





# Cloud Migration High-Level Requirements

Antoine Aubé<sup>1,2</sup>(✉)  and Thomas Polacsek<sup>1</sup> 

<sup>1</sup> ONERA, Toulouse, France

{antoine.aube,thomas.polacsek}@onera.fr

<sup>2</sup> Stack Labs, Toulouse, France

**Abstract.** With the increasing adoption of Cloud Computing in the industry, new challenges have emerged for information systems design. In this context, many requirements are met by choosing adequate cloud providers, cloud services, and service configurations. To provide design support, it is necessary to understand what drives the selection of each of these elements. Therefore, it may be interesting to elicitate these high-level requirements, to have an overview of what significantly impacts the cloud environment selection. Here we focus on a particular case of cloud system design: migrations. Through a qualitative study with cloud migration experts, we identify eleven high-level requirements that drive design decisions. We propose an analysis of these results and two classifications to support elicitation and analysis of requirements in cloud migrations.

**Keywords:** Cloud Computing · System design · Requirements engineering

## 1 Introduction

*Cloud Computing* consists of using services to enable computing resources. These services are called *cloud services*, and they may be *Infrastructure as a Service (IaaS)* when they deploy virtualized infrastructure elements such as machines or networks, *Platform as a Service (PaaS)* when they deploy a framework for executing software, or *Software as a Service (SaaS)* when they are software hosted on cloud and provided to their end-users. Cloud services must also ensure rapid elasticity, which means allocating and deallocating resources very quickly, according to users' needs. [12] By configuring cloud services, users enable a *cloud environment* in which applications can be deployed. With Cloud Computing, the notion of infrastructure tends to disappear, as the hardware is hidden behind catalogs of services called *clouds*.

Adopting Cloud Computing can be attractive in terms of cost. Indeed, users pay only for what they consume. Organizations could also expect cost reduction, as they do not need to maintain an on-premise infrastructure with all the inherent costs (e.g., real estate, electricity, Internet access, maintenance workers). However, the reality is not this simple. One does not design information systems

for the cloud in the same way as traditionally. To be effective, a cloud environment must leverage new concepts brought by Cloud Computing, especially elasticity, to avoid over-consumption. There are also new situations due to the multi-tenancy of the system: it is common to have a part of the system hosted on a cloud and another on-premise. There is a dual objective: to select an environment (i.e., choose the cloud provider, cloud services, and service configurations) and avoid unnecessary costs.

This dual objective is challenging to manage in the early design phases. Indeed, it is not easy to assess the consequences of a cloud service choice on the satisfaction of the requirements in the specification phase. In practice, this lack of foresight leads to overspending and to system designs that do not fulfill the requirements. It is, therefore, a great advantage for cloud adoption to be able to assess these consequences. However, it is necessary to understand beforehand what drives the selection of a cloud provider or a cloud service. Therefore, even if these *drivers*, or *high-level requirements*<sup>1</sup> are various, it may be interesting to elicitate them to have an overview of what high-level requirements have a significant impact on the cloud environment selection. We choose in this study to only focus on migration. Cloud migration means transferring applications to a cloud environment. It involves changes to the system architecture, which depend on the chosen services. We can express our aim in the following research question:

*What are the high-level requirements, the drivers, for cloud environment selection (cloud services and service providers) in the migration context?*

In order to answer this question, we decided to conduct cloud experts interviews. Our approach is motivated by the lack of existing information in the literature. Indeed, as we will see in Sect. 2, while many works focus on organizations' motivation, the identification of the choice drivers is rarely addressed. Similarly, there is also some work that focuses on the choice of environments, but the requirement elicitation in cloud migration is not addressed. To conduct our interviews, we follow a methodology presented in Sect. 3. Section 4 is dedicated to the interviews, and in Sect. 5, we present the eleven high-level requirements that we extracted from the interviews. In Sect. 6.1, we compare them to the works published in the literature while, in Sect. 6.2, we propose two possible classifications for these requirements and conclude in Sect. 7.

## 2 Related Work

Since its introduction, many organizations have adopted cloud computing. Some studies have identified motivations that led these organizations to migrate their

---

<sup>1</sup> While we distinguish requirements (they come from users and stakeholders) and drivers (they are constraints related to the domain), this is not always clear in the early design stage. For instance, security can be as well a requirement and implicitly enforced at some point, as a certain level of security is now considered normal. Therefore, we choose to use both as synonyms in this paper.

systems. For instance, in the early days of cloud computing, Jamshidi *et al.* [10] have led a systematic review of studies about cloud migrations and concludes with six main motivations. In the same year, Nussbaumer *et al.* [13] studied the motivations of Small and Medium-sized Enterprises (*SMEs*). Both conclude with a set of motivations that are closely tied to the technical characteristics of cloud services that make them more convenient than on-premise infrastructure. Eight years after, Bremer *et al.* [3] led another systematic literature review of these motivations with a focus on *SMEs*. They conclude with a more extensive set of motivations that include most previously found, and many new, that are mainly related to organizational aspects and consider Cloud Computing not simply as an alternative to hosting software but as a means to develop business. These works focus on the motivations for adopting cloud computing while we focus on the drivers for selecting a cloud environment. We notice that some highlighted motivations may be considered as such a driver. For instance, an organization that migrates a system for reliability issues (it is their motivation) will consider new requirements for the system's reliability that will drive the selection of services.

One can distinguish migration strategies by taking perspective from the cloud environment selection. Cloud migration strategies are high-level schemes defining activities to move a system to a cloud. Many works propose a classification of these strategies. For instance, Binz *et al.* [2] distinguish three broad classes of system migration strategies which depend on how the system's components are adapted. In contrast, Andrikopoulos *et al.* [1] focus on the kinds of services leveraged by the migration. In the same spirit, the consulting firm Gartner [8] considers five options, widely called *the 5 R's*. Finally, Zhao and Zhou [17] synthesized several migration strategy classifications and proposed theirs, distinguishing five classes mixing software adaptation and kinds of leveraged services. These classifications describe the possibilities for migration, yet it remains unclear which one to choose to satisfy the requirements best.

In contrast with the strategies, migration methodologies describe processes for achieving migration. They are extensively documented. For example, Haselbring and Frey propose a model transformation-based methodology, called *CloudMIG* [7], for automating the migration of a legacy system into a *SaaS*. In contrast, Zhang *et al.* [16] propose a generic migration methodology in which a system redesign to *service-oriented architecture* allows to call legacy functionalities from web services hosted on virtual machines. To sum these methodologies up, *MLSAC* [6] is a model-driven framework to express diverse reengineering methods to migrate legacy applications to a cloud. Gholami *et al.* [9] also led an extensive analysis of migration methodologies. While the tailorability of methodologies is one of the survey criteria to adapt the methodology to one migration, the suitability of the methodology for this migration is not addressed. For us, this suitability depends on the migration requirements and needs to be established to determine the best methodology.

### 3 Methodology

To answer our research question, we conducted a qualitative study to collect feedback from cloud migration experts. Qualitative studies are common in many research fields, such as medical studies [5]. They are often in the form of semi-structured interviews, which are interviews with a pre-established set of open-ended questions. We followed the qualitative surveys empirical standard of *ACM SIGSOFT* [14].

*Interviewers.* The interviewer is a Ph.D. student who has worked as a cloud developer in the industry for three years.

*Sampling.* The sampling aimed to interview industry employees who participated in the initial phases of at least one cloud migration. We wanted to filter participants who only have theoretical knowledge of cloud migration. As recommended by McCracken [11], we have looked for diverse profiles with proximity to our research question; in this sense, we have sought participants with various professions who contribute to cloud migrations.

*Data Collection.* Each interview had a duration of between 45 min and one hour. The interviews were face-to-face or by video conference, one participant at a time. The audio was recorded with the interviewee’s agreement for later analysis.

For each interview, we always follow the same pattern. The interview is divided into three parts. Firstly, we explain the purpose of our study and how the interview will be conducted. We also define the specific terminology we are using in order to avoid any ambiguity (e.g., *cloud environment*, *cloud system*, *requirement*). Secondly, interviewees briefly introduce themselves and the context of

**Table 1.** Interview questions

#	Interview question
1	How did you analyze the components to migrate? What information did you search for?
2	How did you consult the requirements already met by the system to migrate?
3	Do you have metrics or proofs of the satisfaction of those requirements?
4	Did you gather information about how users interact with the system?
5	How did you elicitate the cloud-specific requirements?
6	Were there requirements for the quality of service? If so, what aspects?
7	Were there requirements for the system’s environmental performance? If so, what aspects?
8	Was the cost of workforce accounted for in the operating budget?
9	How were expressed the requirements towards the operating budget? Was it for the whole system, one budget per functional unit, etc.? Were these fixed amounts, or was it more complex? Were there tolerance thresholds, etc.?
10	Were there requirements for the software adaptation endeavor?

the migration project. In this step, we collect the company business domain, the role and responsibilities of the interviewee, and their number of years of experience in the field of cloud migration. Then, in the second step, we have a general discussion where the interviewee briefly introduces the context of migration and his role. Our work of identifying the drivers really begins in the third part of the interview. This is the main part, where we collect the interviewee’s feedback. To structure this step, we built an interview guide with ten questions which can be found in Table 1. Questions 1 to 4 address the analysis of the existing system. Question 5 focuses on the requirements elicitation method, while the others explore the performance of the system (questions 6 and 7), operating budget (questions 8 and 9), and migration cost (question 10).

## 4 Corpus

All the interviewees are cloud consultants: they support organizations in migrating to the cloud. They work with a wide range of organizations: small and large companies, government organizations, and international companies. So this allowed us to get feedback on various migration projects (e.g., big or small teams, different business domains). Although they told us about projects in their careers, all the experts in this study were consultants at Stack Labs, a French company specializing in cloud computing consulting, at the time of the interviews.

**Table 2.** Interviewees’ Demographic Information

#	Domain	Role	Exp. (years)
P1	Satellite imagery	Project Manager	18
P2	Satellite imagery	Cloud Architect	10
P3	Satellite imagery	Data Engineer	17
P4	Agriculture	Cloud Developer	19
P5	Nuclear industry	IT Operator	4
P6	Local authorities	Cloud Enabler	16
P7	Consumer goods selling	DevOps	1
P8	Consumer goods selling	Project Manager	8
P9	Audience monetization	Cloud Enabler	15
P10	Ministry	IT Operator	1
P11	Health	DevOps	8

Regarding the migration context, the role of the interviewees, and their number of years of experience in the field of cloud migration, answers are summarized in Table 2. The participants’ experience varies from 1 to 19 years, but most had more than ten years of experience in the industry. Among the roles, *DevOps*

does not refer properly to the eponym methodology but to a position in charge of both the development and operations; this is how the interviewees named their role.

Note that interviewees encountered difficulties in collecting requirements. For instance, *P1* did not know the maximum budget allocated to operations:

*“I think they had one, but they never told us about it. I guess they were waiting for our proposal to see if it suited their needs.”*

We have two hypotheses to explain this. On the first hand, the interviewees were consultants, so the information may not have been shared to encourage proposing the least expensive architecture possible. On the other hand, the companies might not have been able to estimate such a budget with relevance because they lack expertise in cloud computing.

## 5 Cloud Migration High-Level Requirements

The following paragraphs present our *Cloud Migration high-level Requirements* (CMR). They are sorted by decreasing occurrences in the interviews.

*Operations Effort (CMR1)*. Once a cloud system is in production, human operators are charged to monitor and repair it to keep it running smoothly. These operations activities may be automated and assisted by using adequate cloud services. For instance, logging the system’s traces is essential for analyzing errors and correct. It requires a great deal of time and expertise to design, configure and operate a reliable logging tooling on *IaaS* because it is up to the operations team to ensure many properties, such as cross-logging between machines (in the case of one of them is not available) or disk space availability over time for traces storage. However, this time and expertise may be spared by the use of some *PaaS*, such as *Cloud Logging* (a *Google Cloud* service), which delivers the tooling for reliably storing traces without operations team involvement.

All interviewees mentioned the need to diminish the operations effort as a cloud environment selection driver. The motivation is not the same in all projects.

In some of them, mostly in small projects without an operations-dedicated team, it is to reduce the workload of teams in place. For instance, the migration of *P4* involved the modernization, via the use of a workflows orchestrator, of many processes which were manually triggered. They choose a *PaaS* delivering *Apache Airflow*<sup>2</sup> over other solutions to avoid the team managing that component by itself. For the same reason, *P2* chose to use *serverless* services over virtual machines for some components:

*“They had services written in Java and hosted on Tomcat application servers, which perfectly fits [a PaaS service], so we used it [...] that costs nothing (i.e., in terms of development) and it is far more practical [...], so there was no point to dismiss this possibility.”*

---

<sup>2</sup> Workflow orchestration platform. <https://airflow.apache.org>.

In other projects, reducing operations effort eases the organization to scale and postpones hiring new operators. In this regard, *P7* migrated several systems of a company to the same cloud, using the same *PaaS* containers orchestrator to standardize the technologies used in the company and allow the company to merge operations teams, making it easier for them to enable a new system:

*“They used to have various platforms, some on-premise, others on public or private clouds, and various technologies [...]. That was a pain for the operations teams, so the migration goal was to enforce one platform and one containers manager for all the projects [...].”*

*Cloud Environment’ Costs (CMR2)*. A cloud environment has a recurring cost billed annually to organizations. This cost depends on the resources’ configurations and usage quantities.

Many services may fit the same architectural role but are priced differently. For instance, to store files, a virtual disk may be deployed by an *IaaS*, and its cost depends on the size of the disk, its location, and its type. Objects storage services are *PaaS* alternatives priced on the average stored amount of data over the billed period and at each user interaction (e.g., read, write, list). The cheapest solution depends, in fact, on the system usage by its users.

To comply with the performance and cost requirements, *P5* opted for an *IaaS* to host a software forge so that they could configure a bespoke platform. In contrast, the *PaaS* alternative had either undersized or oversized configurations. This way, they lowered the environment’s costs while maintaining a high operations effort, which is acceptable when migrating from on-premise infrastructure, as *P9* said:

*“For each project (i.e., a customer of the company), there needs to deploy a new platform which means that, on-premise, they would deploy one server, or two, or X depending on the load. Cloud computing is a perfect fit for the situation. The on-premise infrastructure made it until the migration, but at an exorbitant cost [...].”*

Minimizing the environment’s costs was mentioned in most interviews, but only *P5* made it a priority. In most cases, this goal is less important than other criteria. For instance, *P9* could have used *serverless* services, which would have been a cheaper solution, but members of the team did not want this kind of service:

*“We (i.e., the DevOps team) faced dogmatic developers who were against using managed services. Using [an IaaS] has already been difficult for them to accept.”*

*Cloud System Reliability (CMR3)*. Organizations put in place a policy called a *Disaster Recovery Plan* (DRP) to respond efficiently in case of system failure. This plan aims to minimize the impact of failures on their customers (e.g., service

interruption, data loss). Such failures may be caused by a climatic event (e.g., earthquakes, fire) or malicious operation.

When hosting a system on-premise, it is the organization's responsibility to deploy the whole strategy, which may be costly. This may involve, for example, provisioning a stock of hardware in case some machines shut down or ensuring that there is always an on-site operator to handle the incident. However, the actions are limited. For instance, systems in many organizations are deployed in only one premise. There is, therefore, no backup in the event of a premise-wide failure such as a power outage.

The migrations of  $P1$ ,  $P2$ ,  $P3$ ,  $P4$  were initiated after a failure that had consequences on their business. Some of the companies had an insufficient *DRP*, and some others did not have one.  $P2$  told:

*"The goal [of the migration] for the system's operations was to bring robustness [...] The first objective was to make it reliable because, initially, everything relied on some machines and a single engineer to operate the whole. [...] Their initial setup made it very difficult to recover after a disaster."*

Migration was therefore associated with creating a *DRP*. Indeed, clouds have qualities that facilitate a *DRP* implementation and minimize its cost via the responsibility transfer of a part of the system's operations. It also offers opportunities of making *DRP* more robust. For instance, services are delivered with *Service Level Agreements (SLA)* which give insurance about some properties, such as the resources' availability or durability. Resources may also be deployed worldwide to duplicate data and computing units in distant places, so a disaster in one location cannot interrupt the whole service.

The concern of reliability is essential for companies delivering a service with a *SLA* by themselves. The team of  $P9$  worked with was concerned about migrating for this reason, as the loss of system uptime means customers will demand refunds and, possibly, terminate the contract with the company. As of  $P11$ , they reported concerns about their team's user experience. Indeed, the system would crash when too many users interacted with it. As a result, they leveraged the elasticity of cloud computing to handle those occasional spikes.

*Latency (CMR4)*. In many projects, latency experienced by the user was a significant concern because a too-long wait could jeopardize the system's adoption.

It was a network latency issue for the company where  $P3$  intervened. They used to host their system on-premise in France, which resulted in poor performances for their new customer in South America. To solve this issue, the cloud provider has been chosen so that it can deploy the system both in Western Europe and South America.

In other situations, the company focuses on workload processing speed when they deal with heavy computation tasks. One of the goals of the migration of  $P5$  was to reduce the software building time (e.g., compilation, GPU-intensive tasks) of a software forge. They used bare metal machine services to configure tailored machines for the task (e.g., with GPU) and to limit the virtualization



overhead. They then installed a container orchestration system on top of these machines:

*“We had two possibilities: either use a managed Kubernetes<sup>3</sup> or build it ourselves. We made the second choice for performance reasons. Because the virtual machines behind the managed Kubernetes did not meet the characteristics we were looking for, or, otherwise, it would have been far too costly [...].”*

*P4* also had this requirement because the system has a cumbersome workload process twice a year. It must have enough resources to handle it but does not need them for the rest of the year. Elasticity has been the key in this situation, with the tasks orchestration platform leveraging highly scalable serverless computing resources to shorten the periodic workload processing.

A third way to deal with latency was reported by *P9*. Performance was the company’s primary concern, so *P9* measured the user-perceived latency before and after the migration.

*“[...] Before, all the servers were in the same room, now they are in several gigantic data centers, data must pass through local networks, the Internet, etc., so obviously we lose performance. It is something that I measured [...], and they accepted my proposal because I had planned solutions to compensate for the performance loss.”*

The mentioned solutions consisted of software adaptation; for instance, they switched the communication protocol to a lighter one.

The interviews highlighted elasticity and the use of tailored *IaaS* as solutions to the latter issue; we expect the resources’ collocation to be also one, but no participant mentioned it.

*Need of New Skills (CMR5)*. Hosting a system on a cloud requires specific technical skills to develop and operate this system. Some of these skills are related to cloud computing specificities, such as elasticity, which must be considered to build an efficient system.

Some others are cloud provider-centered and depend on the chosen services. For instance, a *IaaS*-based environment can host the same software as an on-premise infrastructure, with little or no adaptations. However, it would take a lot of adaptations to host this software on a serverless service, as it needs to use this service’s capabilities (e.g., use a supported programming language, use a specific library). Adaptations are specific to a service in particular: if the software is to be hosted on another serverless service, other adaptations will be necessary.

Some interviewees have worked in small teams with developers who are also in charge of the operations. In these cases, the company could not hire new employees. They could not even replace some to avoid the loss of business-related knowledge. Cloud services had, therefore, to be chosen to minimize the need for

---

<sup>3</sup> Kubernetes, a widespread containers orchestrator: <https://kubernetes.io/>

formation for the team. For instance, *P2* cited the requirement of keeping *Python* as the development language:

*“It had to be maintainable by them, in terms of development. [...] For instance, the scripts had been written in Python, and we could not suggest that they should be rewritten in Go because the developers were data scientists. They knew Python because it is widely used in their field, and it is quite difficult to switch languages overnight.”*

*P6* indicated that the choice of cloud provider was evident in their migration because the operators had experience with one, so they chose it.

*Digital Sovereignty (CMR6)*. Digital sovereignty does not have a unique definition for organizations, but it is a more and more important theme in the cloud computing industry. Regulations such as the *CLOUD Act* in the United States of America or the European *General Data Protection Regulation (GDPR)* show the will of governments to take this theme in charge.

In practice, organizations concerned with digital sovereignty impose constraints on the location of cloud resources and the legislation ruling cloud providers.

This concern was present during the migration of *P5* because the system deals with sensitive data related to French energy sovereignty. It was therefore required that all data and processing remain in France and that the cloud provider follows the French legislation:

*“The IT direction of [the company] wanted that the system stayed in France. It was about french nuclear plants, so they did not want it to be hosted on [an American cloud provider] [...].”*

Depending on the context, the implementation of digital sovereignty may be less restrictive. For instance, *P6* had to make a trade-off between digital sovereignty and costs:

*“They wanted their data hosted on a sovereign cloud, so we suggested a trade-off: host the data on [an American cloud provider] and the cipher key on [a French cloud provider].”*

Sometimes, this concern implies not migrating some components to a cloud. For example, *P8* related that some data were stored in China and must remain there, but the chosen cloud provider does not have a data center in this country. So the system is hybrid to meet the digital sovereignty requirements.

*Security (CMR7)*. It surprises us that only a few interviews mentioned security as a driver of the cloud environment selection.

*P6* sliced the migration into phases: the first to migrate the whole system on *IaaS*, then several modernization phases to leverage more interesting services. Many services were put aside later because a network topology was established and validated in the first phase. Using these services would have broken

this topology, possibly introducing security issues. It is indeed necessary, as the French state placed some components of the system under the status of *Scientific and technical potential of the Nation*<sup>4</sup>, which imposes the implementation of means to protect it from espionage and piracy.

*P8* stipulated that security was essential for migration. In particular, their company is divided into multiple sectors, and they wanted the insurance of data tightness between sectors. They leverage their cloud provider's identity and access management to achieve this.

*Adherence to the Cloud Provider (CMR8)*. Although the amount of research works aimed to reduce cloud vendor lock-in [15], the interviews seem to indicate that the industry has not embraced these solutions. Instead of putting in place measures to limit the vendor lock-in, the team decides whether to accept or not this adherence.

*P5* said that the team wanted to minimize the vendor lock-in because it was the company's policy. If the cloud provider changes its conditions or pricing, they want to shift to another provider immediately. To achieve this, they relied on open source standards and widely used software such as *PostgreSQL* delivery as a *PaaS* or *Kubernetes*:

*“We chose Kubernetes because it is an open industry standard [...] allowing the company to be most independent to other companies.”*

On the contrary, some companies require one cloud provider in particular. In the case of *P1*, the company wanted to host the system on *Google Cloud* because they use many other *Google* products and want to take advantage of this ecosystem.

*Cloud Migration' Cost (CMR9)*. Many requirements involve adaptations to software (e.g., use of *PaaS*), which represent development tasks that increase the cost of migration.

However, the migration budget was very scarce in some projects related to the interviewees. Consequently, the adaptations were confined to the bare minimum, restricting the choice of services. As mentioned before, *P2* explained that some micro-services from the existing system were migrated to *Google App Engine*<sup>5</sup> with very little work because they were written in Java and running on a Tomcat server, which are compatible characteristics with this service. The rest of the system has been hosted on *IaaS* virtual machines, which are less attractive in terms of costs but need no work for the migration.

*Cloud Migration' Duration (CMR10)*. Some migrations are not constrained by the budget but by time because they have a firm deadline.

<sup>4</sup> French protection program for the Nation's assets: <http://www.sgdsn.gouv.fr/missions/protection-du-potentiel-scientifique-et-technique-de-la-nation/>.

<sup>5</sup> App Engine, a Google Cloud *PaaS*. <https://cloud.google.com/appengine>.

This is what *P2* related:

*“As their activity is seasonal, the migration had to be finalized before the start of the coming season.”*

In their case, because the budget was also an issue, they renounced modernization. For instance, they initially intended to change the database to use a *PaaS* database with similar performance at a much lower cost, but the change would have taken too long. If the budget had been less limited, they might have involved more developers and made these adaptations. This driver is, therefore, not the same as the previous one.

*Ecological Efficiency (CMR11)*. Ecology in information systems has been a popular theme for a while. In 2008, Chen *et al.* [4] published propositions for achieving the ecological efficiency of information systems. However, only a few organizations consider ecology while designing their systems.

Clouds offer opportunities regarding this theme thanks to the pooling of a significant number of resources which, on the first hand, encourages the cloud provider to find solutions to minimize energy consumption and hardware purchases to make substantial savings and, on the other hand, avoids the multiplication of premises and IT resources for each organization.

Some cloud providers integrate features such as low carbon locations or data centers *PUE*<sup>6</sup> dashboards to help their customer design low-impact cloud environments.

In the interviews, only *P8* has worked on a migration where ecological efficiency was considered:

*“[...] there was another issue, an ecological one. For them, it was the most important to take into account. [...] We were not told why; we suspect that it may be related to their brand image or requirements of their shareholders.”*

It affected the cloud environment’s selection: resources located in ecological data centers (powered by renewable or nuclear energies, efficient cooling) and resources’ collocation to reduce network usage. According to them, cloud providers still provide too few indicators to eco-design a cloud system.

## 6 Discussion

### 6.1 Connections with Related Work

*Migration Motivations*. As mentioned in Sect. 2, some previous works established lists of motivations for migrating to cloud computing. Some may also be understood as drivers for selecting the cloud environment. All drivers from cited works

---

<sup>6</sup> Power Usage Effectiveness: a standard measure of how efficiently a data center uses energy.

**Table 3.** Comparison of selection drivers between the study and related works

<i>This study</i>	Jamshidi <i>et al.</i> [10]	Nussbaumer <i>et al.</i> [13]	Bremer <i>et al.</i> [3]
<b>CMR1</b>	“Maintenability”		
<b>CMR2</b>	“Cost saving”	“Cost”	“Cost reduction”
<b>CMR3</b>		“Reliability”	
<b>CMR4</b>	“Elasticity to fluctuation”	“Performance”	
<b>CMR5</b>			“Easiness of use”, “Technical understanding of Cloud Computing”
<b>CMR6</b>			
<b>CMR7</b>		“Security”	“Data security concerns”
<b>CMR8</b>	“Interoperability”		“Dependency on provider”
<b>CMR9</b>			
<b>CMR10</b>			“Reduced set-up time”
<b>CMR11</b>			“Transparency in regard to service”
			“Easy possibility to test services before purchase”
			“Service & Support”

and this study have been listed in Table 3 and annotated whether we consider them as motivations or selection drivers.

The drivers listed as motivations are related to either general characteristics of clouds (e.g., “Scalability”, “Focus on the main business”) or characteristics of companies that might benefit from cloud computing (e.g., “Competitive pressure”, “Organizational size and structure”).

As we can see, the interviews highlighted most selection drivers from the literature, plus two new drivers: *Digital sovereignty* (**CMR9**) and *Ecological efficiency* (**CMR11**). In addition, the driver *Adherence to the cloud provider* (**CMR8**) was addressed only in one of its aspects: the related works documented requirements towards reducing this adherence, while the interviews pointed out that some organizations desire to use (or not use) a cloud provider in particular.

Requirements towards “Service & Support” do not appear in our study. Two reasons can explain this observation. First, the leading cloud providers’ services are now considered well-documented and have a high level of support. This driver, thus, has not appeared in the interviews. Second, the interviewees, being consultants, acted as support during the migration. Support requirements were then satisfied by the qualities of the consultancy delivery rather than those of the cloud services. Their role may also explain why none spoke about “Easy possibility to test services before purchase” because, as cloud experts, they have a comprehensive knowledge of the services and the underlying mechanisms. Other types of participants may have highlighted it. The current state of cloud computing in the industry might explain why “Transparency regarding service” was not mentioned because, except for some *IaaS*, major cloud providers give little to no clue about the ways resources are delivered.

*Migration Methodologies and Cloud Environment Selection Approaches*. Most reviewed approaches for migrating a system and selecting a cloud environment do not indicate how to evaluate their suitability, given the migration requirements. While none participant claimed to have followed any of these approaches, we might draw some associations between them and the identified requirements families.

Both *CloudRecommender* and *COCA-PT* aim at minimizing the costs of the environment, which unambiguously matches **CMR2**. Quinton’s approach assumes pre-made decisions about software, much like Zhang’s methodology. This looks like reported migrations with no software adaptations, so they seem to fit migrations considering **CMR9** or **CMR10**.

## 6.2 Requirements Classification

After having extracted cloud migration drivers from the interviews, it could be interesting to classify them. Indeed, requirements classifications help engineers and business analysts elicit and analyze requirements.

We first attempted several classifications: one based on the *Goal-Oriented Requirements Engineering* distinction between functional and non-functional requirements, one that distinguishes hard and soft goals, and a last with the kind of services used to meet the requirements (*IaaS*, ...). None was relevant to classifying our *CMRs*.

In the following paragraphs, we present two classifications that we consider relevant.

First, one can consider the *Triple Constraints* of project management. Constraints are: *cost*, *delay*, and *quality*. Of course, these constraints interact with each other, e.g., increasing the quality also increases either the cost or the delay. As shown in Table 4, our high-level requirements can be sorted into three categories: the requirements that minimize costs, those that reduce the delay, and those that increase quality. As in project management, satisfying some requirements of one category can lead to less satisfying other categories. For instance, reducing operation efforts (**CMR1**) is implemented by using *PaaS* and, most of the time, involves software adaptations, which conflicts with the need to minimize the migration duration (**CMR10**).

**Table 4.** Project management classification

Category	Cloud Migration Drivers
Cost	<b>CMR1, CMR2, CMR5, CMR9</b>
Delay	<b>CMR10</b>
Quality	<b>CMR3, CMR4, CMR6, CMR7, CMR8, CMR11</b>

One can also consider classification based on the DevOps paradigm (development activities and operation activities). If this dichotomy is interesting, we must also consider the characteristics of cloud services that, if well selected, can satisfy some requirements. In Table 5, the requirements are classified according to these three categories. The *need of new skills* (**CMR5**) requirements affect both software development (e.g., knowledge about a programming language) and operations skills (e.g., serverless architecture monitoring). The *adherence to the cloud provider* (**CMR8**) one comes in two versions: either aiming at the system agnosticism to the cloud provider, which results in little to no development at the cost of operations, or accepting vendor lock-in with the desire to use highly managed services, leading to software adaptations and fewer operations.

**Table 5.** Team’s activities classification

Category	Cloud Migration Drivers
Development	<b>CMR2, CMR5, CMR8</b>
Operations	<b>CMR1, CMR3, CMR5, CMR8, CMR9, CMR10</b>
Cloud characteristics	<b>CMR4, CMR6, CMR7, CMR11</b>

## 7 Conclusion

To better understand the drivers of public cloud environment selection during migrations, we conducted a qualitative interviews study with industry cloud experts. The study resulted in eleven cloud migration drivers that impact the choice of cloud providers, cloud services, and cloud services configuration. We have retrieved most drivers known in the scientific literature with this approach and found new ones.

We also provide two classifications for these requirements to support requirements elicitation and analysis in migration projects.

There are several threats to the validity of this study. We followed a standardized approach for leading the interviews, not for their analysis. Moreover, the corpus is tiny. However, we noticed that only one new high-level requirement emerged from discussions after the 6th interview. We, therefore, believe the study to be close to saturation. Finally, the interviewees are similar in their position as consultants, which may lead to missing some requirements.

Future works should address several questions. We reviewed many migration approaches, but there miss a method to determine which one is tailored for a given migration. We think this method would be based on the drivers identified in this paper. In addition, the identified drivers may lead to conflicting requirements, which must be addressed through trade-offs. Yet, it is hard to make the best trade-offs (e.g., what impact on the cloud environment’s cost for slightly decreased performance?). A methodological framework should be developed to handle this issue.

## References

1. Andrikopoulos, V., Binz, T., Leymann, F., Strauch, S.: How to adapt applications for the cloud environment - challenges and solutions in migrating applications to the cloud. *Computing* **95**(6), 493–535 (2013)
2. Binz, T., Leymann, F., Schumm, D.: Cmotion: a framework for migration of applications into and between clouds. In: *IEEE International Conference on Service-Oriented Computing and Applications*, pp. 1–4. IEEE Computer Society (2011)
3. Bremer, M., Walter, T., Fjodorovs, N., Schmid, K.: A systematic literature review on the suitability of cloud migration methods for small and medium-sized enterprises. ESN: 2701-6277 (2021)
4. Chen, A.J., Boudreau, M., Watson, R.T.: Information systems and ecological sustainability. *J. Syst. Inf. Technol.* **10**(3), 186–201 (2008)
5. DiCicco-Bloom, B., Crabtree, B.: The qualitative research interview. *Med. Educ.* **40**, 314–21 (2006). <https://doi.org/10.1111/j.1365-2929.2006.02418.x>
6. Fahmideh, M., Grundy, J., Beydoun, G., Zowghi, D., Susilo, W., Mougouei, D.: A model-driven approach to reengineering processes in cloud computing. *Inf. Softw. Technol.* **144**, 106795 (2022)
7. Frey, S., Hasselbring, W.: Model-based migration of legacy software systems into the cloud: the cloudmig approach. *Softwaretechnik-Trends* **30**(2), 84–85 (2010)
8. Gartner: Migrating applications to the cloud: Rehost, refactor, revise, rebuild, or replace? (2010). <https://www.gartner.com/en/documents/1485116>. Accessed 07 July 2022
9. Gholami, M.F., Daneshgar, F., Low, G., Beydoun, G.: Cloud migration process - a survey, evaluation framework, and open challenges. *J. Syst. Softw.* **120**, 31–69 (2016)
10. Jamshidi, P., Ahmad, A., Pahl, C.: Cloud migration research: a systematic review. *IEEE Trans. Cloud Comput.* **1**(2), 142–157 (2013)
11. McCracken, G.: *The Long Interview. Qualitative Research Methods*, SAGE Publications (1988). <https://books.google.fr/books?id=3N01cl2gtoMC>
12. Mell, P., Grance, T.: *The NIST Definition of Cloud Computing*. Technical report, 800-145, National Institute of Standards and Technology (NIST), Gaithersburg, MD (2011)
13. Nussbaumer, N., Liu, X.: Cloud migration for SMEs in a service oriented approach. In: *IEEE 37th Annual Computer Software and Applications Conference, Workshops*, pp. 457–462. IEEE Computer Society (2013)
14. ACM SIGSOFT: Qualitative surveys (interview studies) (2022). <https://github.com/acmsigsoft/EmpiricalStandards/blob/master/docs/QualitativeSurveys.md>. Accessed 07 July 2022
15. Silva, G.C., Rose, L.M., Calinescu, R.: A systematic review of cloud lock-in solutions. In: *IEEE 5th International Conference on Cloud Computing Technology and Science*, vol. 2, pp. 363–368. IEEE Computer Society (2013)
16. Zhang, W., Berre, A., Roman, D., Huru, H.: Migrating legacy applications to the service cloud. In: *ACM Object-Oriented Programming, Systems, Languages and Applications (OOPSLA)* (2009)
17. Zhao, J., Zhou, J.: Strategies and methods for cloud migration. *Int. J. Autom. Comput.* **11**(2), 143–152 (2014)