation problems in Section V-F. A. PERFORMANCE & SCALABILITY Cryptocurrency and blockchain-based solutions for differ-ent business models are gaining popularity. However, there is a concern regarding whether it could meet up with the increasing demand coming from different business and gov-ernment based sectors, especially regarding performance and scalability. Recently, researchers are working to address the scalability issues regarding the number of replicas in the network as well the performance concern, such as throughput (number of transactions per second) and latency (required time for adding a block of transactions in the blockchain) [99]. Increasing the number of replicas can have a detrimental effect on the throughput and latency because the network needs to deal with the increased amount of message exchange and processing. Although protocols such as PoW can ensure scalability, it is	Ivon Miranda Santos
Monrat2019-A_Survey_of_Blockchain_From_the_Perspectives_of_Appl	CLOUD-NATIVE ARCHITECTURE > Scalability	PBFT has quadratic message complexity that requires nxn broadcast for n replicas. Although this overhead ensures that consensus will be reached having malicious replicas or Byzantine fail-ures, it creates scalability issues. Any mainstream platform needs to process hundreds and thousands of transactions per second. Otherwise, the economy could not keep moving on without massive delays for consumers and businesses, which proves that scalability and performance is an important concern for this emerging technology. Meanwhile, as the capacity of blocks is very small,	Ivon Miranda Santos
Monrat2019-A_Survey_of_Blockchain_From_the_Perspectives_of_Appl	CLOUD-NATIVE ARCHITECTURE > Scalability	The DCS (decentralized, consistent and scalable) theorem, proposed by Slepak et el., had also emphasized on issues related to scal-ability, such as blocksize [101]. By using the DCS triangle, they showed that decentralized blockchain system can not have all the properties of DCS	Ivon Miranda Santos
Monrat2019-A_Survey_of_Blockchain_From_the_Perspectives_of_Appl	CLOUD-NATIVE ARCHITECTURE > Scalability	For instance, the Financial Action Task Force reported in 2015 on how the founders of Liberty Reserve were able to launder hundreds of millions of US dollars for six years to criminal organizations. Blockchain's wider and deeper applications are potentially constrained by limitations posed by techni-cal/scalability challenges, business model challenges, scan-dals and public perception, government rules and	Ivon Miranda Santos
Monrat2019-A_Survey_of_Blockchain_From_the_Perspectives_of_Appl	CLOUD-NATIVE ARCHITECTURE > Scalability	Specifically, for the financial services sector, blockchain needs to overcome ten key hurdles before becoming a reality in the sector. These include matters concerning with its costs and benefits, cost mutualization, incentives alignment, evolving standards, scalability, gov-ernance, legal risks, security, simplification and regulatory interventions. Laws and regulations could impact	Ivon Miranda Santos

Monrat2019-A_Survey_of_Blockchain_From_the_Perspectives_of_Appl	CLOUD-NATIVE ARCHITECTURE > Scalability	There are still many open issues that need to be further researched and analyzed to cre-ate more workable and effective industrial applications that can fully benefit from the use of blockchain and achieve the intended goals. Examples of these open issues include secu-rity, privacy, scalability, energy issues, and integration with other systems and, more specifically, with regulatory issues. Future work in this field is required to address these issues and close the gaps for more efficient,	Ivon Miranda Santos
Mostajabi2021-A_Systematic_Review_of_Data_Models_for_the_Big_Da	CLOUD-NATIVE ARCHITECTURE > Scalability	The high volume, high variety, and high velocity of data have recently posed the challenge of 3Vs to this field, also known as the Big Data Problem. The 3Vs dimensions of complexities for the big data entails high-speed storage, scalability of database systems, suitable data models, real-time responsiveness and so on. Data model, as the representation schema of data is an essential	Ivon Miranda Santos
Mostajabi2021-A_Systematic_Review_of_Data_Models_for_the_Big_Da	CLOUD-NATIVE ARCHITECTURE > Scalability	This yielded a mound of worthless results when ran on complex data, thereby reducing the efficiency of the queries. With the increase in the volume and the change in the variety of data over time, the unscalability problem arose in the relational data models. The other deficiencies of these models were the lack of open-source	Ivon Miranda Santos
Mostajabi2021-A_Systematic_Review_of_Data_Models_for_the_Big_Da	CLOUD-NATIVE ARCHITECTURE > Scalability	The XML model was devel-oped to manage semi-structured data [5], [6]. The object-relational [7] was another model developed, still facing the scalability problem. Additionally, the explosive	Ivon Miranda Santos
Mostajabi2021-A_Systematic_Review_of_Data_Models_for_the_Big_Da	CLOUD-NATIVE ARCHITECTURE > Scalability	Accord-ing to [9], this growth rate doubles every two years and has increased tenfold within the years 2013 to 2020 (from 4.4 to 44 ZB). There was an urgent need for horizontal scalability and greater flexibility of databases due to the exponential growth of data volume, the change of data from structured to semi-structured [10] and non-structured, and the challenge of their storage. Hence, some practical solu-tions have been	Ivon Miranda Santos
Mostajabi2021-A_Systematic_Review_of_Data_Models_for_the_Big_Da	CLOUD-NATIVE ARCHITECTURE > Flexibility	Accord-ing to [9], this growth rate doubles every two years and has increased tenfold within the years 2013 to 2020 (from 4.4 to 44 ZB). There was an urgent need for horizontal scalability and greater flexibility of databases due to the exponential growth of data volume, the change of data from structured to semi-structured [10] and non-structured, and the challenge of their storage. Hence, some practical solu-tions have been	Ivon Miranda Santos
Mostajabi2021-A_Systematic_Review_of_Data_Models_for_the_Big_Da	CLOUD-NATIVE ARCHITECTURE > Scalability	It has emerged in a general term with unstable and flexible conceptual data models [16] and varied strategies for the databases. Among features of NoSQL databases are high scalability, availability without the need for ACID feature support [9], open-source possibility of the presented models [17], and capability of dynamic data modeling. Given the variety and various	Ivon Miranda Santos
Mostajabi2021-A_Systematic_Review_of_Data_Models_for_the_Big_Da	CLOUD-NATIVE ARCHITECTURE > Flexibility	In general, databases can be categorized into two groups of schema-based and schema-less. The former utilizes a schema to describe the structure of the database, while there is no need to predefine a specific data structure for the latter, which leads to the higher flexibility of the database. The data is stored regarding a predefined structure of schema, which can be	Ivon Miranda Santos
Mostajabi2021-A_Systematic_Review_of_Data_Models_for_the_Big_Da	CLOUD-NATIVE ARCHITECTURE > Scalability	In the process of reviewing, it was attempted to find answers to the questions listed in section III considering the nature and the content of research. In [4], Open Scalable Relational Data Model (OSRDM) was introduced, which supports data variety with full hori-zontal scalability compared to the relational data model. This data model exhibits similar properties of performance and scalability to the NoSQL data model. In this model, a public key	Ivon Miranda Santos
Mostajabi2021-A_Systematic_Review_of_Data_Models_for_the_Big_Da	CLOUD-NATIVE ARCHITECTURE > Scalability	key-value data models. In other words, this model developed on a key-value model with split keys and public hashes, pro-viding full horizontal scalability. Feature keys are also used for flexible	Ivon Miranda Santos

Mostajabi2021-A_Systematic_Review_of_Data_Models_for_the_Big_Da	CLOUD-NATIVE ARCHITECTURE > Scalability	Compared to the relational, column-based, and document-based data models, various data structures are supported by the model and various data types are definable by the users. Representing data relationships as entities and separating entities in the data structure provide this model with full horizontal scalability. In the design of NoSQL database [19], a model-based process, called Mortadelo, was planned to automate database implementation when	Ivon Miranda Santos
Mostajabi2021-A_Systematic_Review_of_Data_Models_for_the_Big_Da	CLOUD-NATIVE ARCHITECTURE > Scalability	In the column-based model, there was no need for encryption and decryption of information based on the storage path of stored data in the linked columns. The scalability of the column-based data model was finally investigated from an operational point of view. It was found that the column-based model had better performance than other data models on tracking specific data for path-based	Ivon Miranda Santos
Mostajabi2021-A_Systematic_Review_of_Data_Models_for_the_Big_Da	CLOUD-NATIVE ARCHITECTURE > Scalability	The sec-ond algorithm executes data partitioning, virtual machine preparation, and deprivation reduction automatically. It also develops scalability of workflows in the MapReduce style and a new DataView for workflows of big data systems. The suggested model was used to analyze collected data from vehicles and provide insight into	Ivon Miranda Santos
Mostajabi2021-A_Systematic_Review_of_Data_Models_for_the_Big_Da	CLOUD-NATIVE ARCHITECTURE > Scalability	In [43], the NoSQL data models were compared with the relational data model and categorized according to their key features. These features include horizontal scalability, parti-tioning ability of large datasets in distributed sources, data replicability for fault tolerance, and mechanism facility for data item compatibility. Other properties of data placement, partition, replication,	Ivon Miranda Santos
Mostajabi2021-A_Systematic_Review_of_Data_Models_for_the_Big_Da	CLOUD-NATIVE ARCHITECTURE > Scalability	NoSQL databases, in particular the column-store Cassan-dra database were used for storing linked data and OWL ontology [44]. Given that linked data storage necessitate scal-able databases and distributed parallel algorithms, a schema was designed to store the ontology classification in the Cassandra database and help OWL scalability. Combining MapReduce algorithms on Cassandra, an optimal model was implemented for the	Ivon Miranda Santos
Mostajabi2021-A_Systematic_Review_of_Data_Models_for_the_Big_Da	CLOUD-NATIVE ARCHITECTURE > Scalability	The graph data models have practical application in social networks concerning their rich and com-plex search demand. This poses the challenges of scalability which limit query performance by using the algorithms such as neighbor search or k-nearest neighbor. The distributed and encrypted graph was introduced to overcome this limitation. The data model efficiency, operational power, scalability, memory consumption, and search time latency were studied on YouTube data. In [46], a graph-based metamodel, GSMM, was proposed on a straight line and tagged graph for structured and unstruc-tured data accompanied by	Ivon Miranda Santos
Mostajabi2021-A_Systematic_Review_of_Data_Models_for_the_Big_Da	CLOUD-NATIVE ARCHITECTURE > Scalability	In this architecture, there were defined modules for query execution and presentation. The model data scalability and performance were investigated in this research. Although experiments showed satisfactory results, this model was not cost-effective for key-value databases in terms of time	Ivon Miranda Santos
Mostajabi2021-A_Systematic_Review_of_Data_Models_for_the_Big_Da	CLOUD-NATIVE ARCHITECTURE > Flexibility	To transform a logical model into a conceptual model, a process was designed for converting relation-entity data model to a graph-based data model in NoSQL [38]. The process was intended to increase flexibility of the concep-tual models. Another NoSQL abstract model (NoAM) was presented based on the common features to	Ivon Miranda Santos
Mostajabi2021-A_Systematic_Review_of_Data_Models_for_the_Big_Da	CLOUD-NATIVE ARCHITECTURE > Flexibility	The XML model was used at the logical level in addition to a graph-based data model. Also, several charac-teristics have been introduced for flexibility in the conceptual data schema. Ultimately, an algorithm was developed for the conceptual model transformation on a set of integrated con-straints by which EB-ER that was automatically mapped into a graph-based data	Ivon Miranda Santos

Mostajabi2021-A_Systematic_Review_of_Data_Models_for_the_Big_Da	CLOUD-NATIVE ARCHITECTURE > Scalability	X cloud class, including the cloud-based columnar database of 1010data System, Azure DocumentDB, Amazon DynamoDB, and Datameer were also investigated. In addition to basic features of the cloud itself, key features of the cloud environment, including scalability, performance, and config-uration were finally introduced for investigation on big data storage.	Ivon Miranda Santos
Mostajabi2021-A_Systematic_Review_of_Data_Models_for_the_Big_Da	CLOUD-NATIVE ARCHITECTURE > Scalability	The hierarchical data model called NewSQL was suggested combining goals of RDB and NoSQL deliver. It aimed to cover horizontal scalability similar to NoSQL and maintain ACID properties similar to the relational model. NoSQL was proposed as a schema-less data model with respect to the stored data volume in the cloud	Ivon Miranda Santos
Mostajabi2021-A_Systematic_Review_of_Data_Models_for_the_Big_Da	CLOUD-NATIVE ARCHITECTURE > Scalability	More-over, the data model and the vector data model within the database were considered necessary for a new spatial data model. Having reviewed index and query, this paper found the features of data partition and scalability required for the data storage. In [51], a hybrid data model was developed from	Ivon Miranda Santos
Mostajabi2021-A_Systematic_Review_of_Data_Models_for_the_Big_Da	CLOUD-NATIVE ARCHITECTURE > Flexibility	MySQL and RDF Store were used for implementation of the model. The flexibility, mobility, and efficiency were the considered features in this model. In [8], Another model was designed for transferring and mapping spatial data from RDB schema to	Ivon Miranda Santos
Mostajabi2021-A_Systematic_Review_of_Data_Models_for_the_Big_Da	CLOUD-NATIVE ARCHITECTURE > Flexibility	To show data sources in various NoSQL data models, an integrated data model is provided and then utilized to test accessibility as a key feature in data management systems. The flexibility of this model makes it applicable to complex systems with several heterogeneous NoSQL databases. This model can be applied to key-value, document-	Ivon Miranda Santos
Mostajabi2021-A_Systematic_Review_of_Data_Models_for_the_Big_Da	CLOUD-NATIVE ARCHITECTURE > Scalability	The distinguishing feature of this model is its agile management in adding new objects and properties to new JSON data with a new program code and no schema change. The distinction between the relational model and this model is horizontal scalability, independence scaling, and clustering without shared storage. In [35], the schema of the MongoDB data model was studied for schema inference from the document-oriented data models in NoSQL, and	Ivon Miranda Santos
Mostajabi2021-A_Systematic_Review_of_Data_Models_for_the_Big_Da	CLOUD-NATIVE ARCHITECTURE > Scalability	generalized to the generalization relationship model in which clusters were represented as a graph. The analysis time, development time, rest time, and total aggregation time are among the studied properties, whereas flexibility, scalability, and efficiency are not among key properties considered in the design of data model in this study. In [55] A schema was suggested for the document-based database in three areas of NoSQL, namely architecture, algo-rithm, and association rules. The purpose of this model was to maintain a balance of compatibility, availability, and scal-ability features in the data model. The architecture consisted of four phases, the representation layer on the client side, the selection of selectable and	Ivon Miranda Santos
Mostajabi2021-A_Systematic_Review_of_Data_Models_for_the_Big_Da	CLOUD-NATIVE ARCHITECTURE > Flexibility	generalized to the generalization relationship model in which clusters were represented as a graph. The analysis time, development time, rest time, and total aggregation time are among the studied properties, whereas flexibility, scalability, and efficiency are not among key properties considered in the design of data model in this study. In [55] A schema was suggested for the document-	Ivon Miranda Santos
Mostajabi2021-A_Systematic_Review_of_Data_Models_for_the_Big_Da	CLOUD-NATIVE ARCHITECTURE > Scalability	In [25], a schema was developed for document-based data models such as MongoDB to support and store temporal data from sensors that continuously transmit data. With the aim of scalability of data integration in addition to flex-ibility evolution, this schema examined the challenges of temporary data modeling. Also, an	Ivon Miranda Santos

Mostajabi2021-A_Systematic_Review_of_Data_Models_for_the_Big_Da	CLOUD-NATIVE ARCHITECTURE > Flexibility	In [25], a schema was developed for document-based data models such as MongoDB to support and store temporal data from sensors that continuously transmit data. With the aim of scalability of data integration in addition to flex-ibility evolution, this schema examined the challenges of temporary data modeling. Also, an	Ivon Miranda Santos
Mostajabi2021-A_Systematic_Review_of_Data_Models_for_the_Big_Da	CLOUD-NATIVE ARCHITECTURE > Scalability	The metadata was divided into the operational and the archive categories, part of which was sent to SQL and part needed for data analysis was sent to this new data model in NoSQL. The purpose of this data model was to improve performance and scalability in analysis and dis-tributed systems. V. RESULTS AND COMPARISON	Ivon Miranda Santos
Mostajabi2021-A_Systematic_Review_of_Data_Models_for_the_Big_Da	CLOUD-NATIVE ARCHITECTURE > Scalability	The key-value model lists all data in ascending order, and the values are accessible only by the key [43], [62]. In modern types of this category, scalability has priority over compatibility. This system is useful when there is only one type of object for which query is done based on a type of	Ivon Miranda Santos
Mostajabi2021-A_Systematic_Review_of_Data_Models_for_the_Big_Da	CLOUD-NATIVE ARCHITECTURE > Scalability	It stores data in a structured and hierarchical manner and with accessibility to different IDs [62]. (c) Column store This model, also known as Wide Column Store, structures the data into columns with the number of key-value pairs [43], and thus provides high scalability [62]. Extended records can be vertically or horizontally partitioned across nodes. In this category, the data model is in the form of rows and columns, and the scalability model is obtained from splitting both rows and columns over multiple	Ivon Miranda Santos
Mostajabi2021-A_Systematic_Review_of_Data_Models_for_the_Big_Da	CLOUD-NATIVE ARCHITECTURE > Scalability	The XML emerged as a response to the weaknesses of relational and object-relational data models. These models have structures with poor scal-ability in which system performance is reduced with increasing size. Nonetheless, there was no response to the unstructured data.	Ivon Miranda Santos
Mostajabi2021-A_Systematic_Review_of_Data_Models_for_the_Big_Da	CLOUD-NATIVE ARCHITECTURE > Scalability	Overall, NoSQL databases hold the first rank in the world of database technology with respect to the cumulative frequency of their several groups. On the one hand, features such as high scalability (horizontal), avail-ability, and schema flexibility provide these databases with the possibility to overcome big data challenges and the suit-ability for data streams. On the other hand, deficiencies such as static schema and limited types of predefined data to define relationships face	Ivon Miranda Santos
Mostajabi2021-A_Systematic_Review_of_Data_Models_for_the_Big_Da	CLOUD-NATIVE ARCHITECTURE > Flexibility	Overall, NoSQL databases hold the first rank in the world of database technology with respect to the cumulative frequency of their several groups. On the one hand, features such as high scalability (horizontal), avail-ability, and schema flexibility provide these databases with the possibility to overcome big data challenges and the suit-ability for data streams. On the other hand, deficiencies such as static schema and limited types of predefined data to define relationships face	Ivon Miranda Santos
Mostajabi2021-A_Systematic_Review_of_Data_Models_for_the_Big_Da	CLOUD-NATIVE ARCHITECTURE > Flexibility	It is obvious that all of these features cannot be simultaneously gathered in a single model. For instance, agility and flexibility were obtained at the expense of low compatibility of data storage and retrieval with the database in the data model [35]. The scalability was increased with the decrease of ACID properties in NoSQL databases [19].	Ivon Miranda Santos
Mostajabi2021-A_Systematic_Review_of_Data_Models_for_the_Big_Da	CLOUD-NATIVE ARCHITECTURE > Scalability	For instance, agility and flexibility were obtained at the expense of low compatibility of data storage and retrieval with the database in the data model [35]. The scalability was increased with the decrease of ACID properties in NoSQL databases [19]. However, the more coverage of these features, the more capable is the data model	Ivon Miranda Santos
Mostajabi2021-A_Systematic_Review_of_Data_Models_for_the_Big_Da	CLOUD-NATIVE ARCHITECTURE > Scalability	Systematic Review of Data Models for Big Data Problem • Scalability: The data model efficient performance	Ivon Miranda Santos
Mostajabi2021-A_Systematic_Review_of_Data_Models_for_the_Big_Da	CLOUD-NATIVE ARCHITECTURE > Flexibility	The new model compatibility with previ-ous data models for data integration. • Flexibility: The data model ability to provide support for	Ivon Miranda Santos

Mostajabi2021-A_Systematic_Review_of_Data_Models_for_the_Big_Da	CLOUD-NATIVE ARCHITECTURE > Scalability	Data should be available whenever necessary. The determining factors in data model design are consid-ered consistency, availability, and scalability [55]. Having reviewed various articles, we observed that scalability, per-formance, and schema-less with frequencies of 10% or more are important features in designing the data model (Figure 8). In this diagram, frequencies of different features are close to each other, suggestive of the	Ivon Miranda Santos
Mostajabi2021-A_Systematic_Review_of_Data_Models_for_the_Big_Da	CLOUD-NATIVE ARCHITECTURE > Scalability	In other words, a data model is more practi-cal when includes simultaneous existence of these features. The NoSQL databases have been introduced as a comple-ment to the previous ones, especially relational databases to remedy deficiencies of scalability, flexibility at the schema level, ability to store large volumes of data, and	Ivon Miranda Santos
Mostajabi2021-A_Systematic_Review_of_Data_Models_for_the_Big_Da	CLOUD-NATIVE ARCHITECTURE > Flexibility	In other words, a data model is more practi-cal when includes simultaneous existence of these features. The NoSQL databases have been introduced as a comple-ment to the previous ones, especially relational databases to remedy deficiencies of scalability, flexibility at the schema level, ability to store large volumes of data, and	Ivon Miranda Santos
Naik2021-Performance_evaluation_of_distributed_systems_in_multi	CLOUD-NATIVE ARCHITECTURE > Scalability	This paper initially presents the simulated development of a Docker Swarm-based distributed system which can be easily replicated in multiple clouds. Subsequently, based on the simulated Docker Swarm-based distributed system, it performs an evaluation of several attributes of this distributed system such as high availability and fault tolerance; automatic scalability, load balancing and maintainability of services; and scalability of large clusters. Keywords—Distributed System; Docker Swarm; Multiple Clouds; Container; Containerization; Virtual Machine; Virtualiza-tion; High Availability; Fault Tolerance; Scalability; Maintainabil-ity; Load	Ivon Miranda Santos
Naik2021-Performance_evaluation_of_distributed_systems_in_multi	CLOUD-NATIVE ARCHITECTURE > Scalability	Subsequently, it presents the simulated development of a Docker Swarm-based distributed system in VirtualBox, which can be easily replicated in multiple clouds. Finally, based on the simu-lated Docker Swarm-based distributed system, it performs an evaluation of several attributes of the Docker Swarm-based distributed system such as high availability and fault tolerance; automatic scalability, load balancing and maintainability of services; and scalability of large clusters. This evaluation demonstrates that the Docker Swarm-based distributed system is relatively easy to design and act as a natural	Ivon Miranda Santos
Naik2021-Performance_evaluation_of_distributed_systems_in_multi	CLOUD-NATIVE ARCHITECTURE > Scalability	It is an enhancement of Docker container technology for creating a cluster of nodes across several machines or clouds and designing distributed systems in multiple clouds. Docker Swarm offers several essential attributes of a distributed system to the Swarm cluster such as availability, reliability, fault tolerance, maintainability and scalability, which is an added advantage to the basic container	Ivon Miranda Santos
Naik2021-Performance_evaluation_of_distributed_systems_in_multi	CLOUD-NATIVE ARCHITECTURE > Scalability	The time period to report the health of a worker is decided by the leader/manager. B. Automatic Scalability, Load Balancing and Maintainability of Services A Docker Swarm-based distributed system also offers automatic scalability, load balancing and maintainability of services. For evaluating these attributes, two services nginx and redis are created on the nodes of the Docker Swarm cluster, which are called nginx-server1 and redis-server2	Ivon Miranda Santos

Naik2021-Performance_evaluation_of_distributed_systems_in_multi	CLOUD-NATIVE ARCHITECTURE > Scalability	Running 3 hours ago This experiment demonstrates that the Docker Swarm-based distributed system provides automatic scalability, load balancing and maintainability of services. C. Scalability of Large Clusters The several recent research studies [16], [17], [18], [19] have been conducted to test and compare the scalability performance of Docker Swarm with other Containers, where Docker Swarm and Google Kubernetes were tested and com-pared based on the large cluster size. Initially, the scalability performance test was carried out by both organizations Docker [16] and Google Kubernetes [17] for the large cluster size. Subsequently, a Docker-sponsored study for the comparison of the scalability performance of Docker Swarm and Google Kubernetes was performed by an independent technology consultant Jeff Nickoloff [18], [19]. He designed a Cloud Container Cluster Common Benchmark framework (available on GitHub [20]) to test the performance of both container platforms while running 30,000 containers across 1,000 node in a cluster. This automated test framework mainly compared	Ivon Miranda Santos
Naik2021-Performance_evaluation_of_distributed_systems_in_multi	CLOUD-NATIVE ARCHITECTURE > Scalability	It illustrated the design and simulated development of the distributed system in multiple clouds using Docker Swarm. Subsequently, the paper demonstrated an evaluation of several attributes of the distributed system such as high availability and fault tolerance; automatic scalability, load balancing and maintainability of services; and scalability of large clusters. This evaluation demonstrated that the Docker Swarm-	Ivon Miranda Santos
Opara-Martins2016-Critical_analysis_of_vendor_lock-in_and_its_i	CLOUD-NATIVE ARCHITECTURE > Flexibility	Advances, Systems and Applications (2016) 5:4 Page 2 of 18from the view point of the business to retain the flexibil-ity to change providers according to business concerns or even keep in-house some of the components that are less mission-critical due to security related risks. Inter-operability and portability among cloud providers	Ivon Miranda Santos
Opara-Martins2016-Critical_analysis_of_vendor_lock-in_and_its_i	CLOUD-NATIVE ARCHITECTURE > Scalability	Adoption of cloud computing by UK businesses The survey affirms that the concept of using cloud com-puting services to address the business IT needs has established a mainstream deployment across organisa-tions of various sizes. To further substantiate this matter, interestingly about 36 % of participants confirmed using a hybrid (public and private) cloud deployment model as opposed to a private cloud. Only 46 % of UK firms par-ticipated in the survey use public cloud services, in spite of the associated security risks (Fig. 4). The rate of adop-tion has been motivated by numerous indicators for effective cloud deployment decision. The most cited rea-sons for adopting cloud computing includes better scal-ability of IT resources (45.9 %), collaboration (40.5 %), cost savings (39.6 %) and increased flexibility (36.9 %). This suggests that organisations are allured to	Ivon Miranda Santos
Opara-Martins2016-Critical_analysis_of_vendor_lock-in_and_its_i	CLOUD-NATIVE ARCHITECTURE > Flexibility	Adoption of cloud computing by UK businesses The survey affirms that the concept of using cloud com-puting services to address the business IT needs has established a mainstream deployment across organisa-tions of various sizes. To further substantiate this matter, interestingly about 36 % of participants confirmed using a hybrid (public and private) cloud deployment model as opposed to a private cloud. Only 46 % of UK firms par-ticipated in the survey use public cloud services, in spite of the associated security risks (Fig. 4). The rate of adop-tion has been motivated by numerous indicators for effective cloud deployment decision. The most cited rea-sons for adopting cloud computing includes better scal-ability of IT resources (45.9 %), collaboration (40.5 %), cost savings (39.6 %) and increased flexibility (36.9 %). This suggests that organisations are allured to	Ivon Miranda Santos
Opara-Martins2016-Critical_analysis_of_vendor_lock-in_and_its_i	CLOUD-NATIVE ARCHITECTURE > Scalability	As shown in Fig. 5, the majority of the respondents identi-fied capacity and scalability (70.3 %), increased collabor-ation, availability, geography and mobility as benefits for migration. However, further analysis have shown, from a	Ivon Miranda Santos

Opara-Martins2016-Critical_analysis_of_vendor_lock-in_and_its_i	CLOUD-NATIVE ARCHITECTURE > Flexibility	However, for smaller enterprises, the adoption of (non-mission critical) cloud-based appli-cations mirrors their use of email messaging, desktop hosting and Customer Relationship Management (CRM) applications for collaboration. Remarkably, the lower cost and flexibility that cloud-based applications offer is ideal for small businesses, as they are agile and often run with teams that are spread over wide geographical re-gions. In essence, these applications are better suited for	Ivon Miranda Santos
Perrons2013-Cloud_computing_in_the_upstream_oil_and_gas_industr	CLOUD-NATIVE ARCHITECTURE > Flexibility	For example, insurance companies are seeking more flexible cost structures, and the cloud allows firms to shift costs to an operating expense rather than a capital outlay, thereby giving them a highly flexible ‘‘pay as you go’’ resource. Deployment flexibility: In the media industry, for instance, the type of content that consumers request can suddenly become viral and then immediately fall out of fashion.	Ivon Miranda Santos
Perrons2013-Cloud_computing_in_the_upstream_oil_and_gas_industr	CLOUD-NATIVE ARCHITECTURE > Flexibility	Demand patterns can therefore swing profoundly from one extreme to the next. A cloud-based infrastructure affords companies in these industries a high degree of flexibility with regards to the amount of computing and data storage resources that they need at any moment in time.	Ivon Miranda Santos
Perrons2013-Cloud_computing_in_the_upstream_oil_and_gas_industr	CLOUD-NATIVE ARCHITECTURE > Scalability	Among the more useful features of IaaS is the fact that customers pay for functionality by the hour, making this an attractive option for customers who want to maintain some semblance of ‘‘business as unusual’’—that is, situations in which the customer is simply filling in a short-term hardware deficiency. Customers might also be attracted to this model because of its near-infinite scalability and the fact that customers are not directly responsible for the management of the hardware used to provide the service. A customer would	Ivon Miranda Santos
Perrons2013-Cloud_computing_in_the_upstream_oil_and_gas_industr	CLOUD-NATIVE ARCHITECTURE > Scalability	, 2009). The attrac-tiveness of this concept lies in the fact that users can enjoy near-infinite scalability and very high system reliability. What is more, because of their highly virtual nature, these resources can be procured competitively from a broad array of specialized vendors and IT	Ivon Miranda Santos
Perrons2013-Cloud_computing_in_the_upstream_oil_and_gas_industr	CLOUD-NATIVE ARCHITECTURE > Scalability	, 2009, p. 15). In this way, hybrid clouds offer the best of both worlds insofar as this approach makes it possible to manage security-related threats carefully while creating a secure conduit through which customers can selectively leverage the scalability of the public cloud whenever and however they want. Hybrid cloud solutions are a clever way to reap many of the benefits of the public cloud while maintaining a higher degree of control over data security, and they are therefore a very useful bridging technology that customers can use to move towards the public cloud while still hanging on to legacy systems or until software vendors can come up with cloud-friendly alternatives. But hybrid systems do come at a cost: they do not offer the near-infinite scalability, extremely high ‘‘outsourceability,’’ and cost efficiency that the totally public cloud does. It therefore follows that these mid-ground solutions do address some of	Ivon Miranda Santos
Perrons2013-Cloud_computing_in_the_upstream_oil_and_gas_industr	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Although private and hybrid cloud architectures are popular within the industry at the moment because of existing constraints, the examples pre-sented in this paper point to a future that is increasingly predicated on the public cloud.	Ivon Miranda Santos
Perrons2013-Cloud_computing_in_the_upstream_oil_and_gas_industr	CLOUD-NATIVE ARCHITECTURE > Flexibility	Although private and hybrid cloud architectures are popular within the industry at the moment because of existing constraints, the examples pre-sented in this paper point to a future that is increasingly predicated on the public cloud. We accordingly believe that companies within the upstream oil & gas industry—including international oil companies, national oil companies, service com-panies, and vendors—would be well advised to build into their systems enough flexibility and modularity to make this change when the time is right, thereby allowing them to take full advantage of the benefits that cloud computing can offer. We also believe that this evidence demonstrates	Ivon Miranda Santos

Petcu2014-Portability_in_clouds_Approaches_and_research_opportu	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Porting Cloud native applications, data or services: even when applications, data or services are written from scratch in a Cloud environment, they are usually locked and targeted for a specific Cloud, and the effort for porting in a different	Ivon Miranda Santos
Petcu2014-Portability_in_clouds_Approaches_and_research_opportu	CLOUD-NATIVE ARCHITECTURE > Coupling	PaaS openness degree of the source programming languages for application development, openness degree of the data formats, coupling degree of the coupled services, platform-dependence degree of the abstraction	Ivon Miranda Santos
Petcu2014-Portability_in_clouds_Approaches_and_research_opportu	CLOUD-NATIVE ARCHITECTURE > Coupling	Cloud portability is currently achieved through open standards, protocols, widely used APIs or through abstraction layers which decouple application development from specific target	Ivon Miranda Santos
Petcu2014-Portability_in_clouds_Approaches_and_research_opportu	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	A reference architecture for Cloud services that describes key components, such as actors, interfaces, data artifacts, profiles and the interrelationships among these components.	Ivon Miranda Santos
Petcu2014-Portability_in_clouds_Approaches_and_research_opportu	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	NIST' spe-cial publications are referring to Cloud architectures, Cloud security, Cloud deployment in the context of various strategies of USA federal government OASIS www.oasis-open.org/, Three important	Ivon Miranda Santos
Petcu2014-Portability_in_clouds_Approaches_and_research_opportu	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Cloud Storage Technical Work Group developing an architecture related to system implementations of Cloud storage technology.	Ivon Miranda Santos
Petcu2014-Portability_in_clouds_Approaches_an	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	The applications can follow an event-driven architecture [20].	Ivon Miranda Santos
Petcu2014-Portability_in_clouds_Approaches_and_research_opportu	CLOUD-NATIVE ARCHITECTURE > Coupling	A key differentiator for CSAL is that it provides a single namespace for each storage type (by this is able to decouple the application code from explicitly making calls to specific storage	Ivon Miranda Santos
Petcu2014-Portability_in_clouds_Approaches_and_research_opportu	CLOUD-NATIVE ARCHITECTURE > Coupling	4.6. Model-driven engineering. Model driven engineering (MDE) is based on two core ideas [5]: ab-straction and automation. Abstraction enables decoupling application development from targeting specific platforms. Automation refers, among others, to the ability to change the level of abstraction automatically, using model transformations (e.g. domain specific languages, shortly DSLs). The DSLs allows to the domain	Ivon Miranda Santos
Petcu2014-Portability_in_clouds_Approaches_and_research_opportu	CLOUD-NATIVE ARCHITECTURE > Coupling	– Offers an abstraction layer that is simple to use – Usually refers to the common denominator of the Cloud services – Available for major languages – Language dependent – Similarity with major Cloud provider APIs – Adaptors needs to be build for emerging new services – Decouple the application code from the un-derlying Cloud service – The connected service programming style usually maintained – Adaptors available for major Cloud services – Require Cloud computing knowledge as de-ployment is usually not supported – Introduces an overhead compared with the	Ivon Miranda Santos
Petcu2014-Portability_in_clouds_Approaches_and_research_opportu	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	System developers need to leverage standards to support the architecture of a system. However, such standards should not drive such an architecture [17]. The software architecture for Cloud-based systems in which standards-reliant components should be implemented as components that are separated from the rest of the system (in order to minimize the impact on the	Ivon Miranda Santos
Petcu2014-Portability_in_clouds_Approaches_and_research_opportu	CLOUD-NATIVE ARCHITECTURE > Flexibility	Therefore the management and service automation levels are the hot-spots of the development activities in the last three years. An experimented developer can see a lack of flexibility in using a uniform interface or a model driven approach. The primary concern is the limitation of the abstractions to the set of smallest	Ivon Miranda Santos
Petcu2014-Portability_in_clouds_Approaches_and_research_opportu	CLOUD-NATIVE ARCHITECTURE > Scalability	Most IaaS Clouds expose limited capabilities to control how a service behaves at run-time, beyond basic low-level scalability rules for VMs (once it has been deployed). Different Cloud services rely on specific rule engines to help enforcing the rules governing the service during its whole life-cycle.	Ivon Miranda Santos

Rai2015-Exploring_the_factors_influencing_the_cloud_computing_a	CLOUD-NATIVE ARCHITECTURE > Scalability	Today, most of the organizations trust on their age old legacy applications, to support their business-critical systems. However, there are several critical concerns, as maintainability and scalability issues, associated with the legacy system. In this background, cloud services offer a more agile and cost effective platform, to support business	Ivon Miranda Santos
Rai2015-Exploring_the_factors_influencing_the_cloud_computing_a	CLOUD-NATIVE ARCHITECTURE > Scalability	2013). For SMEs, the central attraction for adopting cloud technology is its pay-per-use model, which delivers flexible costing options, apart from the scalability and interoperability features, which cloud environments offer. Larger enter-prises are attempting to leverage this technology by considering the business continuity	Ivon Miranda Santos
Rai2015-Exploring_the_factors_influencing_the_cloud_computing_a	CLOUD-NATIVE ARCHITECTURE > Scalability	2012). The traditional legacy system, which supports the core IT processes at organizations, is fraught with maintainability and scalability issues, (Khadka et al. 2013).	Ivon Miranda Santos
Rai2015-Exploring_the_factors_influencing_th	CLOUD-NATIVE ARCHITECTURE > Scalability	Optimum resource utilization iii. Unlimited scalability of resources iv. Less maintainability	Ivon Miranda Santos
Ranchal2020-Disrupting_healthcare_silos_Addressing_data_volume_	CLOUD-NATIVE ARCHITECTURE > Scalability	I. INTRODUCTION MODERN IT systems are increasingly adopting cloud computing and moving their workloads on cloud to take advantage of its various benefits such as economies of scale, energy efficiency, scalability, and elasticity. At the same time, the applications that run on cloud require their data to be available	Ivon Miranda Santos
Ranchal2020-Disrupting_healthcare_silos_Addressing_data_volume_	CLOUD-NATIVE ARCHITECTURE > Flexibility	It enhances the interoperability by enabling the integration of medical imaging devices such as Picture Archiving and Communication System (PACS), Vendor Neutral Archives (VNAs), scanners, printers, and workstations from different manufacturers. While adopting such standards aids towards removing vagueness and enhances usability of the data, the healthcare standards inherently lack precision in their data models to allow flexibility and global adoption. Organizations that generate or consume healthcare data must either implement a	Ivon Miranda Santos
Ranchal2020-Disrupting_healthcare_silos_Addressing_data_volume_	CLOUD-NATIVE ARCHITECTURE > Coupling	These ca-pabilities help in the rapid development of new applications, cross-domain system integration, and coupling with existing healthcare	Ivon Miranda Santos
Ranchal2020-Disrupting_healthcare_silos_Addressing_data_volume_	CLOUD-NATIVE ARCHITECTURE > Scalability	(HIPAA CFR §164.306,164.308, GDPR) • Ability to scan the ingested data for viruses and malware that are kept up to date with latest definitions. (HIPAA CFR §164.306,164.308,164.312) • Offer scalability and high availability of the service to handle increase in requests and storage, and have resiliency to failures. (HIPAA CFR §164.308) • Offer backup and disaster recovery to support	Ivon Miranda Santos
Ranchal2020-Disrupting_healthcare_silos_Addressing_data_volume_	CLOUD-NATIVE ARCHITECTURE > Flexibility	These functions are accomplished using a modular design with each module focusing on a specific functionality. Such a design provides flexibility and enables extensibility. Fig.	Ivon Miranda Santos
Ranchal2020-Disrupting_healthcare_silos_Addressing_data_volume_	CLOUD-NATIVE ARCHITECTURE > Scalability	PDF The DB2 is configured in a master-slave setup, HDFS clusters are configured with a high replication factor, and HBase is configured with multiple region servers to ensure high avail-ability of data. More data nodes can be added to provide storage scalability. Replication of data across data centers leverages a Kafka service and MirrorMaker—raw data is wrapped by a Kafka producer into a message and written to the Kafka cluster, MirrorMaker replicates the message across Kafka clusters in two different data centers, and the Kafka consumer reads the	Ivon Miranda Santos
Ravi2019-Emergence_of_middleware_to_mitigate_the_challenges_of_	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	As of today, despite initiatives by NASA since 2013 for “open cloud architectures”, not many standards have emerged to support this notion because cloud vendors are not interested for such standards as they will lose competitiveness in the	Ivon Miranda Santos

Ravi2019-Emergence_of_middleware_to_mitigate_the_challenges_of_	CLOUD-NATIVE ARCHITECTURE > Scalability	7. Scalability Analysis The scalability analysis of FUSION platform was tested using an open source soft-ware, Tsung. There are two secondary nodes and one primary node deployed across a cluster and the results are plotted using the Tsung-plotter utility tool. The can-didate parameters for the scalability analysis are the load benchmarks, machine details, number of concurrent users, and the session aspects. The load bench-mark signifies the information on the mean inter-arrival time between	Ivon Miranda Santos
Ravi2019-Emergence_of_middleware_to_mitigate_the_challenges_of_	CLOUD-NATIVE ARCHITECTURE > Scalability	After which, the addition of more nodes does not necessarily improve the performance. However, the cloud bursting capacity from Eucalyptus to a public cloud such as AWS can help in achieving higher scalability more quickly. 8.	Ivon Miranda Santos
Ravi2019-Emergence_of_middleware_to_mitigate_the_challenges_of_	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Developing such a framework would certainly decrease the effort and complexity of developing a mobile application that requires accessing distributed hybrid cloud architectures.	Ivon Miranda Santos
Raza2019-A_review_on_security_issues_and_their_impact_on_hybrid	CLOUD-NATIVE ARCHITECTURE > Flexibility	Additionally, Google Cloud use 19%, Oracle Cloud use 9%, and RackSpace use 7.3%. Hybrid cloud computing is about aggregation and integration of computer, networking, applications, storage, security and management into unified, orchestrated management framework which enables enterprise IT and developers to leverage scale, flexibility and cost savings of existing in-house IT investment tools, systems and privacy policies scale to manage in the enterprise data center with their newly adopted cloud services[4], [5]. The IDC report predicts more than 80 per	Ivon Miranda Santos
Raza2019-A_review_on_security_issues_and_their_impact_on_hybrid	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	[13] focused on to identify the security threats in a hybrid cloud architectures for enterprises and suggested control method to access the data in Hybrid cloud approach using multi factor authentication from on-premises Active Directory.	Ivon Miranda Santos
Repschlaeger2012-Cloud_requirement_framework_Requirements_and_e	CLOUD-NATIVE ARCHITECTURE > Flexibility	, 2010). Companies expect to reduce their costs, to gain flexibility and an unlimited resource access (Mueller et al., 2011).	Ivon Miranda Santos
Repschlaeger2012-Cloud_requirement_framework_Requirements_and_e	CLOUD-NATIVE ARCHITECTURE > Flexibility	The framework consists of two parts, the Cloud Computing target dimensions and the Cloud requirements (see Figure 1). The target dimensions - such as cost savings or increasing flexibility – represent objectives which the customer pursues and may characterize its IT strategy or especially the related Cloud strategy. These dimensions cover the Cloud Computing in	Ivon Miranda Santos
Repschlaeger2012-Cloud_requirement_framework_Requirements_and_e	CLOUD-NATIVE ARCHITECTURE > Flexibility	The target dimension ”Scope & Performance” cover the functionality and performance of the Cloud service and consists of four abstract requirements: service characteristics, service optimizing, hardware, and performance. The dimension “Flexibility” describes the ability to respond quickly to changing capacity requirements and competition pressure. It is	Ivon Miranda Santos
Repschlaeger2012-Cloud_requirement_framework_Requirements_and_e	CLOUD-NATIVE ARCHITECTURE > Flexibility	Regarding to our research most of the requirements of the dimensions Costs, Reliability & Trustworthiness, IT Security & Compliance and Service & Cloud Management are independent of the service model. The target dimensions Flexibility and Scope & Performance consist mostly of abstract requirements and evaluation criteria specific to a service model. Furthermore, we divided the scope of the requirements into criteria associated with the provider or related	Ivon Miranda Santos
Repschlaeger2012-Cloud_requirement_framework_Requirements_and_e	CLOUD-NATIVE ARCHITECTURE > Scalability	Provider requirements describe the characteristics of the underlying infrastructure of a Cloud provider, for instance this can be supplier certifications, IT infrastructure features or data center locations. Service requirements, however, describe the service usage, the prices, the scalability or the number of interfaces. Independent of the Service Model Specific to	Ivon Miranda Santos

Repschlaeger2012-Cloud_requirement_framework_Requirements_and_e	CLOUD-NATIVE ARCHITECTURE > Flexibility	Availability and liability include the probability that service commitments and promises can be met by the provider, based on indicators like the service availability, accessibility to several internet service providers and the liability agreements including penalties if the guaranteed service level is not met. Flexibility Provisioning and set-up time are subsumed under the associated flexibility advantage of Cloud Computing. Resources, for instance, can be allocated and de-allocated as required.	Ivon Miranda Santos
Repschlaeger2012-Cloud_requirement_framework_Requirements_and_e	CLOUD-NATIVE ARCHITECTURE > Scalability	The provisioning time is shorter compared to traditional outsourcing and the set-up time to get in contact with the provider (e.g. register or set up a new account) is shorter as well. Interoperability and scalability comprise all features regarding the maximal number of available resources (e.g. user accounts, instances, functions, services) which can be used simultaneously. Additionally the interoperability describes the integration degree separated into internal communication (between services of the provider) and external communication (between services of different providers). The browser compatibility is important	Ivon Miranda Santos
Repschlaeger2012-Cloud_requirement_framework_Requirements_and_e	CLOUD-NATIVE ARCHITECTURE > Flexibility	supported programming languages) is of high relevance on the PaaS level. Contract flexibility and renewal of contract both represent the commitment between the customer and the provider (e.g. contract length) and defined	Ivon Miranda Santos
Repschlaeger2012-Cloud_requirement_framework_Requirements_and_e	CLOUD-NATIVE ARCHITECTURE > Flexibility	These resource guarantees are influenced by the internal IT infrastructure, external partners, suppliers and the amount of users. Flexibility Service and data portability contain the aspects relevant for the service and data mobility.	Ivon Miranda Santos
Rosati2018-Making_the_cloud_work_for_software	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Cloud Migration, Total Cost of Ownership, Monetization, Architecture Migration, Software	Ivon Miranda Santos
Rosati2018-Making_the_cloud_work_for_software_producers_Linking	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Cloud-native architectures have technical advantages in terms of isolation and reusability, thus reducing cost for maintenance and	Ivon Miranda Santos
Rosati2018-Making_the_cloud_work_for_software_producers_Linking	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Cloud-native architectures have technical advantages in terms of isolation and reusability, thus reducing cost for maintenance and	Ivon Miranda Santos
Rosati2018-Making_the_cloud_work_for_software_producers_Linking	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	PaaS clouds with their recent support for containerized microservice architectures are the ideal environments to create cloud-native	Ivon Miranda Santos
Rosati2018-Making_the_cloud_work_for_software_producers_Linking	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	, 2013; Pahl and Xiong, 2013), research exploring the link between cloud architecture and TCO, and therefore on pricing cloud services from an SP	Ivon Miranda Santos
Rosati2018-Making_the_cloud_work_for_software_producers_Linking	CLOUD-NATIVE ARCHITECTURE > Scalability	Right-scaling: conduct a feasibility study to validate quality requirements such as scalability; Right-pricing: determine pricing for the software service based on analysis of direct operational costs driven by predicted usage and experimental consumption figures generated from the	Ivon Miranda Santos
Rosati2018-Making_the_cloud_work_for_software_producers_Linking	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Cloud architecture qualities, and corresponding costs, can be influenced by compute, storage and network resources.	Ivon Miranda Santos
Rosati2018-Making_the_cloud_work_for_software_producers_Linking	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Estimation complexity varies between the two business cases identified earlier, i.e., migrated or cloud-native application.	Ivon Miranda Santos
Rosati2018-Making_the_cloud_work_for_software_producers_Linking	CLOUD-NATIVE ARCHITECTURE > Scalability	From an SP perspective, the selection criteria of a cloud provider include fees and billing model. Many IaaS providers offer monthly basic subscription fees with additional fees for premium services such as scalability, access (e.g., IP endpoint, network bandwidth) or monitoring and	Ivon Miranda Santos
Rosati2018-Making_the_cloud_work_for_software_producers_Linking	CLOUD-NATIVE ARCHITECTURE > Scalability	Another factor impacting resource requirement is the nature of the architecture. Stateless, loosely-coupled architectures help accommodate extra demand and enable scalability by just using additional resources on-demand without much start-up costs (transfer of state to other	Ivon Miranda Santos
Rosati2018-Making_the_cloud_work_for_software	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	We assume here a migration to a PaaS architecture to be cloud-native in style, i.e.,	Ivon Miranda Santos
Rosati2018-Making_the_cloud_work_for_software	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	We assume here a migration to a PaaS architecture to be cloud-native in style, i.e.,	Ivon Miranda Santos
Rosati2018-Making_the_cloud_work_for_software	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	This will also further clarify the impact of cloud software architecture on costs and revenues.	Ivon Miranda Santos
Rosati2018-Making_the_cloud_work_for_software_producers_Linking	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	The ultimate objective would be to move from VMs to so-called cloud-native applications at the platform level that utilize fully cloud-based services for development and deployment.	Ivon Miranda Santos

Rosati2018-Making_the_cloud_work_for_software_producers_Linking	CLOUD-NATIVE ARCHITECTURE > Flexibility	Its application has over 1,000 existing client installs and in this case study, we present the TCO estimation of migrating 240 of these to the new cloud platform over a 3-year period. The main business requirements for the SP to adopt the cloud were (i) to pursue flexibility across different devices and situational contexts, and (ii) to increase the customer base through new market entries. The solution requires meeting	Ivon Miranda Santos
Rosati2018-Making_the_cloud_work_for_software_producers_Linking	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Business Analysis, Cloud Architecture Design, Data Design, Security Framework Design, Development and Test, Performance and Costs	Ivon Miranda Santos
Rosati2018-Making_the_cloud_work_for_software_producers_Linking	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	A cloud compute architecture – made up of a separate compute resource for the web server of the web application (Web Role Virtual Machine), and a separate compute component for carrying out the image processing functions, such as barcode reading (Worker Role Virtual Machine).	Ivon Miranda Santos
Rosati2018-Making_the_cloud_work_for_software_producers_Linking	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Our literature review did not identify a detailed framework that integrated both costing and software architecture within a cloud migration	Ivon Miranda Santos
Rosati2018-Making_the_cloud_work_for_software_producers_Linking	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	From an architecture perspective, container technology and micro-service style architectures are an increasing feature in the enterprise cloud and are impacting cloud-native architectures.	Ivon Miranda Santos
Rosati2018-Making_the_cloud_work_for_software_producers_Linking	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	From an architecture perspective, container technology and micro-service style architectures are an increasing feature in the enterprise cloud and are impacting cloud-native architectures.	Ivon Miranda Santos
Saif2022-CSO-ILB_chicken_swarm_optimized_inter-	CLOUD-NATIVE ARCHITECTURE > Coupling	[40] introduced a VM coupling approach with scaling properties to enhance inter-cloud load	Ivon Miranda Santos
Saif2022-CSO-ILB_chicken_swarm_optimized_inter-cloud_load_balan	CLOUD-NATIVE ARCHITECTURE > Scalability	The Cat Swarm Optimization (CSO) [51] has attempted to select appropriate services for the workloads in the cloud. Still, the scalability of the resources is not main-tained, which might lead to scarcity and wastage of resources. To overcome these research gaps, an autonomic inter-cloud load balancer with self-management capabilities is proposed in this article to address the provisioning and load bal-ancing problems in a	Ivon Miranda Santos
Shastry2022-Approaches_for_migrating_non_cloud-native_applicati	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Abstract— This paper explores different strategies and tools for migrating non cloud-native or legacy applications to cloud.	Ivon Miranda Santos
Shastry2022-Approaches_for_migrating_non_cloud-native_applicati	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	The approach taken involves studying the existing migration strategies and experimenting them on different types of non cloud-native applications to identify the strengths and challenges associated	Ivon Miranda Santos
Shastry2022-Approaches_for_migrating_non_clou	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Applications being built today are 'cloud native', or demonstrate qualities that are indigenous to cloud.	Ivon Miranda Santos
Shastry2022-Approaches_for_migrating_non_cloud-native_applicati	CLOUD-NATIVE ARCHITECTURE > Scalability	Applications being built today are 'cloud native', or demonstrate qualities that are indigenous to cloud. These applications provide significant performance, maintainability, and scalability. This is a significant driver for migrating legacy applications to the cloud, to exercise these	Ivon Miranda Santos
Shastry2022-Approaches_for_migrating_non_cloud-native_applicati	CLOUD-NATIVE ARCHITECTURE > Scalability	It is also called the 'lift-and-shift' process. This is one of the fastest solutions for migration but has limited scalability, as the true benefits of cloud cannot be reaped. Moreover, most legacy applications are specifically designed for local usage and are not equipped to handle distributed workloads.	Ivon Miranda Santos
Shastry2022-Approaches_for_migrating_non_clou	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	In this technique, the basic architecture of application stays the same.	Ivon Miranda Santos
Shastry2022-Approaches_for_migrating_non_cloud-native_applicati	CLOUD-NATIVE ARCHITECTURE > Flexibility	The fundamental driver for choosing the proposed approach was the conversion of the monolith to microservices which would lead to loosely coupled and independently deployable smaller components, or services. The microservices approach has also been found to be more favorable than other architectures such as the Service Oriented Architecture as it provides more flexibility and is granular. Microservices which have been found to be suitable for development of	Ivon Miranda Santos
Shastry2022-Approaches_for_migrating_non_cloud-native_applicati	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	B. Re-architecting Online Shopping system In order to re-architecture the application, the first step was to identify the different components that could be potentially isolated.	Ivon Miranda Santos
Shastry2022-Approaches_for_migrating_non_clou	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	The application is therefore deployed on cloud and has a microservice architecture, hence	Ivon Miranda Santos

Shastry2022-Approaches_for_migrating_non_cloud-native_applicati	CLOUD-NATIVE ARCHITECTURE > Scalability	The application is therefore deployed on cloud and has a microservice architecture, hence making it cloud native. This gives the added benefit of scalability, availability and fault isolation. VIII.	Ivon Miranda Santos
Shastry2022-Approaches_for_migrating_non_clou	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	The application is therefore deployed on cloud and has a microservice architecture, hence	Ivon Miranda Santos
Shastry2022-Approaches_for_migrating_non_cloud-native_applicati	CLOUD-NATIVE ARCHITECTURE > Scalability	Its functionalities were maintained post rehosting, and its accessibility and performance was satisfactory. It improved in terms of database scalability after replatforming. The application was not re-architected. Table III illustrates a comparison of the three migration strategies on Django CMS where metrics such as performance, scalability, throughput etc were assessed. Performance is an aggregate metric that combines first contentful paint(10%), speed index(10%), time to	Ivon Miranda Santos
Shastry2022-Approaches_for_migrating_non_cloud-native_applicati	CLOUD-NATIVE ARCHITECTURE > Scalability	Though the application maintained its functionalities post rehosting, it was difficult to rehost in comparison to the CMS. It was however observed that the database's scalability considerably increased after replatforming. On re-architecting, the application was converted from a monolithic to a microservice architecture. This significantly improved the application's scalability, given that it could successfully be called cloud native. A comparison of the three	Ivon Miranda Santos
Shastry2022-Approaches_for_migrating_non_clou	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	On re-architecting, the application was converted from a monolithic to a microservice architecture.	Ivon Miranda Santos
Shastry2022-Approaches_for_migrating_non_cloud-native_applicati	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	NET similar to the ones experimented here can be rehosted without complications Applications having a more traditional architecture - that are not built with web frameworks but have a web-compatible UI (such as those built with HTML and PHP with separate servers), will need basic configurational changes.	Ivon Miranda Santos
Shastry2022-Approaches_for_migrating_non_cloud-native_applicati	CLOUD-NATIVE ARCHITECTURE > Scalability	Applications with the traditional model using Xampp-like servers which need Databases to be created and separately populated also require similar levels of effort. Database hosting helps improve scalability and boosts fault tolerance by making the data more resilient.	Ivon Miranda Santos
Shastry2022-Approaches_for_migrating_non_cloud-native_applicati	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Applications that follow a monolithic architecture, but are web-compatible can be similarly re-architected using a microservices architecture.	Ivon Miranda Santos
Shastry2022-Approaches_for_migrating_non_cloud-native_applicati	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	Applications that follow a monolithic architecture, but are web-compatible can be similarly re-architected using a microservices architecture.	Ivon Miranda Santos
Shirvani2018-An_iterative_mathematical_decision_model_for_cloud	CLOUD-NATIVE ARCHITECTURE > Flexibility	Furthermore, cloud services are provisioned via network links; it is error prone.27 Hence, in case of inevitable cloud service failure, which is worse than traditional IT, multicloud or intercloud solutions are suitable and promising. Although the speed of data communication between single-cloud modules is higher than that of mul-ticloud options, in case of failure and cybersecurity attacks, deploying multicloud and finding suitable alternatives are effective and reliable tasks due to the automated and quick reconfiguration between clouds without user intervention.27 Moreover, MCE offers low cost, better quality of service (QoS), flexibility, vendor lock-in avoidance, and reliability.27 For instance, DepSky28,29 and multicloud database30 apply data encryption and replication techniques on several datacenters, related to multiple providers in the IaaS level, to create a fault-tolerant system against failure. The Amazon EC2 service level agreement (SLA)31	Ivon Miranda Santos

Shirvani2018-An_iterative_mathematical_decision_model_for_cloud	CLOUD-NATIVE ARCHITECTURE > Flexibility	Integration of services is automatically applied on-demand without human intervention with a standard application programming interface (API)33-35; whereas, in case of failure, finding alterna-tives and system redeployment and reconfiguration will happen quickly.27 Furthermore, a single cloud infrastructure is not often reliable because not only does it depend on the level of a specific provider but also is it delivered via the Inter-net platform, which is due to susceptibility to failure and delay incurrence.27,50 Moreover, in the case of a dying VM, a new one must be replaced to ignore downtime; then, to preserve the system, the minimum time to repair would be as low as possible.27 The aforementioned features can only be found in MCE. In contrast to a single cloud, multicloud offers choice and flexibility for users in which they can provide services in competitive markets with low price and better QoS, thus preventing long-term vendor lock-in.51 Most importantly, a single cloud has limitations in presenting high availabil-ity, data/service integration, and data confidentiality and fails in several situations.28	Ivon Miranda Santos
Shirvani2018-An_iterative_mathematical_decision_model_for_cloud	CLOUD-NATIVE ARCHITECTURE > Flexibility	Multicloud also brings several benefits such as vendor lock-in avoidance and system fault tolerance. As such, open APIs provide portability and a high degree of flexibility. Service deployment is typically available, which depends	Ivon Miranda Santos
Shirvani2018-An_iterative_mathematical_decision_model_for_cloud	CLOUD-NATIVE ARCHITECTURE > Flexibility	Finally, in multicloud, it is possible for an organization to split their service components and delegate them to different providers with different pricing schemes. Although it makes flexibility with handful services available, it raises porting, deployment, and risk costs. Moreover, experiences in this context can mitigate	Ivon Miranda Santos
Shirvani2018-An_iterative_mathematical_decision_model_for_cloud	CLOUD-NATIVE ARCHITECTURE > Scalability	Then, we calculate variables CPV and AMFC as cost function and security risk, respec-tively, for their presented solution based on our formulation. We compare our model with other models according to the fitness value, which shows the balances between objectives, optimality in terms of cost, security SLA and total cost, scal-ability, and time complexity. Furthermore, to reach concrete results, we define 9 different scenarios in which the number of requested web services ranges from 10 through 30, whereas the number of	Ivon Miranda Santos
Shirvani2018-An_iterative_mathematical_decision_model_for_cloud	CLOUD-NATIVE ARCHITECTURE > Scalability	Consequently, it finds clouds with low cost and good security SLA coverage. To assess our model's scalability and time complexity, we also run several scenarios in the worst case with 30 requested web services from MCE in which the number of providers ranges from 30 to 500, as Figure 14 presents. We can conclude that our model is scalable even in large-scale multicloud market along with their abundant services so that	Ivon Miranda Santos
Shirvani2018-An_iterative_mathematical_decision_model_for_cloud	CLOUD-NATIVE ARCHITECTURE > Scalability	We have conducted several scenarios to demonstrate our model robustness and to gain concrete results. It has been proved that our proposed model beats a single-objective optimization and random biobjective model in terms of fitness, optimality, scalability, and time complexity. In other words, we have demonstrated that our model can find a set of optimal solutions in	Ivon Miranda Santos
Shirvani2018-An_iterative_mathematical_decision_model_for_cloud	CLOUD-NATIVE ARCHITECTURE > Flexibility	On the other hand, Op3 is the worst because it constantly increases the service costs to 15%. However, the flexibility and choice features of multicloud provide an opportunity for customers to find better cost and better QoS/SLA in the market. Despite the fact that farther we go, the same amount of cloud service decreases; if we suppose stagnancy in pricing schemes, Op2 is still the best option with the presumption that we are pinning to	Ivon Miranda Santos
Shirvani2018-An_iterative_mathematical_decision_model_for_cloud	CLOUD-NATIVE ARCHITECTURE > Flexibility	First, in the case of MCE adoption, we envisage solving the minimum and maximum allowable cloud usage, in which the minimum is needed to avoid vendor lock-in and create a failure-resistant system, whereas the maximum is needed to have more flexibility, better choice and cost, reliability, and security compared with the competitive market. Although reliance on MCE is a promising alternative, further number of used clouds may	Ivon Miranda Santos

Shyamasundar2017-Information_flow_control_for_building_security	CLOUD-NATIVE ARCHITECTURE > Flexibility	Abstract—A hybrid cloud is a cloud computing environment in which an organization provides and manages some internal resources (private cloud) while the other resources are provi-sioned externally (public cloud). Rapid deployment of hybrid clouds for utility, cost, effectiveness and flexibility has made it necessary to assure the security and privacy of hybrid clouds as it transcends different domains. Further, successful hybrid cloud implementation requires a well-structured architecture supporting the functionalities of both private and public clouds	Ivon Miranda Santos
Shyamasundar2017-Information_flow_control_for_building_security	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Further, successful hybrid cloud implementation requires a well-structured architecture supporting the functionalities of both private and public clouds and the seamless transitions between them.	Ivon Miranda Santos
Shyamasundar2017-Information_flow_control_for_building_security	CLOUD-NATIVE ARCHITECTURE > Flexibility	In this paper, we present an approach to building a hybrid cloud that preserves the given security and privacy policy by integrating an RWFM security module into a cloud service manager. An advantage of RWFM is that it provides a uniform solu-tion for securing various kinds of hybrid cloud architectures ranging from the simple pairwise federation to the complex interclouds, and supporting varying degrees of flexibility in workload placement ranging from a simple static placement to fully dynamic migration. Further, RWFM framework is forensic-ready by design, because the labels of data and services readily	Ivon Miranda Santos
Shyamasundar2017-Information_flow_control_for_building_security	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	An advantage of RWFM is that it provides a uniform solu-tion for securing various kinds of hybrid cloud architectures ranging from the simple pairwise federation to the complex interclouds, and supporting varying degrees of flexibility in workload placement ranging from a simple static	Ivon Miranda Santos
Shyamasundar2017-Information_flow_control_for_building_security	CLOUD-NATIVE ARCHITECTURE > Flexibility	HYBRID CLOUD While there are substantial benefits in using cloud services from cost as well as flexibility, it suffers from the fact that the users are not clear about the security and privacy of their data. Storing data on places/locations on which either the	Ivon Miranda Santos
Shyamasundar2017-Information_flow_control_for_building_security	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	A SECURE ARCHITECTURE FOR HYBRID CLOUD In this section, we provide a general approach for securing a hybrid cloud by integrating an RWFM	Ivon Miranda Santos
Shyamasundar2017-Information_flow_control_for_building	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	An example hybrid cloud architecture	Ivon Miranda Santos
Shyamasundar2017-Information_flow_control_for_building_security	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	In this paper, we have demonstrated that the RWFM model provides a secure architecture for a hybrid cloud assuring full security and privacy compliance in its private cloud and has the capability to keep track of the influences by the	Ivon Miranda Santos
SOrheller2018-Implementing_cloud_erp_solutions_A_review_of_soci	CLOUD-NATIVE ARCHITECTURE > Scalability	Hence, the increasing interest for cloud deployments of ERPs [1, 2] is not surprising. The benefits of cloud-based ERPs relate to cost effectiveness, time savings, scalability and ease of updates [1, 3, 4]. Nevertheless, the implementation of cloud ERPs is not straightforward and there are significant issues	Ivon Miranda Santos
Sousa2016-Automated_Setup_of Multi-Cloud_Environments_for_Micro	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Meanwhile, microservice architectures are becoming increasingly popular in cloud computing as they promote decomposing applications into small services that can be independently deployed and scaled, thus optimizing resources usage.	Ivon Miranda Santos
Sousa2016-Automated_Setup_of Multi-Cloud_Environments_for_Micro	CLOUD-NATIVE ARCHITECTURE > Scalability	However, setting up a multi-cloud environment to deploy a microservices-based application is still a very complex and time consuming task. Each microservice may require different functionality (e.g. software platforms, databases, monitoring and scalability tools) and have different location and redundancy requirements. Selection of cloud providers should take into account the individual requirements of each service, as well as the global requirements of reliability and scalability. Moreover, cloud providers can be very	Ivon Miranda Santos
Sousa2016-Automated_Setup_of Multi-Cloud_Environments_for_Micro	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	In a microservices architecture, applications are composed of small services that run in separate processes and can be deployed to different en-	Ivon Miranda Santos
Sousa2016-Automated_Setup_of Multi-Cloud_Environments_for_Micro	CLOUD-NATIVE ARCHITECTURE > Scalability	Having independent well-isolated services facilitates this. On the other hand running microservices across private and public clouds from different providers allows for improved scalability and reliability.	Ivon Miranda Santos

Sousa2016-Automated_Setup_of Multi-Cloud_Environments_for_Micro	CLOUD-NATIVE ARCHITECTURE > Scalability	In addition, each of the application's services may have different requirements. They may be written in different programming languages, using different application frameworks or databases, and may have different scalability and availability requirements. Thus, a cloud provider that supports one of the application's services might	Ivon Miranda Santos
Sousa2016-Automated_Setup_of Multi-Cloud_Environments_for_Micro	CLOUD-NATIVE ARCHITECTURE > Scalability	Microservices can be developed by different teams, relying on different technologies and methodologies and may therefore require functionalities at different levels of ab-straction. In a multi-cloud environment, microservices can be deployed across private and public clouds from different providers to implement scalability and redundancy mecha-nisms or to comply with location constraints.	Ivon Miranda Santos
Sousa2016-Automated_Setup_of Multi-Cloud_Environments_for_Micro	CLOUD-NATIVE ARCHITECTURE > Scalability	The ontologies are extensi-ble and allow defining complex requirements involving logical operators and composition rules. Finally, cloud variables and instance groups can be used to define scalability and redun-dancy rules across providers or regions, as well as location and colocation rules for services. B. Feature models for managing cloud variability Feature models are widely used to model commonalities and variability across a software	Ivon Miranda Santos
Sousa2016-Automated_Setup_of Multi-Cloud_Environments_for_Micro	CLOUD-NATIVE ARCHITECTURE > Scalability	of concerns between different user roles. We support the description of requirements for microservices-based applica-tions and mechanisms to achieve multi-cloud requirements of location, scalability and redundancy. Overall, our approach differentiates from others by taking into account variability in cloud configuration options, multi-cloud requirements and service	Ivon Miranda Santos
Stavru2013-Challenges_for_migrating_to_the_service_cloud_paradi	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Agile Software Development, Software Migration, Software Modernization, Cloud Computing, Service-Oriented Architecture.	Ivon Miranda Santos
Stavru2013-Challenges_for_migrating_to_the_service_cloud_paradi	CLOUD-NATIVE ARCHITECTURE > Flexibility	Not involving real customers (or at least somebody who could represent them) in the early phase could result in customer feedback received late in the development lifecycle, customer negotiation and relationship issues when real customers interests are conflicting (e.g. in terms of service functionality, interface, etc.); complex impact analysis and lack of flexibility and agility because of unknown customers (especially when the services are publicly available), etc. The prerequisite to define business processes and tasks in advance (O2), before the actual design, implementation and verification of services (if top-down or domain composition approach is incorporated) also poses some limitations on the use of agile methods and techniques, which	Ivon Miranda Santos
Stavru2013-Challenges_for_migrating_to_the_service_cloud_paradi	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Technical challenges T1 Addressing architectural and technical constraints Cloud Computing poses some architecture constraints on the way software systems are build, incl. decomposition, decoupling, componentization, etc. and some technical	Ivon Miranda Santos
Stavru2013-Challenges_for_migrating_to_the_service_cloud_paradi	CLOUD-NATIVE ARCHITECTURE > Coupling	Technical challenges T1 Addressing architectural and technical constraints Cloud Computing poses some architecture constraints on the way software systems are build, incl. decomposition, decoupling, componentization, etc. and some technical	Ivon Miranda Santos
Stavru2013-Challenges_for_migrating_to_the_service_cloud_paradi	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Cloud Computing and its challenges further affect the way agile methods and techniques could be incorporated into the Service Cloud Paradigm. Trust (O1), security and privacy (O3) of data and computation are as much important for the organization as for its customers, so the organization-customer collaboration, central to agile software development, might need to be extended to include the cloud provider (in order to increase transparency, visibility, responsiveness,	Ivon Miranda Santos

Stavru2013-Challenges_for_migrating_to_the_service_cloud_paradi	CLOUD-NATIVE ARCHITECTURE > Flexibility	, so needed for the building trust and confidence). Vendor lock-ins (O2) might further hinder organizational flexibility and agility (e.g. as one could not change its cloud provider effortless and in a timely manner), while external dependencies (O2) could decrease the business value delivered to customers (e.g. due to new requirements coming from the cloud infrastructure or the organization is pressured to use specific and expensive software licenses coming from the could provider, etc.) and could	Ivon Miranda Santos
Tona2020-DPS-AA_Intranet_migration_strategy_model_for_clouds	CLOUD-NATIVE ARCHITECTURE > Scalability	Thus, Intranet provides access to corporate information repository to the authorized user with a minimal cost, time and effort [5, 6]. This research study is an effort to design a migration strategy model for next generation intranets having on-demand Scalability, Open boundary Accessibility, Service reliability, High Availability (Uptime), High responsive, Fault tolerant with promised Security, Intelligent Search Content Filtration, Cost-effectiveness, and Anytime, Anywhere over Any device Accessibility. This design is proposed to be Migration Strategy	Ivon Miranda Santos
Tona2020-DPS-AA_Intranet_migration_strategy_model_for_clouds	CLOUD-NATIVE ARCHITECTURE > Scalability	traditional/on-premise Intranets. The next generation intranets needs on-demand Scalability, Open boundary Accessibility, Service reliability, High Availability (Uptime), Cost effectiveness, High responsive, fault tolerant and high Performance with promised Security, Anytime, Anywhere over Any device Availability[4,5,7,8]. As on date such issues, challenges and limitations (on-demand Scalability, Open boundary Accessibility, Service reliability, High Availability (Uptime), High responsive, Fault tolerant with promised Security, Intelligent Search Content Filtration, Cost-effectiveness, and Anytime, Anywhere over Any device Accessibility) in the existing designs and practices over intranets are not addressed by specific design or models. This makes this research a worth initiating activity towards a new knowledge contribution in the domain and the	Ivon Miranda Santos
Tona2020-DPS-AA_Intranet_migration_strategy_model_for_clouds	CLOUD-NATIVE ARCHITECTURE > Scalability	Relevant to this study but, Intranet is maintained over the local or on premise servers. In proposed study, Intranet is proposed to be maintained over the global cloud for overcoming the limitations of existing intranets like on-demand Scalability, Open boundary Accessibility, Service reliability, High Availability (Uptime), High responsive, Fault tolerant with promised Security, Intelligent Search Content Filtration, Cost-effectiveness, and Anytime, Anywhere over Any device Accessibility.	Ivon Miranda Santos
Tona2020-DPS-AA_Intranet_migration_strategy_model_for_clouds	CLOUD-NATIVE ARCHITECTURE > Scalability	2014 Migrating an enterprise application over the cloud platform using application refactoring method to assist application developers Relevant to this study but, in proposed study not only application developers are assisted i.e. all Stakeholders of the organization are proposed to be benefited from usage point of views. The drawback of this study was a lack of focus on the points; on-demand Scalability, Open boundary Accessibility, Service reliability, High Availability (Uptime), High responsive, Fault tolerant with promised Security, Intelligent Search Content Filtration, Cost-effectiveness, and Anytime, Anywhere over Any device Accessibility.	Ivon Miranda Santos
Tona2020-DPS-AA_Intranet_migration_strategy_model_for_clouds	CLOUD-NATIVE ARCHITECTURE > Scalability	The identification and classification of application workloads running on the existing platform is considered before migration over the cloud especially in reference to intranet, it is desirable. As per our view the features like on-demand Scalability, Open boundary Accessibility, Service reliability, High Availability (Uptime), High responsive, Fault tolerant with promised Security, Intelligent Search Content Filtration, Cost-effectiveness. Anytime, Anywhere over Any device Accessibility, and are not considered	Ivon Miranda Santos

Tona2020-DPS-AA_Intranet_migration_strategy_model_for_clouds	CLOUD-NATIVE ARCHITECTURE > Scalability	It provided product onto a cloud IaaS or PaaS platform with mapping a licensing model onto a cloud monetization model along with income & expenses in the cloud in relation to the chosen cloud delivery model Relevant to this study because investigation & analysis of the existing system with reference to costing is required in the proposed study for intranet migration as well. Except the cost other features like on-demand Scalability, Open boundary Accessibility, Service reliability, High Availability (Uptime), High responsive, Fault tolerant with promised Security, Intelligent Search Content Filtration, and Anytime, Anywhere over Any device Accessibility. are again	Ivon Miranda Santos
Tona2020-DPS-AA_Intranet_migration_strategy_model_for_clouds	CLOUD-NATIVE ARCHITECTURE > Scalability	Data Collection Procedures The end user's awareness towards alternative Intranet model like Intranet over cloud vs., on-premise existing intranets for on-demand Scalability, Open boundary Accessibility, Service reliability, High Availability (Uptime), High responsive, Fault tolerant with promised Security, Intelligent Search Content Filtration, Cost-effectiveness, and Anytime, Anywhere over Any device Accessibility features data were collected and analyzed. In this section other robust features promised by alternative models were also	Ivon Miranda Santos
Tona2020-DPS-AA_Intranet_migration_strategy_model_for_clouds	CLOUD-NATIVE ARCHITECTURE > Scalability	Based on the survey, interview inputs and technical observations of researcher, it has been observed and analyzed that the awareness of user's community towards modernized Intranets like Intranet over cloud is missing in planning, designing and developments. Using new features like horizontal and vertical scalability, salable capacity, cross boundary accessibility, high uptime, information sharing dynamism, etc. has not been plasticized over existing state of art intranets in educational institutions of Ethiopia.	Ivon Miranda Santos
Tona2020-DPS-AA_Intranet_migration_strategy_model_for_clouds	CLOUD-NATIVE ARCHITECTURE > Scalability	As presented in the fig. 2, this model is designed and proposed for next generations intranets where features like on-demand Scalability, Open boundary Accessibility, Service reliability, High Availability (Uptime), High responsive, Fault tolerant with promised Security, Intelligent Search Content Filtration, Cost-effectiveness, and Anytime, Anywhere over Any device Accessibility can be incorporated or enhanced and named as "DPS-AA Migration Strategy Model". Before designing the migration model, the study conducted survey, interview and technical	Ivon Miranda Santos
Tona2020-DPS-AA_Intranet_migration_strategy_model_for_clouds	CLOUD-NATIVE ARCHITECTURE > Scalability	- 1) Data Security and Privacy, 2) Infrastructure & Platform Scalability, 3) Open Boundary Accessibility, 4) High Service Availability, 5) Cost-Effectiveness, and 6) Additional Add-On features, which are not present in the existing state of art Models (like Notification for Academic and Management Staff Leaders, Intelligent Search/Content Filtration Features, and User's & Application level	Ivon Miranda Santos
Tona2020-DPS-AA_Intranet_migration_strategy_model_for_clouds	CLOUD-NATIVE ARCHITECTURE > Scalability	The research study tried to investigate and analyze the issues, challenges and limitations in existing state of art intranets and their performance, features by analyzing the satisfaction levels of the participants. Finally, the research designed a Migration Strategy Model named as "DPS-AA Model" for the Intranet over Cloud Platforms towards enhancing the performance and extending the features like on-demand Scalability, Open boundary Accessibility, Service reliability, High Availability (Uptime), High responsive, Fault tolerant with promised Security, Intelligent Search Content Filtration, Cost-effectiveness, and Anytime, Anywhere over Any device Accessibility. As a contextual analysis of salient features of the intranet networks; this research conducted feature based analysis of three different types of computer networks namely Internet (the world-wide network of computers accessible to	Ivon Miranda Santos

Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Scalability	Such software is deployed on the self-managed on-premises servers. Monolithic architecture systems introduced many difficulties when transitioning to cloud platforms and new technologies due to scalability, flexibility, performance issues, and lower business value. As a result, people are bound to consider the new software paradigm with the separation of concern concept.	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Flexibility	Such software is deployed on the self-managed on-premises servers. Monolithic architecture systems introduced many difficulties when transitioning to cloud platforms and new technologies due to scalability, flexibility, performance issues, and lower business value. As a result, people are bound to consider the new software paradigm with the separation of concern concept.	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	This paper illustrates the taxonomical classification of microservice architecture and a systematic review of the current state of the microservice architecture by comparing it to the past and future using the PRISMA model.	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	The results showcase that most researchers and enterprise-grade companies use microservice architecture to develop cloud-native applications.	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	The acquired results can facilitate the researchers and architects in the software engineering domain who aspire to be concerned with new technology trends about service-oriented architecture and cloud-native	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	The acquired results can facilitate the researchers and architects in the software engineering domain who aspire to be concerned with new technology trends about service-oriented architecture and cloud-native	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_syst	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Keywords - Microservices, Systematic review, PRISMA, Cloud computing, Architecture.	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Amidst the popularity of cloud computing, people are keener towards a developed application that possesses the capability of using cloud services and deploying on the cloud environment. That kind of application is called a cloud-native application. New technology trends are also	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Software architecture is influenced by cloud-native technologies as well [3]. As a result, microservice architecture is introduced to the world to overcome the issues of existing software architectures. In the microservice architecture, all the services are deployed in the distributed environment and necessary services are called for to satisfy the	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_syst	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Software architecture is influenced by cloud-native technologies as well [3].	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Clarified and compared the microservice architecture's different qualities and how they evolved from the earlier stage to the current state and then used current highly practiced research methodologies, techniques, research approaches, and methods to conduct the	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	This research study reveals that microservice architecture has addressed the issues which persist in the monolith software architecture.	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Through this study, persons who are looking for converting a monolithic application to the microservice architecture can gain ample ideas to	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	Through this study, persons who are looking for converting a monolithic application to the microservice architecture can gain ample ideas to	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	The main challenge in transforming applications to microservice architecture is to define the services as independent modules.	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	The main challenge in transforming applications to microservice architecture is to define the services as independent modules.	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	The study focused on three main areas such as types of research conducted related to the microservices, motivation behind microservice architecture, and the emerging trends in	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	A systematic grey literature review was conducted by the group of research related to the gap of the microservice architecture and also to the advantages of the microservice architecture [8].	Ivon Miranda Santos

Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	Handling the distributed storage and application testing on the distributed environment is mentioned as pains on the microservice architecture in the application development stage.	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	In the operational time, the main pain point is the huge network consumption because of the inter-service communication in a microservice	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	When developing the overall solution using the microservice architecture, all teams need to conclude what sort of communication mechanism should be used for the interservice	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_syst	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	Nowadays, most architects use cloud-native software architecture for the development of	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_syst	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	According to the researchers, the best cloud-native software architecture is microservice	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Scalability	According to the researchers, the best cloud-native software architecture is microservice architecture [9]. They comprise several non-functional requirements for the cloud-native applications such as elasticity, scalability, automated deployment, and vendor lock-in avoidance [10]. Docker the Rocket	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_syst	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	According to the researchers, the best cloud-native software architecture is microservice	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	They comprise several non-functional requirements for the cloud-native applications such as elasticity, scalability, automated deployment, and vendor lock-in avoidance [10].	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Docker Swarm and Mesos automated container management help microservice architecture to bring those non-functional requirements to the	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_syst	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	A German university conducted research related to the features of the microservice architecture	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	According to their research, security, performance resilience, reliability, latency, and fault tolerance are the most important features of	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Scalability	According to their research, security, performance resilience, reliability, latency, and fault tolerance are the most important features of the microservice architecture. Most people use the microservice architecture to get proper scalability, extensibility, and agility. With these intentions, engineers focus on the security of the microservices because it is deployed in the	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_syst	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Most people use the microservice architecture to get proper scalability, extensibility, and agility.	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	In the qualitative research approach, they collect feedback from the engineers who are involved in the microservice architectures via interviews, open-ended questionnaires, and surveys.	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	In summary, identified that most of the industries use the microservices architecture for their upcoming developments and some of them are faced with difficulties in the microservice	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	The key motivation of this study is to find out the encouragement of converting the old system to the microservice architecture and fill the gap of the microservice architecture by the current state of practice using the systematic review of the	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_syst	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	What are the main motivations to convert the monolithic application to microservice	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_syst	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	What are the main motivations to convert the monolithic application to microservice	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	What are the technologies & architectural patterns used in microservice architecture with	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	The aim is to address the above-mentioned research questions by studying the past research activities, and in the meantime provide taxonomical classification regarding the	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_syst	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	a) What are the main motivations to convert the monolithic	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_syst	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	a) What are the main motivations to convert the monolithic	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	In this scenario, intend to elaborate on the main motivations behind converting monolithic systems into a microservice architecture.	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	The existing issues people face in the monolithic, and how they mitigate those problems via the microservice architecture will be further discussed	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	b) What are the technologies & architectural patterns used in microservice architecture with technological	Ivon Miranda Santos

Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Here, aim to bring several architectural patterns for problem-solving in the microservice architecture, as well as discuss the new technological tools and frameworks that are used	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Noticed that some of the research studies elaborate on the problems associated with the microservice architecture in the software	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	General Search string – (“Microservices” OR “Microservice” OR “micro-service”) AND (“monolithic application to microservice architecture”) OR (“microservice framework” OR “microservice development tools”) OR (“microservice architectural patterns”) OR	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	General Search string – (“Microservices” OR “Microservice” OR “micro-service”) AND (“monolithic application to microservice architecture”) OR (“microservice framework” OR “microservice development tools”) OR (“microservice architectural patterns”) OR	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_syst	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	A. What are the Main Motivations to Convert the Monolithic Application to Microservice	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_syst	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	A. What are the Main Motivations to Convert the Monolithic Application to Microservice	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Scalability	It can categorize people’s intentions to choose the architecture as product, cost, and process. When considering the product, most of the engineers focus on the product scalability over the cloud, product maintainability, performance, and as well as product security [33]. Most people are moving towards digital services, and service consumers tend to use digital services as well.	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_syst	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Due to this fact, people are motivated to move their architecture to microservice-based	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_syst	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	In a microservice architecture, all the services work independently and are deployed separately.	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_syst	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	But in the microservice architecture, developers can implement several checkpoints to validate	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	On the other hand, they struggle with the microservice architecture as microservice architecture is deployed in the distributed environment, and service to service communication will add some latency to the	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_syst	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	For such reasons, a lot of people move their systems towards a microservice architecture.	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	But with the microservice architecture, people can easily patch and adapt to the new requirements within short cycles with the leverage of Continuous Integration (CI) and Continuous	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_syst	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	With the microservice architecture, software companies can run an	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	B. What are the Architectural Patterns used in Microservice Architecture with Technological Advancement?	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_syst	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Microservice architecture is one of the emerging architectures in the software industry.	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	With the motivation towards microservice architecture, software companies convert their monolithic-based software and service-oriented architecture software to microservice-based	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	4 Architectural patterns in a microservice architecture Based on the above table 1, Spring Boot is the most popular and the number one trending microservice framework of the world, that	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Flexibility	Vert. X microservice framework does not give proper flexibility to bring the software quality attributes as required [63]. It is a slowly evolving framework compared to the Spring boot.	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Scalability	Continuous deployment of the software solution Scalability Emerging technology adaptation	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_syst	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	There are several microservice architecture patterns defined to design the data management.	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Scalability	New API management platforms are invented for the software industry daily. People move to microservice architecture to attain better scalability in their services. Microservices can be scaled into many services based on the traffic load, and if the traffic is low, such a service can be	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_syst	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	C. What are the Main Motivations to Convert the Monolithic Application to Microservice	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_syst	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	C. What are the Main Motivations to Convert the Monolithic Application to Microservice	Ivon Miranda Santos

Weerasinghe2022-Taxonomical_classification_and_syst	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	This paper elaborates the findings on the challenge when using microservice architecture.	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_syst	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Another type is converting monolithic systems or service-oriented traditional systems to	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	The first challenge in developing a microservice is to determine the scope of one service in a microservice architecture.	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	In the software industry, the scope of microservice is segregated into programming operational service on the overall microservice architecture such as DB service, messaging	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_syst	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Hence, based on that, can't determine the connection points within microservice	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_syst	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Data security across the microservice is a big concern in the microservice architecture.	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Some protocols generate more complexities to the microservice architecture, while some do not support the cloud-native environments.	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	In the microservice architecture, one single software solution has several services and needs to write deployment scripts to every microservice.	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_syst	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Supporting the microservice architecture is very challenging because of the service distribution.	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Bringing the fault tolerance to the microservice architecture is crucial as many services are involved with the distributed environment.	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	In a microservice architecture, enabling monitoring needs more effort and resources due to the small number of services deployed in the	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	After a critical review, identified the three main categories that microservice research is moving towards in near future AI, cloud, and architecture.	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	5 Taxonomy on microservices future trends Most of the monolithic systems are moved to microservice-based architecture to bring quality	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_syst	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	When moving to the cloud-native architecture, cost and application performance are the biggest	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	To get better performance with low cost in the cloud-native environment, the application should possess the capability to run on low computational specifications and need to perform	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Most of the microservice-related research mainly focuses on the cloud-native concepts and integration with artificial intelligence to the	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_syst	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Most of the cloud providers invent serverless architecture services and function as a service.	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	When the application is deployed into the production operation, the troubleshooting with the tracing is quite complicated in the microservice	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Scalability	Research questions are chosen to capture all aspects of the microservice research area. Most people are moving their monolithic system to microservice architecture to achieve the quality attributes such as scalability, performance, security, maintainability, etc. In the current context, with the emergence of world pandemic situations, most of the services are served via online platforms and a lot of users are moving	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Most people are moving their monolithic system to microservice architecture to achieve the quality attributes such as scalability, performance, security, maintainability, etc.	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	To achieve complete benefits of the cloud services application, developers need to change the application architecture to the cloud-native	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	To achieve complete benefits of the cloud services application, developers need to change the application architecture to the cloud-native	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Legacy application architecture	To achieve complete benefits of the cloud services application, developers need to change the application architecture to the cloud-native	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_systematic_revie	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	With the technological advancement and the architectural patterns, some of the challenges are overcome but they are still faced with several issues in the microservice architecture.	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_syst	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	The main concern of the microservice architecture is the latency because of the	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_syst	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Continuous research of this to find a solution for the performance issue in the microservice	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_syst	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	Therefore, future research needs to focus on the cloud-native development of the microservice	Ivon Miranda Santos
Weerasinghe2022-Taxonomical_classification_and_syst	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Therefore, future research needs to focus on the cloud-native development of the microservice	Ivon Miranda Santos

Wright2011-A_commodityfocused_multi-cloud_marketplace_exemplar_	CLOUD-NATIVE ARCHITECTURE > Scalability	An application is not designed for and deployed to a fixed resource infrastructure but designed for and deployed into a dynamic pool (or cloud) of resources. This separation of concerns can bring mobility because an application is not tied to a particular physical resource, and scalability since more resources can always be added. An application cloud is most often dynamic and designed to scale to meet demand from users for	Ivon Miranda Santos
Zhou2017-Cost_reduction_in_hybrid_clouds_for_enterprise_computi	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Unlike previous studies, we consider a general hybrid cloud architecture that involves multiple public clouds rather than only one.	Ivon Miranda Santos
Zhou2017-Cost_reduction_in_hybrid_clouds_for_enterprise_computi	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	It is reported that about 60% of IT decision-makers in the US and UK choose to adopt hybrid cloud architecture to deploy their applications [2].	Ivon Miranda Santos
Zhou2017-Cost_reduction_in_hybrid_clouds_for_enterprise_computi	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	In addition, while existing works have done a good job exploring the benefits of deploying applications in the simple two clouds environment(which contains both a private and a public cloud), they seldom evaluate the benefits of employing a hybrid cloud architecture with a private cloud and multiple geographically distributed public clouds.	Ivon Miranda Santos
Zhou2017-Cost_reduction_in_hybrid_clouds_for_enterprise_computi	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Mathemati-cally, we model a typical hybrid cloud architecture as a node set H = H ∪ h0, where h0 represents the on-premise data center and H represents M public cloud sites located in	Ivon Miranda Santos
Zhou2017-Cost_reduction_in_hybrid_clouds_for_enterprise_computi	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Finally, our framework performs better in reducing enterprise costs leveraging the hybrid cloud architecture under controllable time overhead than the other two strategies.	Ivon Miranda Santos
Zhou2017-Cost_reduction_in_hybrid_clouds_for_enterprise_computi	CLOUD-NATIVE ARCHITECTURE > Scalability	It cannot compute the solution for a DAG with nodes number larger than 1K in ten hours. In other words, our algorithm exhibits better scalability. V. CONCLUSION	Ivon Miranda Santos
Zhou2017-Cost_reduction_in_hybrid_clouds_for_enterprise_computi	CLOUD-NATIVE ARCHITECTURE > Cloud architecture	Unlike previous works, this work considers a more general hy-brid cloud architecture involving multiple public clouds rather than one.	Ivon Miranda Santos
Alonso2023-Understanding_the_challenges_and_novel_architectural	CLOUD-NATIVE ARCHITECTURE > Design applications for cloud native	We are assisting to a change of paradigm, from the usage of a single cloud provider to the combination of multiple cloud service types, afecting the way in which applications are designed, developed, deployed and operated	Ivon Miranda Santos
Alonso2023-Understanding_the_challenges_and_novel_architectural	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	The result is an efective heterogeneity of architec-tures, methods, tools, and frameworks, copying with the multi-cloud application concept.	Ivon Miranda Santos
Alonso2023-Understanding_the_challenges_and_novel_architectural	CLOUD-NATIVE ARCHITECTURE > Cloud-native architecture	The goal of this study is manifold. Firstly, it aims to characterize the multi-cloud concept from the application development perspective by reviewing existing defnitions of multi-cloud native	Ivon Miranda Santos
Alonso2023-Understanding_the_challenges_and_novel_architectural	CLOUD-NATIVE ARCHITECTURE > Multi-cloud native applications	The goal of this study is manifold. Firstly, it aims to characterize the multi-cloud concept from the application development perspective by reviewing existing defnitions of multi-cloud native	Ivon Miranda Santos