Future Generation Computer Systems 28 (2012) 66–77

Contents lists available at SciVerse ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

OPTIMIS: A holistic approach to cloud service provisioning

Ana Juan Ferrer [a], Francisco Hernández [b], Johan Tordsson [b,*], Erik Elmroth [b], Ahmed Ali-Eldin [b], Csilla Zsigri [c], Raül Sirvent [d], Jordi Guitart [d], Rosa M. Badia [d], Karim Djemame [e], Wolfgang Ziegler [f], Theo Dimitrakos [g], Srijith K. Nair [g], George Kousiouris [h], Kleopatra Konstanteli [h], Theodora Varvarigou [h], Benoit Hudzia [i], Alexander Kipp [j], Stefan Wesner [j], Marcelo Corrales [k], Nikolaus Forgó [k], Tabassum Sharif [l], Craig Sheridan [l]

[a] *Atos Origin, Barcelona, Spain*
[b] *Department of Computing Science, Umeå University, Sweden*
[c] *The 451 Group, London, UK*
[d] *Barcelona Supercomputing Center, Barcelona, Spain*
[e] *School of Computing,University of Leeds, UK*
[f] *Fraunhofer – SCAI, Sankt Augustin, Germany*
[g] *British Telecom, London, UK*
[h] *National Technical University of Athens, Greece*
[i] *SAP Research Belfast, UK*
[j] *High Performance Computing Center Stuttgart, Germany*
[k] *Institut für Rechtsinformatik, Leibniz Universität Hannover, Germany*
[l] *Flexiant Limited, Livingston, UK*

ARTICLE INFO

Article history:
Received 20 January 2011
Received in revised form
9 May 2011
Accepted 28 May 2011
Available online 21 July 2011

ABSTRACT

We present fundamental challenges for scalable and dependable service platforms and architectures that enable flexible and dynamic provisioning of cloud services. Our findings are incorporated in a toolkit targeting the cloud service and infrastructure providers. The innovations behind the toolkit are aimed at optimizing the whole service life cycle, including service construction, deployment, and operation, on a basis of aspects such as trust, risk, eco-efficiency and cost. Notably, adaptive self-preservation is crucial to meet predicted and unforeseen changes in resource requirements. By addressing the whole service life cycle, taking into account several cloud architectures, and by taking a holistic approach to sustainable service provisioning, the toolkit aims to provide a foundation for a reliable, sustainable, and trustful cloud computing industry.

1. Introduction

Contemporary cloud computing solutions, both research projects and commercial products, have mainly worked on functionalities closer to the infrastructure, such as improved performance for virtualization of computing, storage, and network resources, as well as fundamental issues such as virtual machine (VM) migrations and server consolidation. When higher-level concerns are considered, existing solutions tend to focus only on functional aspects, whereas quality factors, although very important, are typically not considered. In order to move from a basic cloud service infrastructure to a broader cloud service ecosystem, there is a great need for tools that support higher-level concerns and non-functional aspects in a comprehensive manner.

In this work we introduce five higher-level challenges that in our view must be addressed for a wider adoption of cloud computing:

1. Service life cycle optimization.
2. Dependable sociability = Trust + Risk + Eco + Cost.
3. Adaptive self-preservation.

* Corresponding author.
E-mail addresses: ana.juanf@atosresearch.eu (A.J. Ferrer), hernandf@cs.umu.se (F. Hernández), tordsson@cs.umu.se (J. Tordsson), elmroth@cs.umu.se (E. Elmroth), ahmeda@cs.umu.se (A. Ali-Eldin), csilla.zsigri@the451group.com (C. Zsigri), Raul.Sirvent@bsc.es (R. Sirvent), jordi.guitart@bsc.es (J. Guitart), rosa.m.badia@bsc.es (R.M. Badia), karim@comp.leeds.ac.uk (K. Djemame), wolfgang.ziegler@scai.fraunhofer.de (W. Ziegler), theo.dimitrakos@bt.com (T. Dimitrakos), srijith.nair@bt.com (S.K. Nair), gkousiou@telecom.ntua.gr (G. Kousiouris), kkonst@telecom.ntua.gr (K. Konstanteli), dora@telecom.ntua.gr (T. Varvarigou), benoit.hudzia@sap.com (B. Hudzia), kipp@hlrs.de (A. Kipp), wesner@hlrs.de (S. Wesner), corrales@iri.uni-hannover.de (M. Corrales), nikolaus.forgo@iri.uni-hannover.de (N. Forgó), tsharif@flexiant.com (T. Sharif), csheridan@flexiant.com (C. Sheridan).

0167-739X/$ – see front matter
doi:10.1016/j.future.2011.05.022

4. Multi-cloud architectures.
5. Market and legislative issues.

Notably, these five concerns cover most of the ten key obstacles to growth of cloud computing identified in a recent report [1], as well as address several open issues [2–4]. When approaching these concerns, we focus on a holistic approach to cloud service provisioning and argue that a single abstraction for multiple coexisting cloud architectures is imperative for a broader cloud service ecosystem. Thus, our work is based on the assumptions that clouds will be available as private and public, that they will be used in isolation or in a variety of conceptually different combinations, and that they will be internal or external to individual organizations or cross-organizational consortia. The outcome of our multidisciplinary research within these five challenges are incorporated in the *OPTIMIS Toolkit*. We present the design of the toolkit and discuss how it addresses the higher-level concerns introduced here.

The main stakeholders throughout this work are service providers and infrastructure providers, although it can be foreseen that our results can also impact actors such as brokers, and service consumers (end-users). Henceforth, we consider the following definitions for these roles:

- *Service Providers (SPs)* offer economically efficient services using hardware resources provisioned by infrastructure providers. The services are directly accessed by end-users or orchestrated by other SPs.
- *Infrastructure Providers (IPs)* offer computing, storage, and network resources required for hosting services. Their goal is to maximize their profit from tenants by making efficient use of their infrastructures, possibly by outsourcing partial workloads to partnering providers.

The element of interaction between SPs and IPs is a service. Notably, SPs and IPs have conflicting economical and performance goals that result in interesting problems in cases where they are both part of the same organization. It is foreseen that both types of providers are in need of more feature-rich analysis and management tools in order to provide economically and ecologically sustainable services throughout the whole service life cycle.

The outline of the paper is as follows. Sections 2–6 present details of the five higher-level concerns enumerated above. Section 7 presents a high-level view of the OPTIMIS Toolkit, discusses how the toolkit addresses the five challenges, and illustrates how it can be used to instantiate various cloud architectures. Experimental results are described in Section 8. Finally, we share our concluding remarks in Section 9.

2. Service life cycle optimization

There are three fundamental steps in the service life cycle, construction of the service, deployment of the service to an IP, and operation of the service.

2.1. Service construction

In the service construction phase, the SP builds the service and sets it up for deployment and operation on the IP. The activities performed include preparation and configuration of VM's images as well as specification of dependences among service components. Currently, there is no programming model specifically tailored for clouds. On the one hand developers are limited to use application-specific platforms [5], restrictive computing paradigms [6,7], or platforms for a single cloud middleware [8]. A common way of offering these solutions is by wrapping them as a PaaS environment or by offering proprietary APIs for particular middlewares limited to single infrastructure providers. On the other hand, developing high-level services from raw infrastructure through use of IaaS is a manual and ad-hoc process, hindering broader cloud adoption as service development becomes expensive and time consuming.

The challenge of service construction resides in designing and developing easy ways to create complex services [9]. To this end, applications need to be abstracted from their execution environment and the development of new services, including those composed from adapting and combining legacy-and licensed software, must be facilitated. For the latter, novel license management technologies are required to significantly extend currently available solutions for management of license tokens in distributed environments [10]. The composition of services as a mix of software developed in-house, existing third-party services, and license-protected software is a clear contrast to commonly used approaches for service composition [11].

2.2. Service deployment

In the service deployment phase, the service is placed on an IP for operation. From the SP point of view, the main objective during this phase is to select the most suitable IP for hosting a service, whereas for IPs the main objective is to decide whether accepting a new service is beneficial for its business goals. The key process during deployment is the negotiation of SLA terms between SP and IP. However, this negotiation is performed manually in current deployment solutions [2] and SPs are limited to use single providers as differences in contextualization mechanisms [12,13], necessary for instantiating a service once deployed, hinder multi-cloud deployment.

Moreover, contemporary cloud SLA mechanisms [14,15] are typically limited to cost-performance tradeoffs. For example, it is not possible to automatically evaluate levels of trust and risk, or to negotiate use of license-protected software. To overcome current limitations, deployment optimization tools need to support deployment given a set of policies and allow SPs to specify required SLA terms for services. The policies governing deployment must include the degree of trust expected from a provider, the level of risk with regard to cost thresholds, energy consumption limits, performance levels, etc.

2.3. Service execution

Service execution is the last phase in the service life cycle and consists of two different but related procedures, performed by SPs and IPs. The overall objectives of these stakeholders differ and as a result there is a conflict of interest in management tasks. On one hand, the SP performs a set of management operations in order to meet the high-level Business Level Objectives (BLOs) specified during service construction. These include, for instance, constant monitoring of service status and mechanisms for monitoring and continuous assessment of the risk level of IPs in order to apply the corresponding corrective actions. On the other hand, an IP performs autonomic actions to, e.g., consolidate and redistribute service workloads, replicate and redistribute data sets, and trigger actions to increase and decrease capacity to adhere to SLAs, i.e., enact elasticity rules, with the overall goal of achieving the most efficient use of its infrastructure and hence maximize its own objectives, potentially at the expense of the goals of the SPs.

Contemporary tools for service execution optimization focus on mechanisms for monitoring service status and for triggering capacity variations to meet elasticity requirements [16]. These tools tend to use only SLAs and infrastructure status for making decisions and either neglect business-level parameters such as risk, trust, reliability, and eco-efficiency, or consider them in isolation. For instance, eco-efficient policies for the operation of hosting centers aiming to minimize its power consumption have

been investigated [17,18]. Similarly, trust mechanisms have been studied in the context of grid resource selection in order to choose providers that are likely to provide better service according to their reputation [19]. In the same way, risk information has been used for decision making [20]. Finally, some resource management proposals for data centers and e-commerce systems are driven by business objectives or incorporate business level parameters in their management policies [21–23], though most of them target revenue as the only objective.

According to this, SPs and IPs require software components that in addition to the traditional performance indicators also take into account business-level parameters (e.g., risk, trust, reliability, etc.) in order to make decisions in a synergistic fashion to contribute to the overall provider goals. In order to achieve this type of decision making process, all management activities must be harmonized through the use of cloud governance processes that integrate all service requirements, from high-level BLOs to infrastructure requirements.

3. Dependable sociability = trust + risk + eco + cost

Traditionally, relationships between stakeholders have been focused on cost-performance trade-offs. However, economical factors are not enough for an open and highly dynamic environment in which relationships are created in an on–off basis with a possible high degree of anonymity between stakeholders. On the one hand, it is necessary to offer methods and tools to quantitatively assess and evaluate stakeholders, e.g., through audit and monitoring functions to analyze the probability of service failure, the risk of data loss, and other types of SLA violations. On the other hand methods to measure stakeholder satisfaction such as individual and group perceptions, reputation [24] of stakeholders regarding ecological aspects, or previous experiences, must also be considered. Altogether, these mechanisms can be used to confirm the dependability and reliability among stakeholders. We now introduce various quality factors that have traditionally not been considered but that we believe improve the decision making capabilities of both SPs and IPs.

3.1. Trust—reputation management

Trust is a multifaceted aspect not only related to risk and security aspects, but also to perceptions and previous experiences. Selection of an IP depends on the trust that it will provision the service correctly and securely. Conversely, knowing a customer's reputation enhances the admission control evaluation and reduces the risk of breaking economical or ecological goals of an IP.

Trust is often calculated by reputation mechanisms [25,26]. A reputation is a subjective measure of the perception that members of a social network has of one another. This perception is based on past experiences. The reputation ranks aggregate experiences of all members of the social network—in this case the social network includes all stakeholders, i.e., the combination of SPs and IPs. Tools to determine the integrity of data disclosed by stakeholders as well as mechanisms to act accordingly, e.g., to blacklist dishonest providers, are also necessary.

3.2. Risk assessment

Underpinning a successful cloud infrastructure is delivering the required QoS levels to its users in a way that minimizes risk, which is measured in terms of a combination of the likelihood of an event and its impact on the provision of a functionality. Earlier work in risk management for distributed systems has mainly focused on operational aspects such as failures and performance degradation, and assumed a very IaaS centric view under a specific resource reservation model [27]. Factors such as trust, security, energy consumption, and cost have not been traditionally considered. Moreover, tools for the definition, assessment and management of risk based on variations in levels of the proposed factors for both stakeholders and throughout the service life cycle (SPs during service construction, deployment, and operation; and IPs during admission control and internal operations) are virtually nonexistent. Such risk management mechanisms must also consider aspects such as energy consumption, the cost of reconfiguration and migration, and the reliability and dependability of the provided services, in order to maintain secure, cost-effective, and energy-efficient operations.

3.3. Green assessment

Environmental concerns reflected in upcoming legislation have increased the awareness of the ecological impact of the ICT industry. As a result, the level of ecological awareness can now be a deciding factor between competing providers. However, environmental concerns are not the only reason for the growing interest in green data centers, rising electricity prices can also guide the deployment of services to locations in which they are provisioned in a more efficient way. The consequence is that IPs must now focus more than ever on energy efficiency aspects.

Cloud computing in itself contributes to reduce power consumption by consolidating workloads from different customers in a smaller number of physical nodes, turning off unused nodes [28]. The key concern to confront is the tradeoff between performance and power consumption [29–31], i.e., to minimize power consumption but still accomplish the desired QoS [32]. To address this tradeoff, energy efficiency must be treated equal as the other critical parameters, already including service availability, reliability, and performance. To this end, a broad set of mechanisms are required, ranging from tools for logging and assessing the ecological impact at the service level, theoretical models that characterize the power consumption of services depending on configuration parameters (e.g., clock frequency, resource usage, and number of threads used), to predicting future energy impact based on run-time state, historical usage patterns, and estimates of future demands.

3.4. Cost and economical sustainability

Fulfilling high trust levels between stakeholders, reduced risk, and eco-efficient provisioning is trivial if cost is not an issue. Economical aspects are necessary to balance the previous three goals. Furthermore, cost must be an explicit parameter throughout the entire service life cycle. Current commercial providers offer a variety of capabilities under different pricing schemes, but it is hard to differentiate among the offerings without sufficient knowledge of the repercussions on internal performance, ecologic, and economic goals. To improve this situation, more complex economic models are needed. These models must include features to compare economical repercussions between alternative configurations. To this end, such models must employ business related terms that can be translated to service and infrastructure parameters during development and deployment of services. During the operation of a service, it is necessary to optimize economical factors through a combination of runtime monitoring, analysis of historical usage patterns, and predictions of future events. The latter helps to anticipate future service economic trends. All of these actions create an economic policy framework in which stakeholders can specify the autonomic behavior expected from the elements under their management responsibility.

4. Adaptive self-preservation

Service and infrastructure management in clouds is difficult due to the ever-growing complexity and inherent variability in

services. Rapid responses to events are necessary to satisfy agreed SLAs. Accordingly, human administration becomes unfeasible and building self-managed systems seems to be the only way to succeed.

Although self-management for cloud infrastructures is a novel research area, approaches for automated adjustment of resource allocations for virtualized hosting centers can be applied [33,34]. However, many of these approaches have two main limitations. First, they typically exhibit a lack of expressiveness in self-management due to a lack of a holistic view of management. This results in management actions, e.g., resource allocation, monitoring, or data placement, that are performed in isolation and are as such not optimal. Second, existing solutions tend to use only SLAs and infrastructure status for making decisions, neglecting business-level parameters such as risk, trust, reliability, and eco-efficiency, or considering them in isolation, as discussed in Section 2.3.

To overcome this problem, SPs and IPs require that all management actions are harmonized by overarching policies that incorporate, balance, and synergize aspects of risk assessment, trust management, eco-efficiency, as well as economic plausibility. The management actions must be handled by software components able to monitor and assess their own status and adapt their behavior to ever changing conditions. Aspects to consider in this decision are overall BLOs, infrastructure capabilities, historical usage patterns, and predictions of future demands. The result must be an integrated solution capable of a wide range of autonomic management tasks [35] including self-configuration, i.e., automatic configuration of components, self-healing, i.e., automatic discovery and correction of faults, and self-optimization, i.e., automatic optimization of resource allotments and data placement. Example autonomic management tasks include SLA enforcement, recovery of service operation upon resource failure, VM placement optimization (including migration [36]), enactment of elasticity policies (vertical and horizontal scalability), consolidation of services, management of advance reservations, and data replication for fault tolerance or performance improvements.

5. Multi-cloud architectures

There are at least two fundamentally different architectural models for cloud service provisioning using multiple external clouds:

- In a *federated cloud*, an IP can sub-contract capacity from other providers as well as offer spare capacity to a federation of IPs. Parts of a service can be placed on remote providers for improved elasticity and fault tolerance, but the initial IP is solely responsible for guaranteeing the agreed upon SLA (with respect to performance, cost, eco-efficiency, etc.) to the SP.
- In a *multi-provider hosting* scenario, the SP is responsible for the multi-cloud provisioning of the services. Thus, the SP contacts the possible IPs, negotiates terms of use, deploys services, monitors their operation, and potentially migrates services (or parts thereof) from misbehaving IPs. IPs are managed independently and placement on different providers is treated as multiple instances of deployment.

Each model has benefits and drawbacks. However, to date, these models have only been studied in isolation [13,16,37], which essentially creates either–or situations. Instead, for a more flexible provisioning model, it is important to be able to use multiple clouds without distinguishing whether a service is hosted within a single cloud or across multiple providers, i.e., clouds must be able to be combined into arbitrary, hierarchical architectures. To this end, it is imperative to create a single abstraction without regard of architectural style. To accomplish this, there are a number of challenges that must be solved, including: verification of SLA adherence; metering, accounting and billing of services running out of a provider's boundaries; managing software license authorizations, particularly when migrating a service to different providers; replication, synchronization, and backup of data between providers; evaluation of economical efficiency associated with using external providers; legal implications regarding data protection and privacy aspects; and establishing an inter-cloud security context for governing all interactions between clouds.

6. Market and legislative issues

Clouds bring change to user behavior. The focus of attention is moving away from how a service is implemented or hosted to what the service offers, a shift from buying tools that enable a functionality to contracting third-party services that deliver this functionality on demand in a pay-per-use model [38]. There is a massive surge in interest around private and hybrid clouds. With new application use cases emerging on a regular basis, numerous commercial on-ramps are seeking to provide access to multiple clouds, and startups and incumbent providers alike are targeting cloud service brokerage. These changes in the landscape create opportunities for new roles, relationships, and value activities.

We believe that the hybrid cloud is where the market is heading. The appetite among enterprises for a range of execution environments to serve the needs of different workloads reinforces that successful cloud strategies will enable the *best execution venue* practices supported in hybrid cloud environments. These allow service providers to choose, via policy automation, different venues (private, public, federated, etc. clouds) in which to run workloads, depending on costs, latency, security, locality, eco-efficiency, or other SLA requirements. In short, we work toward a cloud market model where providers are fungible, transparent and compliant, and consumers can easily and efficiently use cloud functionality to their best advantage.

In addition to the new market opportunities, the emerging cloud landscape introduces additional concerns related to legal compliance. It is important to assess from the very beginning those associated legal risks in cloud computing and create a framework for minimizing or mitigating those risks, particularly when presupposing that data moves geographically. In such cases, data protection and privacy, being issues of cross-border jurisdictional nature as they concern the acquisition, location, and transfer of data [39], are important and call for a data protection framework and security infrastructure [40,41]. Furthermore, legally and non-legally binding guidelines concerning green IT strategies and legislative and jurisdictional issues are key to infrastructure and service providers when it comes to decision making [42,43]. In addition, intellectual property and contractual issues concerning ownership and rights in information and services located in the cloud need to be tailored and taken into account when designing a cloud computing toolkit [44].

7. The OPTIMIS toolkit

Our response to the challenges presented in the previous section is a multi-disciplinary research line with inclusion of the main outcomes in the OPTIMIS Toolkit. The toolkit consists of a set of fundamental components realizing an anticipating a variety of architectures for simultaneous use of multiple clouds. Fig. 1 illustrates the high-level components of the toolkit: the *Service Builder (SB)*, the *Basic Toolkit*, the *Admission Controller (AC)*, the *Deployment Engine (DE)*, the *Service Optimizer (SO)*, and the *Cloud Optimizer (CO)*.

The SB is used during the service construction phase and enables developed services to be delivered as Software as a

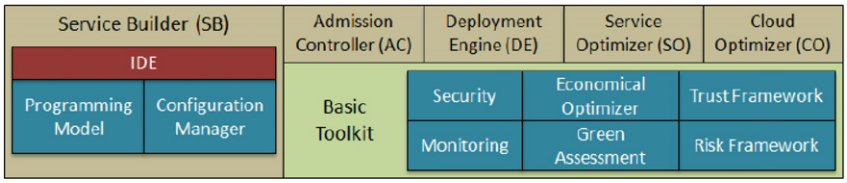


Fig. 1. High level view of the components in the OPTIMIS toolkit. The basic toolkit addresses quantitative and qualitative analyzes that help in making optimal decisions regardless of the invoking component. Organizations can act as SPs and/or IPs depending on the components that they chose to adopt (from the admission controller, deployment engine, service optimizer, and cloud optimizer).

Service (SaaS). A service programmer has access to an integrated development environment that simplifies both development and configuration of the service using a novel programming model for service development. In our programming model, a service is a collection of *core elements*, that include services built from source code, existing services, licensed software, and legacy-software not developed specifically for clouds, as well as a set of dependences between these core elements. During operation of the service, the core elements are orchestrated by a runtime environment that analyzes the dependences defined during service construction.

Each core element has a set of functional and non-functional requirements associated, e.g., requested performance, amount of physical memory and CPU characteristics, response time, elasticity aspects, service and security level policies, ecological profile, etc. In addition, there are also requirements among the core elements and between the service and its potential users. Once the core elements are implemented, these are packed by the SB along with any external software components into VM images. The configuration of these VM images are then encoded in a service manifest based on the Open Virtualization Format (OVF) [45] with a set of extensions to specify the functional and non-functional requirements of the service. This service manifest is the input to the service deployment phase, and the completion of the manifest marks the end of the service construction phase.

The Basic Toolkit provides functionalities common to components that are used during service deployment and execution. Some of the functionalities address the quantitative and qualitative requirements that we in Section 3 discuss under the term Dependable Sociability. Monitoring and security are functionalities that must be considered during several stages of the service life cycle. The Basic Toolkit provide general purpose functionalities that evaluate similar aspects of management. However, component behavior is customized depending on the invoking module. This customization is fulfilled through the use of internal policies that adapt the decision making processes, e.g., based on the invoking component and the current stage of the service life cycle.

For example, to create a comprehensive trustworthy system, the toolkit considers all relationships of the types SP–IP and IP–IP. Trust estimations are determined from the trust rank of the SP (or IP) in other members of its own social network according to a transitive trust chain. The reputation mechanism delivers trust measurements at two levels: for IPs, trust reflects their performance and the ability to accomplish promised levels of service, whereas for SPs, trust measurements are relevant for establishing successful business networks. For IPs, the trust tools include methods to identify SPs in long term relationships and to analyze SPs' historical behavior that can help to improve management operations by e.g., predicting future capacity.

We now illustrate how the OPTIMIS Toolkit is used during deployment of services. The first scenario is a simplified one where an IP delivers capacity to a SP as shown in Fig. 2. More complex scenarios can also be realized as described later in the section. Service deployment starts from where service construction ends, with a service manifest that describes the VM images and associated requirements for service configuration.

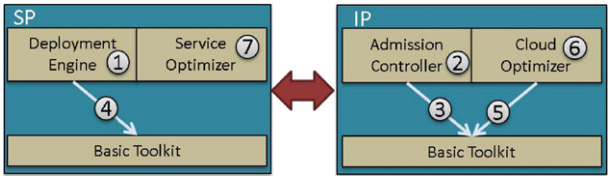


Fig. 2. Service deployment scenario that illustrates the interaction between the high-level components of the SP and IP.

During service deployment the SP finds, by use of the DE, the best possible conditions for operation of the service, negotiates terms of deployment, and launches the service in the IP. The DE send the service manifest (SLA template) to the IP to receive deployment offers (*Step* 1 in Fig. 2).

An IP receiving a deployment request performs a probabilistic admission control to decide whether to admit the new service or not (*Step* 2). This test balances revenue maximization lead by business goals against penalties for misbehavior, i.e., from breaking SLAs of currently provisioned services. Example policies include *over-provisioning* (overbooking for revenue optimization) as well as *under-provisioning* (reserving capacity to minimize the risk of failures). The test is carried out by the Admission Controller (AC) component with help of use of the Basic Toolkit (*Step* 3). An integral part of the admission control test is workload analysis of the current infrastructure and the new service, to be performed in combination with capacity planning.

Using the Basic Toolkit, the DE evaluates the IP's offers to run the service in order to chose the most suitable one (*Step* 4). This analysis is carried out considering both qualitative and quantitative factors discussed in Section 3. After selecting a deployment offer, the DE prepares the service images for deployment. In this step, information required for the VMs to be able to self-contextualize once they boot is embedded in ISO images that are bundled together with the VM images.

In the IP, the process of accepting a new service starts by allocating space for the VMs and determining their initial placement. The latter is a complicated process as placement must consider the (predicted) elasticity of the service and the non-functional constraints specified in the deployment manifest. Some of these constraints can even have legal ramifications regarding e.g., data protection and privacy or environmental guidelines as discussed in Section 6. Allocation of resources for the service is performed by the CO with help of components for management of VMs and data that both make extensive use of the functionalities in the Basic Toolkit (*Step* 5). Once resources are allocated by the CO with help of these managers, the VM images are booted and self-contextualized with help of previously the scripts installed (*Step* 6). The SO in the SP is notified once the deployment process completes (*Step* 7).

The SO and CO also perform repeated management decisions during service operation, the SO on behalf of the SP and the CO for the IP. The SO continuously checks that the IP provisions the service according to the agreed SLAs, otherwise the SO can migrate the service to a different IP. On the other hand, the CO optimizes the IP's infrastructure resources. This includes, for instance, monitoring

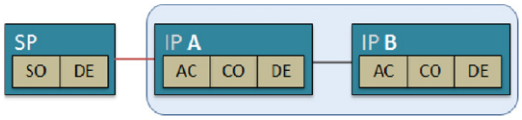


Fig. 3. Federated cloud architecture where the SP establishes a contract with IP A that is a member of the federation that includes IP B. The service is delivered using resources of either IP, or from both.

of infrastructure status, recovery from failures (e.g., by using checkpoints), and mechanisms for optimizing power consumption (e.g., by consolidating and migrating VMs). A complicating factor in this infrastructure optimization is that the CO also dynamically increases and decreases the number of VMs for a service (i.e., performing elasticity management) to protect SLAs with SPs to avoid penalties and preserve reputation. The SO and CO both utilize the Basic Toolkit as the basis for their quantitative and qualitative analysis.

7.1. Flexible multi-cloud architectures

The combination of the five main components of the OPTIMIS Toolkit and their implementation by SPs and IPs, gives rise to a number of plausible multi-cloud scenarios where resources from more than one IP can be combined in novel ways. Some example compositions follow.

Federated architecture

In this scenario (Fig. 3), several IPs (A and B) use the OPTIMIS Toolkit to establish a cooperation in which any IP can lease capacity from the other. The cooperation is carried out according to internal IP business policies. The SP is unaware of this federation as its contract is with a single IP (in this case IP A). However, the SP can indirectly pose constraints on which IPs in the federation that can be used through non-functional requirements such as affinity of service components or juridical restrictions to prevent VM migration across country borders (data protection areas). In the federated scenario, the contracted provider (IP A) is fully responsible toward the SP even in the case of subcontracting of resources from the federation.

Multi-cloud architecture

In this scenario (Fig. 4), the SP is responsible for the multi-cloud aspect of service operation. If IP A does not fulfill the agreed objectives, the SP can cancel the contract and move the service to a different IP (IP B). Notably, the SP is responsible both for negotiating with each IP and for monitoring the IPs during service operation. In more complex variations of this scenario, parts of the service can be hosted on multiple providers. By using APIs and adapters externally to the OPTIMIS components, the toolkit can also achieve interoperability with non-OPTIMIS providers (IP C in Fig. 4). However, in such cases, the SP has to resort to less feature-rich management capabilities, and the risk levels for service provisioning increase accordingly.

Aggregation of resources by a third party broker

This scenario, illustrated in Fig. 5, introduces a new stakeholder, the broker, which aggregates resources from multiple IPs and offers these to SPs. The broker thus acts as a SP to IPs and as an IP to SPs. Given the conflicting goals of these respective providers, there are many interesting concerns regarding the independence, honesty, and integrity of the broker. Benefits with this model for SPs include simplicity and potential cost reductions, as the broker can provide a single entry point to multiple IPs and may obtain better prices due to bulk discounts from IPs. Management is simplified for an IP that offers capacity to a broker as the number of customers is decreased. Accordingly, trust and risk become easier to predict as the IP is likely to have fewer and longer term contracts

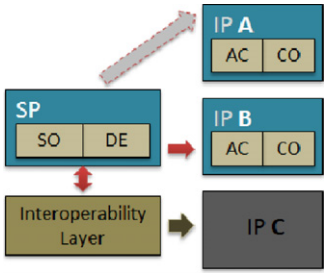


Fig. 4. Multi-cloud architecture in which the SP terminates the contract with IP A and re-deploys the service to IP B. The SP can also use infrastructure from an IP (C) that does not implement the OPTIMIS Toolkit. In this case the SP uses an interoperability layer that is external to the OPTIMIS components.

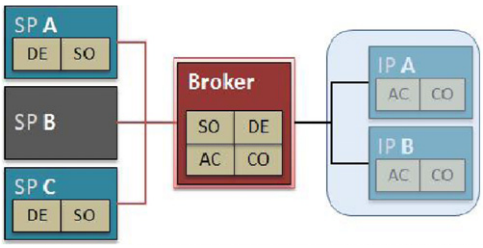


Fig. 5. In the brokering architecture, a third party broker aggregates resources from several IPs (A and B) and offer these resources to SPs.

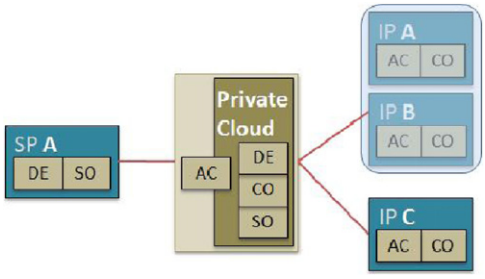


Fig. 6. Hybrid cloud architecture. An organization moves part of its operation to external providers (the federation formed by IPs A and B, as well as C). The organization can also sell capacity to the SP during periods of low load.

with brokers, instead of a multitude of short interactions with potentially unknown SPs.

Hybrid cloud architecture

In this scenario (Fig. 6), any organization that operates a private cloud is able to externalize workloads to public IPs. This is accomplished by monitoring the normal operation of the private cloud, through the CO component, and using capacity from public clouds (IPs A, B, and C, the former two in federation) when the local infrastructure is insufficient. The implementation of this scenario is significantly simplified once an organization is using the OPTIMIS Toolkit for managing its private cloud. Furthermore, this scenario can be extended if the organization makes use of the AC. In this case, the organization is able to offer capacity from its private cloud to others (SP A) when that capacity is not needed for internal operations.

8. Evaluation

This section describes experimental evaluations of selected parts of the OPTIMIS Toolkit, namely how cost and risk can be included in elasticity policies and how to evaluate cloud providers through risk assessment.

Table 1
Comparison of elasticity policies.

Policy	Failed	Cost	Failed (%)	Cost (%)
UR-DR	−17 829	349 648	−0.11	2.2
UR-DP$_{C1}$	−54 296	196 790	−0.34	1.24
UR-DP$_{C2}$	−5 813	566 795	−0.037	3.6

8.1. Elasticity policies for cost-risk tradeoffs

The key aspect for elasticity is to allocate and deallocate the right number of VMs to a service, and at the right time. Over-provisioning, i.e., allocating too many VMs, reduces the risk for (availability) SLA violations, but increases costs as some VMs are not needed. Conversely, allocating too few VMs reduces costs but increases the risk for failing to provision the service in accordance with the SLA. An ideal elasticity policy should be able to proactively and accurately estimate the future service load and thus be able to reduce the SLA violation rate, with a minimal amount of over-provisioning. For elasticity, we define a risk asset to the number of service requests the IP failed to provision proactively. We define the cost to be the number of VMs over-provisioned at any time unit i.e., resources allocated but not used.

An elasticity policy defines two decision points, one for when and how much to scale up and another one for scale down. We here study three different elasticity policies, all based on closed loop control systems. The first and simplest policy scales up and down reactively and is henceforth denoted *UR-DR*. The second policy, denoted *UR-DP*$_{C1}$, scales up reactively, but scales down using a proactive controller, where the gain parameter is based on the periodical change of the system load. Similarly, the *UR-DP*$_{C2}$ policy combines reactive scale up with proactive scale down, where the gain parameter is the ratio between the load change and the average system service rate over time. A more in-depth discussion of control approaches to elasticity is found in our previous work [46].

We evaluate the three elasticity policies using a 17 days and 19 h subset of the web server traces from Wikipedia [47]. In this evaluation we assume that each VM can service 100 requests simultaneously and that load balancing is perfect. The number of VMs suggested by the elasticity policy is thus a fraction of the number of requests anticipated for the Wikipedia site. A summary of the experiments is shown in Table 1. In this table, the Failed column shows the number of VMs needed (but not allocated) to serve all requests, i.e., the degree of under-provisioning. The Cost column shows the number of VMs allocated but not used, i.e., the amount of over-provisioning. In the two right-most columns these two metrics are given as percentages of the total number of VMs.

Fig. 7 shows each service request (marked '+') in Wikipedia workload and the allocated capacity (marked '×'), i.e., number of VMs multiplied by 100. Fig. 7(a), (c) and (e), show the complete 17 days 19 h trace, whereas Fig. 7(b), (d), and (f) detail the number of service requests and allocated VMs for a period of 17 min. As shown in Table 1 and illustrated in Fig. 7, the three policies result in different risk levels associated with different costs. The *UR*−*DC*1 policy provides the lowest cost combined with the highest risk. The *UR-DR* policy results in a medium risk level coupled with a medium cost, whereas *UR-DP*$_{C2}$ provides the lowest risk but at the highest cost. The general trend observable here is that when doubling the over-provisioning from 1.24% to 2.2% the risk is lowered to one third, and when further increasing over-provisioning to 3.6%, risk is cut to around a tenth. We remark that for the Wikipedia traces, both risk and cost are very low, with less than 0.4% under-provisioning and less than 4% resources over-provisioned by any policy. The good results are due to this workload being rather smooth and regular with the exception of a sharp peak at around 600,000 s.

To further understand how risk and cost vary with the workload size, we performed a second set of experiments where the number of service requests varies. In this experiment, we multiply the number of requests per time unit in the Wikipedia traces by a factor from 1 to 10 and by 20, 30, and 40. Figs. 8 and 9 illustrate how the risk and cost, both defined as in the previous experiment, change with the workload size. The *UR-DR* and *UR-DP*$_{C1}$ policies show similar behavior, namely that when the workload size increases, the risk (under-provisioning) increases up to around 0.8% whereas the cost (over-provisioning) decreases to around 1%. On the contrary, the *UR-DP*$_{C2}$ shows a stable behavior, with risk around 0.04% and cost just above 3.5% for all workload sizes. These results suggest that the *UR-DP*$_{C2}$ elasticity policy is more robust and should be used for services with unknown workloads for which low-risk provisioning is desired.

8.2. SP and IP evaluation through risk assessment

The OPTIMIS risk assessor provides the functionality to evaluate providers (*SP*/*IP*) based on a number of criteria. These evaluations are core parts in provider decision making, e.g., for service deployment and admission control. The SP uses the following criteria:

- Past performance: Record with respect to SLA acceptance and violation rate in past SLAs.
- Maintenance: Based on the IP's policy for maintaining their infrastructure.
- Infrastructure: Based on available resources, fault-tolerant mechanisms available, use of redundant network, etc.
- Security: Based on the IP's security policy regarding access to resources.
- Customer support: The IP's policy with regards to customer support, e.g. do they have a 24/7 contact number?

On the other hand, the IP uses the following criteria for assessing an SP:

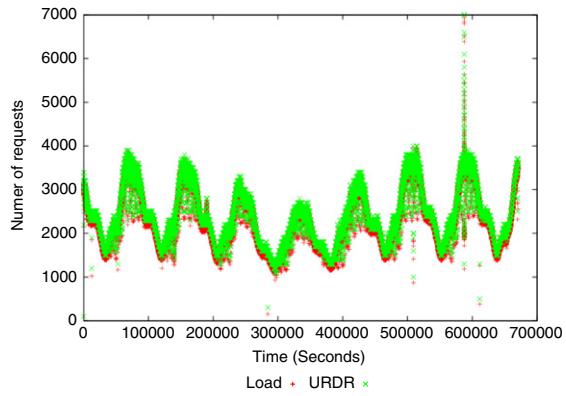
- Past performance: Record with respect to SLA acceptance and violation rate in past SLAs.
- Legal: Based on the SP's policy for supporting legal aspects.
- Security: Based on the SP's security policy regarding the use of resources.

A value between 0 and 1 is computed for each of the criterion by evaluating a provider with respect to a number of sub-criteria. These values are used as the basis for the evaluation. There are two important features of the evaluation system. First, it takes into account provider preferences. Different providers are likely to value the criteria differently. Therefore providers are able to specify the importance of each of the criteria on a scale of 0 to 10. These are then translated into criteria weights that encapsulate the amount of influence a particular criterion should have and incorporated into the provider evaluation. Second, it is able to handle missing data. Some providers may be unwilling to share all of the information necessary to compute the criteria values. Alternatively, data may have been corrupted.

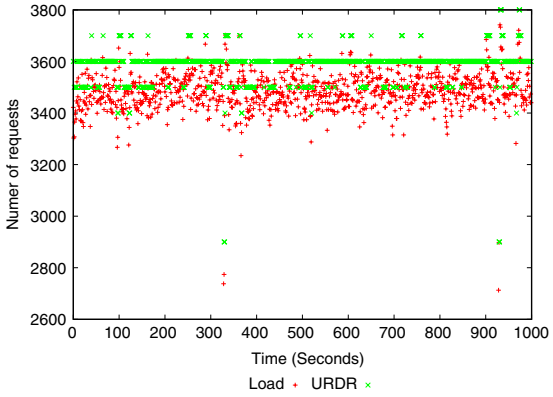
The assessment is achieved through an implementation of Dempster–Shafer Analytical Hierarchy Process (DS-AHP), whereby each decision alternative (in this case each provider) is mapped onto a belief and plausibility interval [48].

Consider a set of providers as corresponding to the proposition that the providers in that set are preferable to all other providers considered in the evaluation but not to each other. The end-user preference weights, w_i, are computed for each criterion, $i = 1 \ldots N$. Pair-wise comparison of decision alternatives (for providers) are used to derive weights for the criteria, $r_j^{(i)}$ for the ith criterion and jth provider. A weight or Basic Probability Assignment (BPA) is computed for each provider as:

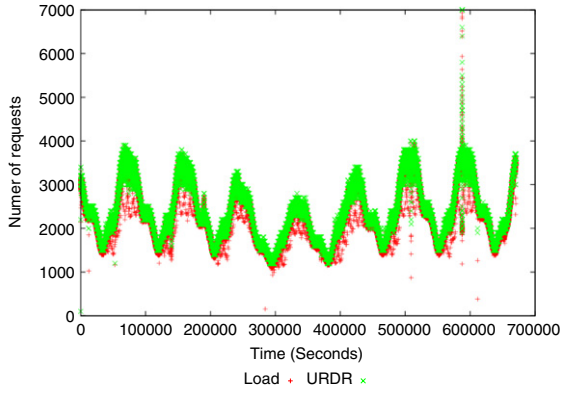
$$m_j = \sum_i w_i r_j^{(i)}.$$



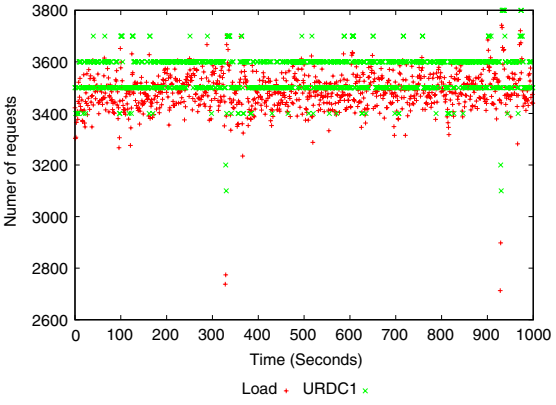
(a) *UR-DR* performance over 7 days and 19 h.



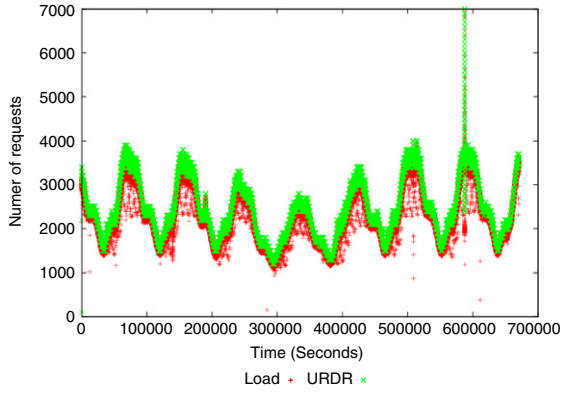
(b) UR-DR: zooming on a period of 17 min.



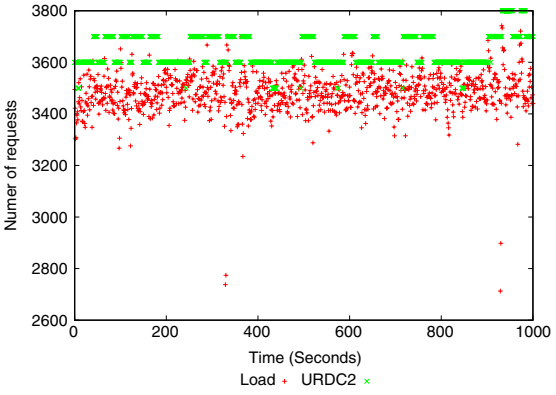
(c) *UR-DPC*$_1$ performance over 7 days and 19 h.



(d) *UR-DP*$_{C1}$ zooming on a period of 17 min.



(e) *UR-DP*$_{C2}$ performance over 7 days and 19 h.



(f) *UR-DP*$_{C2}$ zooming on a period of 17 min.

Fig. 7. Performance of elasticity policies.

If data is missing $r_j^{(i)}$ values cannot be computed. Instead of pair-wise comparison, for any given criterion, each decision alternative (provider) is evaluated relative to the *frame of discernment* (the entire decision space). Providers which are indistinguishable with respect to a criterion are grouped together as a single proposition (that the providers in this group are the best alternative). This results in the BPAs in the form $m_i(s)$ where s is a set of one or more providers. Criteria BPAs are combined using Dempster's rule of combination:

$$(m_1 \oplus m_2)(y) = \frac{\sum_{s_1 \cap s_2 = y} m_1(s_1) m_2(s_2)}{1 - \sum_{s_1 \cap s_2 = \emptyset} m_1(s_1) m_2(s_2)},$$

where $\emptyset$ is the empty set.

Belief in the proposition A is defined as the exact belief that either A or some subset of A is true:

$$Bel(A) = \sum_{\mathbf{B} \subseteq \mathbf{A}} m(B). \quad (1)$$

Here, $m(B)$ is a basic probability assignment for the proposition B [48]. The plausibility of proposition A is a measure of the extent to which we do not disbelieve proposition A and is defined as:

$$Pls(A) = \sum_{\mathbf{A} \cap \mathbf{B} \neq \emptyset} m(B). \quad (2)$$

These form an interval, [Bel(A), Pls(A)] with respect to proposition A. These intervals, for each proposition corresponding to a single provider, are used to compute the order of preference for

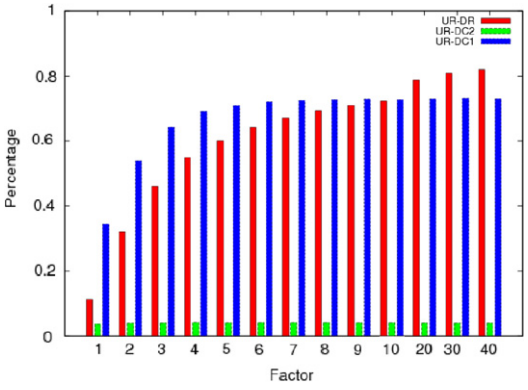


Fig. 8. Effect on risk of varying workload sizes and elasticity policies.

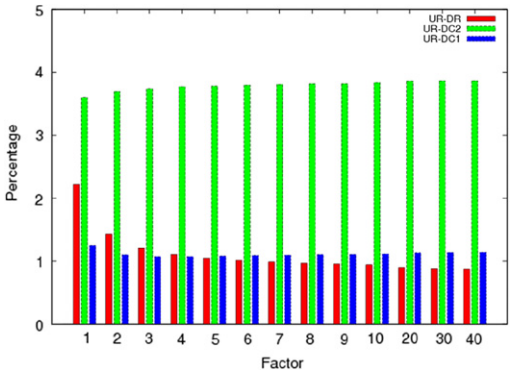


Fig. 9. Effect on cost of varying workload sizes and elasticity policies.

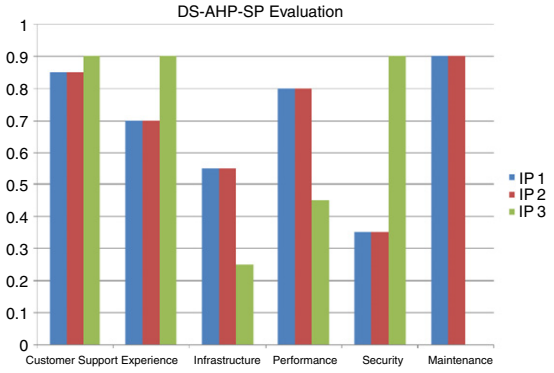


Fig. 10. SP-DS-AHP assessment of three IPs.

the providers. A preference value for provider A relative to provider B is computed as:

$$P(A > B) = \frac{max[0, Pls(A) - Bel(B)] - max[0, Bel(A) - Pls(B)]}{[Pls(A) - Bel(A)] + [Pls(B) - Bel(B)]}. \quad (3)$$

If $P(A > B) > 0.5$ then provider A is preferred. Comparison with simulated data is used in order to obtain a provider ranking between 0 and 1, where 1 corresponds to better than all providers, 0 corresponds to worse than all providers, and 0.5 is average.

Preference values are derived to directly compare the providers. Fig. 10 shows SP's assessment with rating results 0.66, 0.33, and 1 for providers IP_1, IP_2, and IP_3 respectively. Provider 3 is regarded as preferable to both IP_1 and IP_2 with certainty since belief in IP_3 as the best choice is greater than the plausibility of either IP_1 or IP_2. Hence the computed ranking is IP_3, IP_1, and IP_2. Fig. 11 shows IP's assessment with rating results 0.33, 1, and 0.66 for providers SP_A, SP_B, and SP_C respectively, which leads to the final ranking: SP_B, SP_C, SP_A. More details on DS-AHP are found in [48].

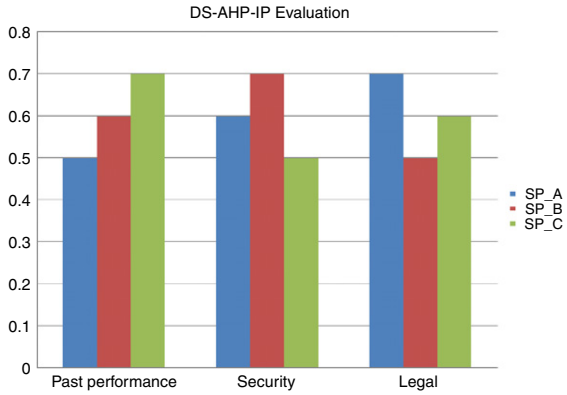


Fig. 11. IP-DS-AHP assessment of three SPs.

9. Concluding remarks

We present five fundamental challenges for wide adoption of cloud computing: service life cycle optimization, dependable sociability, adaptive self-preservation, multi-cloud architectures, and market and legislative issues. We believe that addressing these concerns as a whole is key to boost the delivery of advanced services. Our approach is two-fold, we perform fundamental research in a wide range of areas spanning from VM management and programming languages to business models and IT law to address these challenges, and we incorporate our findings in the development of the OPTIMIS Toolkit.

The focus of the toolkit is on cloud service and infrastructure optimization throughout the service life cycle: construction, deployment, and operation of services. All management actions in the toolkit are harmonized by overarching policies that consider trust and risk assessment to comply with economical and ecological objectives without compromising operational efficiencies. Assessing risk of economical and ecological parameters is a unique, albeit challenging, goal. Governance processes and policies are defined to harmonize management activities throughout the service life cycle.

The purpose of self-service tools is to enable developers to enhance services with non-functional requirements regarding allocation of data and VMs, as well as aspects related to elasticity, energy consumption, risk, cost, and trust. The OPTIMIS Toolkit incorporates risk aspects in all phases of the service life cycle and uses trust assessment tools to improve decision making in the matching of SPs and IPs. The ecological impact of service provisioning is integrated in all relevant decision making. The toolkit also ensures that the desired levels of risk, trust, or eco-efficiency are balanced against cost, to avoid solutions that are unacceptable from an economical perspective. The OPTIMIS tools are aimed to enable SPs and IPs to perform monitoring and automated management of services and infrastructures, so as to compare different alternative configurations in terms of business efficiency. Notably, mechanisms required to design policies that fulfill legislative and regulatory constraints are also taken incorporated in the toolkit, e.g., to address adoption challenges from regulatory and standards compliance requirements such as privacy and data protection.

Our goal is also to enable and simplify the creation of a variety of provisioning models for cloud computing, including cloud bursting, multi-cloud provisioning, and federation of clouds. Provisioning on multi-clouds architectures and federated cloud providers facilitates novel and complex composition of clouds that

considerably extend the limited support for utilizing resources from multiple providers in a transparent, interoperable, and architecture independent fashion.

Acknowledgments

We acknowledge the anonymous reviewers for their insightful feedback. Financial support for this work is provided by the European Commission's Seventh Framework Programme ([FP7/2001-2013]) under grant agreement number 257115, OPTIMIS.

References

[1] M. Armbrust, A. Fox, R. Griffith, A.D. Joseph, R.H. Katz, A. Konwinski, G. Lee, D.A. Patterson, A. Rabkin, I. Stoica, M. Zaharia, Above the clouds: a Berkeley view of cloud computing, in: Technical report, Electrical Engineering and Computer Sciences, University of California at Berkeley, 2009. Technical Report No. UCB/EECS-2009-28.

[2] R. Buyya, C.S. Yeo, S. Venugopal, J. Broberg, I. Brandic, Cloud computing and emerging it platforms: vision, hype, and reality for delivering computing as the 5th utility, Future Generation Computer Systems 25 (6) (2009) 599–616.

[3] Luis Rodero-Merino, Luis M. Vaquero, Victor Gil, Fermín Galán, Javier Fontán, Rubén S. Montero, Ignacio M. Llorente, From infrastructure delivery to service management in clouds, Future Generation Computer Systems 26 (8) (2010) 1226–1240.

[4] L. Vaquero, L. Rodero-Merino, J. Caceres, M. Lindner, A break in the clouds: towards a cloud definition, SIGCOMM Computer Communications Review 39 (1) (2008) 50–55.

[5] Google. Google App Engine, Visited May 2011, code.google.com/appengine.

[6] J. Dean, S. Ghemawat, Mapreduce: simplified data processing on large clusters, Communications of the ACM 51 (1) (2008) 107–113.

[7] C. Vecchiola, X. Chu, R. Buyya, Aneka: a software platform for. NET-based cloud computing, in: High Speed and Large Scale Scientific Computing, IOS Press, 2009, pp. 267–295.

[8] Microsoft. Windows Azure platform. Visited May 2011, www.microsoft.com/windowsazure.

[9] Gabor Kecskemeti, Gabor Terstyanszky, Peter Kacsuk, Zsolt N'emeth, An approach for virtual appliance distribution for service deployment, Future Generation Computer Systems 27 (3) (2011) 280–289.

[10] J. Li, O. Wäldrich, W. Ziegler, Towards SLA-based software licenses and license management in grid computing, in: Proceedings of the CoreGRID Symposium, Springer Verlag, 2008, pp. 139–152.

[11] D. Jordan, J. Evdemon (chairs), Web services business process execution language version 2.0, 2007. Visited June 2010, http://docs.oasis-open.org/wsbpel/2.0/wsbpel-v2.0.pdf.

[12] K. Keahey, T. Freeman, Contextualization: providing one-click virtual clusters, in: Fourth IEEE International Conference on eScience, IEEE, 2008, pp. 301–308.

[13] K. Keahey, M. Tsugawa, A. Matsunaga, J. Fortes, Sky computing, IEEE Internet Computing 13 (5) (2009) 43–51.

[14] A. Andrieux, K. Czajkowski, A. Dan, K. Keahey, H. Ludwig, T. Nakata, J. Pruyne, J. Rofrano, S. Tuecke, M. Xu, Web services agreement specification (WS-agreement). May 2011, www.ogf.org/documents/GFD.107.pdf, Visited.

[15] M. Comuzzi, C. Kotsokalis, G. Spanoudakis, R. Yahyapour, Establishing and monitoring SLAs in complex service based systems, in: Proceedings of the 2009 IEEE International Conference on Web Services, IEEE Computer Society, 2009, pp. 783–790.

[16] B. Rochwerger, D. Breitgand, E. Levy, A. Galis, K. Nagin, I. Llorente, R. Montero, Y. Wolfsthal, E. Elmroth, J. Caceres, M. Ben-Yehuda, W. Emmerich, F. Galán, The reservoir model and architecture for open federated cloud computing, IBM Journal of Research and Development 53 (4) (2009) 1–11.

[17] J.S. Chase, D.C. Anderson, P.N. Thakar, A.M. Vahdat, R.P. Doyle, Managing energy and server resources in hosting centers, ACM SIGOPS Operating Systems Review 35 (5) (2001) 103–116.

[18] G. Chen, W. He, J. Liu, S. Nath, L. Rigas, L. Xiao, F. Zhao, Energy-aware server provisioning and load dispatching for connection-intensive internet services, in: 5th USENIX Symposium on Networked Systems Design and Implementation, NSDI'08, ACM, 2008, pp. 337–350.

[19] B.K. Alunkal, I. Veljkovic, G.V. Laszewski, K. Amin, Reputation-based grid resource selection, in: Workshop on Adaptive Grid Middleware, IEEE, 2003.

[20] K. Djemame, I. Gourlay, J. Padgett, G. Birkenheuer, M. Hovestadt, O. Kao, K. Voss, Introducing risk management into the grid, in: 2nd IEEE International Conference on e-Science and Grid Computing, e-Science'06, IEEE, 2006.

[21] M. Buco, R. Chang, L. Luan, C. Ward, J. Wolf, P. Yu, Utility computing SLA management based upon business objectives, IBM Systems Journal 43 (1) (2004) 159–178.

[22] D. Gilat, A. Landau, A. Sela, Autonomic self-optimization according to business objectives, in: 1st International Conference on Autonomic Computing, ICAC 2004, IEEE, 2004, pp. 206–213.

[23] A. McCloskey, B. Simmons, H. Lutfiyya, Policy-based dynamic provisioning in data centers based on slas, business rules and business objectives, in: IEEE/IFIP Network Operations and Management Symposium, NOMS'08, IEEE, 2008, pp. 903–906.

[24] Tian Chunqi, Baijian Yang, R2Trust, a reputation and risk based trust management framework for large-scale, fully decentralized overlay networks, Future Generation Computer Systems (March) (2011).

[25] A. Jøsang, R. Ismail, C. Boyd, A survey of trust and reputation systems for online service provision, Decision Support Systems 43 (2007) 618–644.

[26] Yining Liu, Keqiu Li, Yingwei Jin, Yong Zhang, Wenyu Qu, A novel reputation computation model based on subjective logic for mobile ad hoc networks, Future Generation Computer Systems 27 (5) (2011) 547–554.

[27] K. Djemame, I. Gourlay, J. Padgett, K. Voss, O. Kao, Risk Management in Grids, in: R. Buyya, K. Bubendorfer (Eds.), Market-Oriented Grid and Utility Computing, Wiley, 2009, pp. 335–353.

[28] L. Liu, H. Wang, X. Liu, X. Jin, W.B. He, Q.B. Wang, Y. Chen, GreenCloud: a new architecture for green data center, in: Proceedings of the 6th International Conference on Autonomic Computing, ACM, 2009, pp. 29–38.

[29] Y. Chen, A. Das, W. Qin, A. Sivasubramaniam, Q. Wang, N. Gautam, Managing server energy and operational costs in hosting centers, ACM SIGMETRICS Performance Evaluation Review 33 (1) (2005) 303–314.

[30] J. Kephart, H. Chan, R. Das, D. Levine, G. Tesauro, F. Rawson, C. Lefurgy, Coordinating multiple autonomic managers to achieve specified power-performance tradeoffs, in: 4th International Conference on Autonomic Computing, ICAC 2007, IEEE, 2007.

[31] Dang Minh Quan, Federico Mezza, Domenico Sannenli, Raffaele Giafreda, T-Alloc: a practical energy efficient resource allocation algorithm for traditional data centers, Future Generation Computer Systems (May) (2011).

[32] Damien Borgetto, Henri Casanova, Georges Da Costa, Jean-Marc Pierson, Energy-aware service allocation, Future Generation Computer Systems (May) (2011).

[33] P. Padala, K.G. Shin, X. Zhu, M. Uysal, Z. Wang, S. Singhal, A. Merchant, K. Salem, Adaptive control of virtualized resources in utility computing environments, ACM SIGOPS Operating Systems Review 41 (3) (2007) 289–302.

[34] Y. Song, Y. Sun, H. Wang, X. Song, An adaptive resource flowing scheme amongst VMs in a VM-based utility computing, in: 7th IEEE International Conference on Computer and Information Technology, CIT 2007, IEEE, 2007, pp. 1053–1058.

[35] J. Kephart, D. Chess, The vision of autonomic computing, Computer 36 (1) (2003) 41–50.

[36] Tiago C. Ferreto, Marco A.S. Netto, Rodrigo N. Calheiros, Csar A.F. De Rose, Server consolidation with migration control for virtualized data centers, Future Generation Computer Systems (May) (2011).

[37] J. Tordsson, R.S. Montero, R.M. Vozmediano, I.M. Llorente, Cloud brokering mechanisms for optimized placement of virtual machines across multiple providers, 2010, in press (doi:10.1016/j.future.2011.07.003).

[38] W. Fellows, IT is cloud: to infrastructure and beyond, The 451 Group – CloudScape (June) (2010).

[39] H. Grant, T. Finlayson, Cloud computing & data protection. March 2009. Visited May 2011, http://www.twobirds.com/English/News/Articles/Pages/Cloud_Computing_Data_Protection_030609.Aspx.

[40] Art. 29 data protection working party. February 2010, http://ec.europa.eu/justice/policies/privacy/workinggroup/index_en.htm, Visited May 2011.

[41] P. Hustinx, Data protection and cloud computing under EU law, in: Third European Cyber Security Awareness Day, BSA, European Parliament, April 2010, pp. 1–7.

[42] The challenge of energy efficiency through information and communication technologies, February 2009. Parliament resolution of 4 February 2009. visited July 2010, http://www.europarl.europa.eu/sides/getDoc.do?pubRef=-//EP//TEXT+TA+P6-TA-2009-0044+0+DOC+XML+V0//EN.

[43] Computer OECD committee for information and communications policy. Recommendation of the council on information and communication technologies and the environment (OECD guidelines) — C(2010)61, April 2010. Visited May 2011, http://webnet.oecd.org/oecdacts/Instruments/ShowInstrumentView.aspx?InstrumentID=259.

[44] Y. Poullet, J. Van Gyseghem, J. Gerard, C. Gayrel, J. Moiny, Cloud Computing and its implications on data protection, chapter discussion paper prepared by the Research Centre on IT and Law — CRID, page 8. Council of Europe, March 2010. Visited May 2011, www.coe.int/t/dghl/cooperation/economiccrime/cybercrime/Documents/Reports-Presentations/2079_reps_IF10_yvespoullet1b.pdf.

[45] DMTF. Open Virtualization Format (OVF). Visited May 2011, http://www.dmtf.org/standards/ovf.

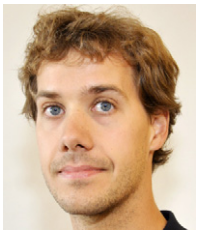
[46] A. Ali-Eldin, J. Tordsson, E. Elmroth, An adaptive hybrid elasticity controller for cloud infrastructures, 2011 (submitted for publication).

[47] G. Urdaneta, G. Pierre, M. van Steen, Wikipedia workload analysis for decentralized hosting, Elsevier Computer Networks 53 (11) (2009) 1830–1845.

[48] Z. Hua, B. Gong, X. Xu, A DS-AHP approach for multi-attribute decision making problem with incomplete information, Journal of Expert Systems with Applications 34 (2008) 2221–2227.

Ana Juan Ferrer has an Engineering Degree in Computer Software from Universitat Autònoma de Barcelona (1998). In Atos Origin since 2006, in Atos Research and Innovation, currently she is Head of Lab of the Service Engineering & IT Platforms Lab, group that focus its research in Cloud Computing, Service Engineering and Green IT. From June 2010 she manages the OPTIMIS project, investigating platforms and architectures for escalable and trustful Cloud services platforms. She also participates in NUBA, focusing in research on IaaS Cloud federation models. In the past, she has worked in NEXOF-RA, definition of the NESSI Open Framework Roadmap; and BEinGRID, industrial application of Cloud and Grid, and Crosswork project. Before Atos Origin she has wide experience as consultant and software architect in the Internet environment and e-business. Ana currently leads the Green IT working group in INES (Spanish S&S initiative), and participates in the ATOS Origin Scientific Community.

Francisco Hernández is an Assistant Professor in grid and cloud computing in the Department of Computing Science at Umeå University, Sweden. His research interests are in the areas of distributed systems and high performance computing. He holds a Ph.D. in Computer Science from the University of Alabama, Birmingham. He received the Graduate School Dean's Award for excellence in graduate studies at the Ph.D. level for the year 2006. Francisco currently serves as the co-chair of the Virtualized Service Platform Collaboration Workgroup (VSP CWG). He also holds a BS in Systems Engineering, from Universidad Francisco Marroquín, Guatemala.



Johan Tordsson is an Assistant Professor in Computer Science at Umeå University. Tordsson's research interests include several topics related to autonomic resource management for grid and cloud computing infrastructures, e.g., scheduling, elasticity, and quality of service. Academic background includes M.Sc., Lic.Eng, and Ph.D. Degrees from Umeå University and a postdoctoral visit at Universidad Complutense de Madrid. His current research activities include being the lead architect for the OPTIMIS Project (FP7 IP).

Erik Elmroth is a Professor and the Head of the Department of Computing Science at Umeå University. He is leading the Umeå University research on grid and cloud computing, including the group's participation in eSSENCE, the three EU FP7 IP-projects RESERVOIR, OPTIMIS, and VISION Cloud, and the UMIT research laboratory. Previous appointment highlights include an year at the NERSC, the Lawrence Berkeley National Laboratory, a semester at the Massachusetts Institute of Technology (MIT) as well as several years for Swedish and international research councils and governmental strategy bodies.

Ahmed Ali-Eldin is a Ph.D. student in the Cloud and Grid computing research group in Umeå University since October, 2010. He obtained his M.Sc. degree in Communication and Information Technology from Nile University, Egypt while working as a research assistant there (2008–2010). Ali-Eldin also holds a B.Sc. in Computer and Systems Engineering from Ain-Shams university, Egypt. His current research interest is self-* for clouds.

Csilla Zsigri is an economist with international work experience. Currently, she is the EU Business Development & Program Manager at The 451 Group (technology-industry analyst company). Csilla joined The 451 Group to extend business activities in Europe with special regard to EU research activities. Csilla also covers EU research trends, project outcomes and commercial opportunities for the 451 EURO service.

Previously, Csilla worked as a business consultant for Atos Origin's Research and Innovation unit in Barcelona, coordinating business and marketing activities in numerous European ICT Research and Technology Development projects. Before that, she did business development, consultancy work and in-company trainings for ConAction, a Hungarian management consulting company. Her focus was on seeking new business opportunities for the firm, as well as leading projects from launching new services on the local market to setting up new companies abroad. Csilla also has experience working at Nokia, being involved in a turnkey project for Vodafone, where she had to deal with key market players from the telecommunications industry.

Csilla holds an Economist M.Sc. degree specialized in Business and Management Consulting.

Raül Sirvent has a Ph.D. Degree in Computer Science at the Computer Architecture Department (UPC, 2009). He has been involved in research activities at the European Center of Parallelism of Barcelona (CEPBA) from 2002 to 2005. Since 2005, he has been holding a permanent position at the Barcelona Supercomputing Center inside the Computer Sciences department (Grid Computing and Clusters). His main research interests are related to high performance computing, grid and cloud programming models and tools, automatic workflow generation, fault tolerance mechanisms and the use of semantics for scheduling. He has been involved in FP6 projects CoreGRID, BEinGRID, and BREIN, the Spanish initiative "Red Temática para la Coordinación de Actividades Middleware en Grid" and one of the founders of the GRID superscalar project. Currently he is working in the Spanish project NUBA which focuses on the federation of cloud infrastructures, and the FP7 project OPTIMIS, which aims at optimizing cloud services taking trust, risk, eco-efficiency, cost and legal issues into account.

Jordi Guitart received the MS and Ph.D. Degrees in Computer Science at the Technical University of Catalonia (UPC), in 1999 and 2005, respectively. Currently, he is an associate professor at the Computer Architecture Department of the UPC and an associate researcher at Barcelona Supercomputing Center (B.SC.) within the Autonomic Systems and eBusiness Platforms research line. His research interests are oriented toward innovative resource management approaches for modern distributed computing systems. He is involved in a number of European projects.

Rosa M. Badia has a Ph.D. in Computer Science (1994) from the Technical University of Catalonia (UPC). She is a scientific researcher from the Consejo Superior de Investigaciones Científicas (CSIC) and manager of the Grid Computing and Cluster research group at the Barcelona Supercomputing Center (B.SC.). She was involved in teaching and research activities at the UPC from 1989 to 2008, where she has been an Associate Professor since 1997. From 1999 to 2005, she was involved in research and development activities at the European Center of Parallelism of Barcelona (CEPBA). Her current research interests are performance prediction and modeling of MPI programs and programming models for complex platforms (from multicore to the grid/cloud). She has published more than 100 papers in international conferences and journals on the topics of her research. She has participated in several European projects and is currently participating in projects NUBA (at Spanish level), TERAFLUX, TEXT, SIENA, OPTIMIS, VENUS-C and ScalaLife and is a member of HiPEAC2 NoE.

Karim Djemame (Co-Investigator) was awarded a Ph.D. at the University of Glasgow, UK, in 1999, and is currently holding a Senior Lecturer position at the School of Computing, University of Leeds. He sits on a number of international programme committees for grid/cloud middleware, computer networks and performance evaluation. He was the investigator of various e-Science/grid projects including DAME, BROADEN, and AssesGrid. His main research areas focus on grid/cloud computing, including system architectures, resource management, and risk assessment. Dr. Djemame is a member of the IEEE.

Wolfgang Ziegler is the head of the Grid and Cloud Middleware Research Group of the Department of Bioinformatics. His research areas are grid and cloud computing, resource management and scheduling, management of virtual organizations, and service oriented architectures. He has been the working group co-chair of the Open Grid Forum for more than 10 years. Currently, he is the OGF area director for applications and co-chairs the Grid Resource Allocation Agreement Protocol Working Group (GRAAP-WG), which develops the OGF standard WS-Agreement and WS-Agreement Negotiation. He is participating in several national and European grid and cloud projects as work-package leader, where SLAs and license management in distributed computing environments play a major role, e.g. D-Grid, PHOSPHORUS, SmartLM, OPTIMIS.

Theo Dimitrakos

Srijith K. Nair is a Senior Security Researcher at BT Innovate & Design, the R&D part of BT where he looks at fundamental security challenges facing enterprise level infrastructures, including, but not limited to security aspects of cloud based services. He has experience in managing cross organizational teams, having served as a work package leader in European Union funded ICT project OPTIMIS. He was also part of multiple expert groups in the security industry that, among others, provide advice to the European Network and the Information Security Agency (ENISA). He has a Ph.D. in Computer Science from Vrije Universiteit, Amsterdam and has published several peer-reviewed papers in international journals, conferences and workshops and has also served on the program committee of several international conferences and journals. He is a member of the ACM and the IEEE.

George Kousiouris received his diploma in Electrical and Computer Engineering from the University of Patras, Greece in 2005. He is currently pursuing his Ph.D. in grid and cloud computing at the Telecommunications Laboratory of the Dept. of Electrical and Computer Engineering of the National Technical University of Athens and is a researcher for the Institute of Communication and Computer Systems (ICCS). He has participated in the EU funded projects BEinGRID IRMOS and OPTIMIS and the national project GRID-APP. In the past, he has worked for private telecommunications companies. His interests are mainly computational intelligence, optimization, computer networks and web services.

Kleopatra Konstanteli received her diploma in Electrical and Computer Engineering in 2004 from the National Technical University of Athens (NTUA). In 2007, she received a Master's Degree in Techno-economical Systems from the NTUA in cooperation with the University of Athens and the University of Piraeus. She is currently pursuing her doctoral-level research in Computer Science, and at the same time, working as a research associate in the Telecommunications Laboratory of Electrical and Computer Engineering of NTUA and participating in EU funded projects. Her research interests are mainly focused on the field of distributed computing.

Theodora Varvarigou received the B. Tech Degree from the National Technical University of Athens, Athens, Greece in 1988, the MS Degrees in Electrical Engineering (1989) and Computer Science (1991) from Stanford University, Stanford, California in 1989 and the Ph.D. Degree from Stanford University as well in 1991. She worked at AT&T Bell Labs, Holmdel, New Jersey between 1991 and 1995. Between 1995 and 1997, she worked as an Assistant Professor at the Technical University of Crete, Chania, Greece. In 1997, she was elected as an Assistant Professor, and since 2007, she has been a Professor at the National Technical University of Athens, and the Director of the Postgraduate Course "Engineering Economics Systems". Prof. Varvarigou has great experience in the area of semantic web technologies, scheduling over distributed platforms, embedded systems and grid computing. She has published more than 150 papers in leading journals and conferences in this area.

Benoit Hudzia is a Senior Researcher at SAP Research, CEC Belfast. He is leading the UK funded project Virtex on virtualization technologies, participates in the EU-funded cloud computing project, Reservoir, and is involved in the SAP internet of the services framework research program. He received a Ph.D. from the University College, Dublin in the field of parallel and distributed computing, focussing on P2P and grid systems. During his studies in Electronics and Computer Science Engineering at the University of Paris 6 (M.Sc.) and the engineering school EFREI (Paris) (Meng), his main emphasis was on distributed systems and parallelism for enterprise applications.

Alexander Kipp holds a Diploma in Computer Science. Since 2006, he has been employed at the High Performance Computing Center (HLRS) at the University of Stuttgart, working on cloud-/VO-/web service-related research projects on both national and international levels. He is the deputy head of the Intelligent Service Infrastructures department at HLRS, which focuses on realizing next generation cross domain service infrastructures. In this context, he is responsible for the Green-IT activities of HLRS. His research interest lies in the areas of adaptive and energy aware service infrastructures, in particular, in the area of service virtualization and corresponding technologies.

Stefan Wesner holds a Ph.D. in Mechanical Engineering from the University of Stuttgart and a diploma in Electronic Engineering from the University of Saarland. He is the Managing Director of the High Performance Computing Center, Stuttgart and heads the Applications and Visualization Department. His current research interests are automated management, SLA-based service provision for distributed environments and new paradigms for parallel computing. He was one of the first involved in the research on business-oriented grids and the application of service-level agreements for grids. He was the technical coordinator of the Akogrimo and BREIN projects and is currently involved in several research projects in the fields of cloud and high performance computing.

Marcelo Corrales holds an LL.M degree, and is a lawyer admitted to the Paraguayan Bar Association since 2004. In 2007, he got a Master's Degree in Law and Information Technology and, in 2009, a Master's Degree in European Intellectual Property Law from Stockholm University. Since October 2007, he has been a research associate at the Institute for Legal Informatics (IRI). His research interests include intellectual property rights and privacy issues in the realm of information law.

Nikolaus Forgó studied Law, Philosophy and Linguistics in Vienna and Paris. In 1997, he got his Dr. iur. (Dissertation in legal theory). Between 1990 and 2000, he worked as an Assistant Professor at the University of Vienna (Austria). Since 2000, he has been a full-time Professor for Legal Informatics and IT-Law at the University of Hanover; since 2007, the co-head of the Institute for Legal Informatics. Publications, teaching and consulting experience in all fields of IT-law, legal informatics, civil law, legal history and legal theory on national and international levels.

Tabassum Sharif completed his B. Eng. in Electronic and Electrical Engineering at the School of Electrical and Electronic Engineering with the Corp of Royal Electrical and Mechanical Engineers. He has spent almost 8 years within the military specializing in telecommunications and various other communication projects. He has previously worked with organizations within the financial services industry, insurance industry and prepayment industry before coming to the ISP industry where he is currently employed as the Operations Manager. He brings a wealth of experience in translating theoretical ideologies and best practices into real world environments.

Craig Sheridan holds a B.Sc.(Hons) in Network Computing and has worked for Flexiant since inception in various roles including project management and systems administration and has been a Support Manager and Systems Administrator for XCalibre Communications for the past 5 years and has various IT related certifications. He has a wide range of expertise in cloud technology. Prior to this, he worked for 8 years for Motorola as a Radio Frequencies Analyser.