Scalable Computing: Practice and Experience
Volume 13, Number 4, pp. 303–314. http://www.scpe.org
ISSN 1895-1767

A PANORAMA OF CLOUD SERVICES

DANA PETCU*

Abstract. Cloud computing paradigm has attracted a lot of attention in the last five years as coming during an economic crisis with an appealing offer in reducing the infrastructure and maintenance costs. After first wave of enthusiasm in adopting the concept, a clearer image has been formed about the benefits and limitations of Cloud computing and a lot of different supporting technologies were developed. As consequence a new threat is raised by the high number of the proprietary technologies that makes difficult the decision of the proper technological selection according to the real business needs. In this context the aim of this paper is to offer a snapshot on the current concepts and the available technologies, especially of the ones that can allow the development of a solid market of Cloud services and applications. A particular attention is given to the trend of federating and brokering Cloud services in the process of forming new markets. Moreover, we propose a classification of groups of services from multiple Clouds based on models similar to the ones used in computer graphics to express colors. Furthermore, a technological solution aligned to the market requirements is presented as case study, pointing also to the role of open-source codes for promoting the Cloud service usage on large scale.

Key words: Multi-Cloud, Cloud Federations, Cloud service markets

AMS subject classifications. 15A15, 15A09, 15A23

1. Introduction.

The term of Cloud Computing has been coined one half decade ago to name a new approach of providing services via Internet. The Cloud term is not accidentally or trendy: it is related to the already classical form of representing of the Internet connections between their multiple end users. By this image it catches in a simple word a long-time expressed desire to see the connectivity, storage, processes or applications as utilities connected via the Internet (which becomes finally The Computer by incorporating in it these utilities).

The interest in the utility concept 'pay-as-you-go' for e-infrastructure services promoted by Cloud computing supporters has been amplified by the context of the economic crisis, creating the illusion that the future is Cloudy. The hype of Cloud computing sustained by the big companies has rapidly created an ad-hoc market of services that are quite diverse due to several reasons, like different understanding of the concepts, complexity of the underlying software stacks, or need to promote earlier legacy software that are able to support the new concepts.

The technologies and services that are supporting the previous described concepts have been developed into a very large and fast evolving pool of Cloud computing offers, creating an ad-hoc e-market in which the main actors, developers, providers and end-users have clear roles. The providers are offering new services that are allows them to create a benefit from sharing their un-spend e-infrastructure resources. The users are primarily interested in the high availability, reliability and ubiquity of the services. The developers are interested to enlarge the base of end-users of their products. Unfortunately this market is driven by the providers and developers needs and end-users are struggling with the diversity of the concept approaches and the lack of uniformity in what concerns the interfaces or protocols (leading to a vendor lock-in).

In this context we consider useful to start the next section with a light presentation of the terminology currently used in Cloud computing and to point towards one particular problem emerging from the diversity of the current Cloud service offers (vendor lock-in). Section 3 is devoted to the emerging Federations and Markets of Clouds. We propose a classification of different views on these groups of services using models similar with the ones used in computer graphics. Moreover, we identify the main problems to be solved in order to build Federations and Market of Clouds. A particular example of middleware supporting the Federations and Markets of Clouds is exposed in Section 4. The last section is dedicated to the conclusions and future expectations.

2. Overview of the Cloud services' offers.

The aim of this section is to provide an overview of the categories of services that are currently offered and to point towards the limitations of the market offer. A special attention is provided to the open-source as solution for the vendor-lock in problem and the need of the free movement in the market of Cloud services.

2.1. Basic terminology.

Despite the interest in the new concept, the definition of what Cloud Computing is still not generally accepted, and its borders and relationship with other distributed computing paradigms are still discussed. We consider here only two definitions of well known authorities: Expert Group of European

*Institute e-Austria Timişoara and West University of Timişoara, Romania, (petcu@info.uvt.ro).

Commission on Cloud Computing [28] and NIST [8]. The definition of the Expert Group is focusing at Clouds as execution environments, concluding that an environment can be called Cloudified, if it enables a large dynamic number of users to access and share the same resource types, respectively service, whereby maintaining resource utilisation and costs by dynamically reacting to changes in environmental conditions, such as load, number of users, size of data. In the NIST definition, the Cloud is seen as a model: Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction.

Pros and contras for these definitions can be formulated by different stakeholders, like services providers, developers or end-users as they views can be different, and such debate is not subject of this paper. Important are the main characteristics of the Clouds, for which an agreement is close to be reached. According to the above mentioned NIST report, the essential characteristics are: (a) on-demand self-service; (b) broad network access; (c) resource pooling with multi-tenancy; (d) rapid elasticity; (e) measured service. These characteristics are perceived mainly from a user perspective. They are further split and re-grouped into two categories by the Expert Group, from a provider and developer point of view:

1. intrinsic characteristics, specific to the Clouds: like elasticity; multi-tenancy, high availability, and automated management;
2. extrinsic characteristics, that are extended or inherited from parental domains of utility computing, service architectures, or general IT: like virtualization, pay-per-use, market mechanism (from utility computing), resource management, metering (from service architectures), tool support, programming, or data management (from general IT).

One of the advances that Cloud computing is bringing and is not pointed in the above mentioned documents, but which we consider highly relevant for this paper is the implementation of the idea of programmable e-infrastructures. Using the programming tools available to the developers of applications, e-infrastructure services can be allocated, de-allocated or configured. This is a big step forwards to the self-adaptability of the execution environments as well as agility at the level of applications.

2.2. Classification of the services and tools. Despite the controversial disputes on the Cloud definition, there is an almost well-established consensus on the delivery and deployment models in Cloud computing. We remind them in what follows for the sake of continuous flow of presentation, after which we present other controversial or new classifications.

One of the basic concepts in Cloud computing is the delivery as-a-Service. The delivery is done using the Internet protocols and standards. Three main categories of service models (or delivery models) are recognized (e.g. in NIST report by [8]):

1. Infrastructure as a Service (IaaS) a consumer can get service from a full computer infrastructure through the Internet (Internet-based services such as storage and databases are typically considered a part of the IaaS).
2. Platforms as a Service (PaaS) offers full or partial application development environments.
3. Software as a Service (SaaS) provides a complete turnkey application via the Internet.

IaaS delivers a computing hardware infrastructure over the Internet and is enabled to split, assign and dynamically resize these resources to build custom infrastructures, just as demanded by customers. What makes the Cloud a novelty is the self-management capabilities it offers, the possibility of an almost immediate resizing of the assigned resources, and the application of the pay-per-use revenue model.

PaaS offers an additional abstraction level: rather than supplying a virtualised hardware infrastructure, they provide the software platform where customer services run on. Sizing of the hardware resources demanded by the execution of the user services is made by the PaaS provider in a manner transparent to the user. IaaS and PaaS systems have in common their aim to be a platform for their users.

SaaS, in contrast, groups together Cloud systems in order to create a final aggregated service itself. These services are software products that can be in the interest of a wide variety of users.

Many other resources can also be offered as Cloud services, such as Storage as a Service, Messaging as a Service, Network as a Service, Data as a Service, Communication as a Service, Database as a Service, Information as a Service, Process as a Service, Application as a Service, Integration as a Service, Security as a Service, Management as a Service, Testing as a Service etc. They are usually just particular types of one of the three groups of delivery models mentioned above (IaaS, PaaS, and SaaS).

A particular attention is given recently to the emerging technologies for a new model of Business-Processes-as-a-Service (BPaaS). Early implementations of this concept are no earlier than two years ago [23]. It was proposed by [30] to set BPaaS to the same level as the other three models, instead in SaaS category, due to the impact that can have on the business community.

The basic deployment models are the Private and the Public Clouds. In a Private Cloud the services are provisioned for the use of a single organization with multiple known members and these services are relying on- or off- premise e-infrastructures. A Public Cloud is designed to serve a general public, the owner of the resources (hardware and software) being the Cloud provider. Between the two models is the Community Cloud that serves two or more organizations that have agreed about the membership, security, mission and other common concerns, and can comprise one of more Private Cloud installations. The combination of the two or more Clouds bound only by technologies that are enabling data and application portability are classified by NIST report as Hybrid Clouds.

Several taxonomies and ontologies were already published to differentiate the Cloud concepts and terms and their inter-relationships and with the particular terms that are used by different providers. An example of such ontology can be found in the book chapter by [20].

Without willing to complicate the existing classifications, for the purpose of this paper and its understanding, we consider that the Cloud technologies and tools that are available to support the above described models should be split into two main categories: hosted services and deployable services. A hosted service is an integration of hardware and software exposed as a service compliant with the Cloud characteristics on a wide area network by a certain organization. A deployable service is a software that includes an interface of a service compliant with the Cloud characteristics and that is installable on certain e-infrastructures and is. Deployable services can be used to build hosted services residing on- or of-premises e-infrastructures.

Already classical examples can be used as examples to distinguish between the two categories. At the IaaS level, Amazon EC2 is a hosted service, while Eucalyptus is a deployable one (managing virtual machines); Amazon S3 is a hosted service, while Riak is a deployable one (for key value store); Amazon SQS is a hosted service, while RabbitMQ is a deployable one (for message queues). At the PaaS level, Google App Engine is a hosted service, while VMWare CloudFoundry is a deployable service. At SaaS level, Google Mail is a hosted service, while VMWare Zimbra is a deployable service.

The hosted services are the most common services and therefore the common understanding of Cloud services is referring to this group. The way in which their interfaces are conceived is very convenient for the application developers as they are hiding complex processes and heterogeneous resources. On another hand, the variety of the design of the interfaces of the hosted services creates a dependence between the application that is developed and the Cloud for which is developed (the vendor-lock-in problem) and hinders the interoperability between multiple Cloud services (the interoperability problem).

The deployable services have the potential to overcome the vendor-lock in and interoperability problems if they are adopted by several Cloud providers on a wide scale. Moreover, most of them are offered as open source, so that the developer community can help their improvement and adaptation to their or community needs.

2.3. The vendor lock-in problem. The heterogeneity of multiple Clouds is reflected in the variety of services offered by various Cloud providers, their interfaces, as well as in the variety of the hardware and software stacks that are used. The developers of Cloud-aware applications are facing a big problem in selecting the proper Cloud services that are matching their application needs.

On another hand, the usage of services from multiple Clouds has been introduced first with the idea of Hybrid Clouds, when Private Clouds are combined with the Public Clouds. The outages of Public Clouds have brought into discussions the migration of the applications and data from one Cloud to another. Moreover, small Cloud providers who have emerged recently are facing the problem of limited resources and they are interested to make agreements to other providers to support scalability beyond their resources in peak cases. These scenarios are more often discussed in the latest years in conjunction with a technical problem that has arrived with the increase the number of Cloud providers: vendor lock-in.

The reasons of vendor lock-in are various: proprietary APIs of the services, lack of accepted standards, particular services that are subject of high investments and so on. The problem is not due to the vendor will, but instead is a reflection of the large set of hardware and software stacks needed to build a Cloud service. Heterogeneity is encountered to both low and high levels, from virtualization technologies, to programming environments. It is expected that the middleware provided by the Cloud provider or even meta-providers like

Cloud brokers are hiding this heterogeneity. If this is happening at the Cloud provider level to a certain level (at least from the point of the view of the users), the meta-level is still lacking break-through offers beyond the research prototypes. Therefore we considered useful to identify which are the challenges in building middleware to deal with heterogeneity between Clouds.

3. Dealing with the multiple Cloud services. This section intends to make a survey of the solutions that are involving multiple Cloud services. The next sub-section is discussing the different concepts that were considered in the context of the meta-level of multiple Clouds. The second subsection is introducing a new classification method. The third subsection is devoted to the challenges associated with two specific meta-levels, Federations and Markets.

3.1. How to Name Each Case of Grouping Services from Multiple Cloud?. In this sub-section we present an overview of various approaches to nominate the multiple Cloud usage scenarios.First we should remind that the meta-computing idea was coined almost two decades ago to point the idea of grouping several computing e-infrastructures, and the idea of grouping Clouds has several commonalities with the meta-computing.

NIST recent report [8] has divide the usage scenarios in two categories, according to the number of Clouds involved at a moment of time: sequential, when Clouds services from different providers are used one after another, or simultaneously, when Cloud services from different providers are used in the same time. Sequential is encountered in the case of migration from one Cloud to another, in the case of the selection of the service at deployment (contrary to the selection at the design time, scenario in which only one Cloud is involved), or when interfaces for software and data transfer are build between Cloud providers in agreement between them. Migration can be required from various reasons, like changing to adapt to resource availability, to the resource cost or to adapt to the changes in application requirements (like emerging deadlines). Simultaneous usage is often encountered in the Hybrid Clouds, when parts of the applications are residing on-premise resources (Private Cloud) and parts on Public Cloud resources.

InterCloud term was introduced by [1] in analogy with the Internet and based on a similar vision: to connect individual Cloud infrastructures and giving control to the users. The initial term has supposed a certain agreement between Clouds in what concerns the interfaces. Another term that is used is Cloud-of-Clouds as analogy with the Grid which is a Cluster-of-Clusters. Other terms like Cross-Cloud or Sky Computing [11] have introduce the brokerage of Cloud services.

A two-level classification is provided by [4] where the multiple Clouds scenarios are split in two, in another dimension, according to the software stacks: Horizontal Federations when Cloud providers are federate for scale and capacity enlargement reasons; and Vertical Supply Chain when Cloud providers are leveraging services from other providers.

In the paper by [6] the multiple Cloud usage scenarios are split in three cases: (a) Bursting Private Clouds (expansion of Private towards Public ones); (b) Federated Clouds (partnership); (c) Multi-Clouds (providers working with external services).

According the article of [19], the coupling between the acquired resources is considered as criteria to split the multiple Cloud usage scenarios: (a) loosely coupled federation in which the inter-operation is low (monitoring is limited, no control on external resources, no migration of virtual machines); (b) partially coupled federation, when an agreement between the providers has been established concerning different issues like interchanging monitoring information, virtual networks across Cloud boundaries, or control over remote resources; (c) *tightly coupled federation*, when the agreement allows full control on remote resources and their monitoring, creation of cross-site networks or virtual storage across Cloud boundaries.

In the same paper [19] the authors are discussing four potential architectures of the frameworks supporting the multiple Cloud usage scenarios: (1) Hybrid Cloud (Cloud bursting), coupling on-premise infrastructure with remote resources from Public Clouds (in the loosely coupled category); (2) Cloud broker with a broker that serves users and has access to several Public Clouds (loosely coupled); (3) Aggregated Clouds when several providers aggregate their resources (partially coupled); (4) Multi-tier Clouds when a hierarchical agreements are established so that a Cloud provider has full control over the resources of different Cloud sites (tightly coupled).

Browsing the literature, we see that the above terms are often used with different meanings, therefore we considered necessary to propose a classification scheme that tries to cover as much as possible the above described cases.

Without any intention to complicate the image, but for reasons exposed in the next section, in this paper we introduce another keyword, namely Market of Clouds. This market is expected to provide a single interface for

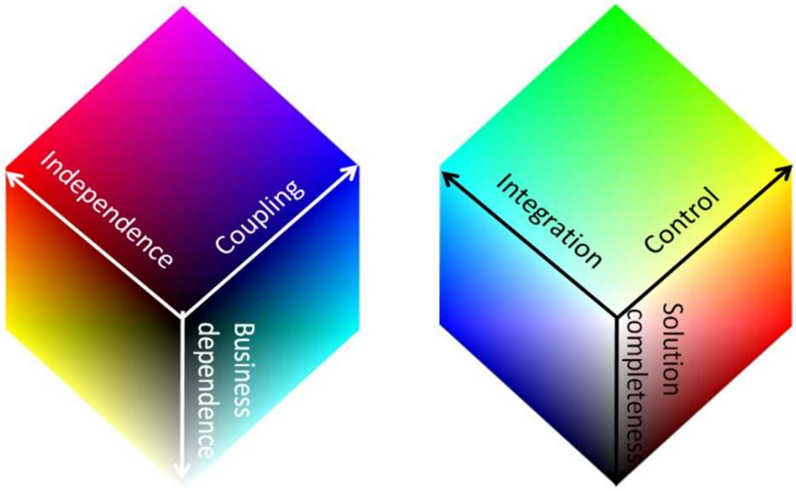

Fig. 3.1: Users' requirements in RGB model vs. Providers' requirements in CMY model

a consumer to address resources from multiple Clouds. The key element of a market is the broker mechanism, which operates outside of the Clouds, monitors the connected Clouds, detects their failures and react to them in order to comply with the clients requests by having the permission to move virtual machines, appliances, applications or data from one Cloud to another. While Federations of Clouds can be similar with Grids, Markets can be seen as following the Web services concepts.

3.2. Colors of user and providers. In what follows we propose a classification of the cases of grouping services from multiple Cloud. We use an analogy with the basic color models from graphics: Red-Green-Blue (RGB), used for example by displays, and Cyan-Magenta-Yellow (CYM), used for example by printers.

We consider that in RBG model (Fig. 3.1) we have on the axes:

x axis (Red): the degree of the independence from the Cloud provider. Zero is associated with the Federation since the user is addressing one Cloud and this Cloud is deals with the multiple Cloud services. One is associated with the Market which allows the user to select the Cloud.

y axis (Green): the degree in which new business are build. Zero is associated with the Horizontal Federation or one Cloud. One is associated with the case of Vertical Supply Chain.

z axis (Blue): the degree in which the coupling is done between Cloud services. Zero is associated with lack of coupling. One means a tight coupling.

The origin of the RGB-like system is black corresponding to the Horizontal Federation with no coupling (collection of isolated Clouds). Hybrid Clouds are red: (1,0,0). Brokers are yellow: (1,1,0). Aggregated services are examples of points in one axes plane. Multi-tier Clouds are cyan: (0,1,1). The 'maximum' of all values, (1,1,1), is represented by Market of tightly coupled services allowing vertical supply chains. This is an expression of the desire of the Cloud users; therefore we consider that the RGB representation reflects their wishes with black being the worst case.

We consider that in the CMY model (complementary to the RGB model) we have the opposite:

x axis (Cyan): opposite to the x axis from RGB model, expresses the integration degree with other Clouds, zero being the Market, and one, the Federation;

y axis (Magenta): opposite to the y axis from RGB model, expresses the completeness of the solutions offered by a certain offer, at zero being the Horizontal Federation or the single Cloud, and at one the Vertical Supply Chain;

z axis (Yellow): opposite to the z axis from RGB model, expresses the control over own resources in a Federation or Market, at zero being the full control, and at one being full controlled.

The origin of the CMY-like model is white, corresponding to the above mentioned 'maximum' wish of the users. The black is the 'maximum' of all values; in this case represents the Horizontal Federation in which loosely coupled services are offered. This maximum can reflect the wish of the providers to have full control

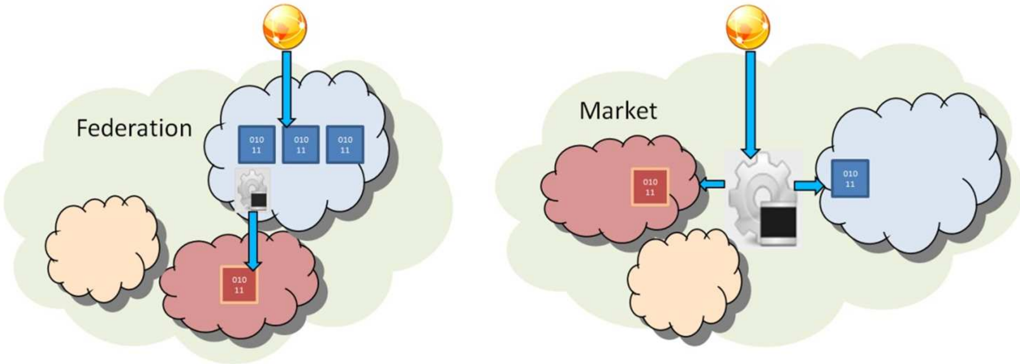

Fig. 3.2: Federation vs. Market of Clouds (the application is represented by the ball, the services rented in the Cloud are represented by boxes)

on their resources, to satisfy all the requirements of the users, but being in a Federation with other providers without high obligations (or even alone, one Cloud).

The fact that the two models are opposite to each other reflects also the current status of the offer of the Cloud service markets. The user expectations are not always in agreement with the offers. The consensus and a market equilibrium can be probably found where the colors of the parties are identical: in the middle of the cubes, at a middle gray.

3.3. Challenges in Federation of Clouds and Markets of Clouds. The implementation of middleware supporting multiple Clouds is not trivial. One reason is the fact that the existing models for interoperability and orchestration of the services are designed for static environments, while Clouds are typically dynamic and even in a Federation agreements among the Cloud providers are dynamically established.

In this section we will analyze just one dimension - the red one from the previous RGB model in an effort to identify the research and development issues that are making the complete middleware for Federations and Markets still unavailable.

We remind that the two cases are different in the way they are treating the user and the cooperation between the Clouds. Figure 3.2 is trying to express this difference in a graphical way.

The main problems identified until now in the middleware developments for Federations of Clouds are the followings:

1. Supported by the interoperability inside the Federation, there is a need for a component placed at the Cloud provider site (similar to a broker, named here manager) allowing the match-making with available external services and authentication procedures for these external services.
2. Live virtual machine migration should be coupled with load balancing to increase power efficiency. Problems that should be surmounted are for example related to the migration beyond the network boundaries without losing the already established network connections and storage of virtual machines on shared file systems on large scale.
3. An interoperability framework based of common understanding of Cloud providers on the main terms that are used is a key element of an efficient Federation. We have identified recently the interoperability and portability issues [24].
4. Cloud computing providers are limiting the connectivity of the virtual machine and their network traffic. Network overlay technologies can be used as a solution to overcome these limitations.

Table 3.1 is presenting some examples of current prototypes that are implementing some innovative solutions to these problems.

Several other issues have been treated until now only at theoretical level:

- Meta-schedulers are needed to support a coordinated distribution of different Clouds workloads. The process is slowed also due to the fact that several Clouds do not support scalable load balancing.
- The scheduling in such environments is a complex decision mainly due to their dynamics: resources behaviour

Table 3.1: Issues in Federation of Clouds and available solutions

Subject	Examples of prototypes
Federation manager supporting the external services selection	CCFM [4] is a Cross-Cloud Federation Manager with discovery, match-making and authentication features
	ORCA [17] enables computational and network resources from multiple clouds and network substrates to be aggregated into a single virtual resource
Live migration of virtual machines	Shrinker [27] is a modification of KVM hypervisor based on the detection of inter-virtual-machines data similarities
Interoperability frameworks	PSIF [15] models and tries to resolve semantic interoperability conflicts raised during the deployment or the migration of an application between PaaSs
Network virtualization techniques for distributed resources in different administrative domains	TinyViNe [32] for Nimbus installations with MPI jobs benchmarks. Other solutions are presented in the same paper

is unpredictable, local schedulers should interact with each other, resource sharing is based on service level agreements that can be changed dynamically. A review of approaches at theoretical level is provided by [31].

– The level of integration of different security technologies should permit a new provider to join the Federation without changing his security policies or authorisation processes. Moreover, the user already authorized to a certain Cloud should be able part of the resources of the resources (like in Grids).
– A monitoring meta-system, hopefully independent from any provider solution, of the resources of different providers from the Federation should be designed and developed.
– Automated operations should use intelligent management systems. An approach using rule-based techniques was proposed by [12]. Currently Cloud providers must manually resolve sub-optimal configurations, and maintain an on-going balance between capacity utilization, cost, and service quality [3]. Self-adaptability to the changes of each provider service availability is strictly necessary in the near future.
– Integration-as-a-Service is a way to abstract the technical details and the interaction with the cloud services and to provide a way to treat these interactions as part of the abstract description of a Cloud-based solution. Referring to this idea, the Cloud Blueprinting, introduced in by [21], includes a detailed deployment plan of an applications and a high-order packaged integration solution that provides a description of the integration needs for the interaction between Cloud services provided by different providers.

In the case of Markets of Clouds, the main issues for research and development are related to the followings:

1. Brokers are acting as intermediaries between providers and clients, being able to allocate resources among multiple Cloud offers. A broker assists the clients in selecting the appropriate service that best suits (using several criteria) their requirements and needs. Potentially, the request is split by the broker such that different providers receive sub-requests for provisioning or instantiation of resources. The broker should provide a single entry point for a specific market and, in the best case, a delegation mechanism in what concerns the user credentials should be part of the broker and the output of the brokering process should be the allocation of the proper resources. An overview of the requirements for a broker is provided by [18]. Note that beyond the research prototypes enumerated in Table 3.2 there are already commercial offers (like Rightscale's Multi-Cloud Engine that is able to broker capabilities related to virtual machine placement in several Public Clouds) or in-production research prototypes like OpenCirrus.
2. Using the same APIs the dynamic allocation of the resources and binding components of the applications to the acquired resources should be possible.
3. Search engines with matching algorithms and based on semantic technologies are needed; several user requirements should be supported (functional and non-functional ones).
4. The diversity of services complicated the service selection. A methodology to compare Cloud service

Table 3.2: Issues in Markets of Clouds and available prototypes

Subject	Examples of prototypes
Brokers	Cloudbus [2] uses several brokers which are interacting with a coordinator
	Zeel/i [7] allows single-sign (using the Cloud credentials of the Zeel/i) and the selection of Cloud resources according to specific requirements
	Extension of OpenFlow [9] which is solving the selection problem expressed as mixed integer program
	SORMA [22] use bidders and sellers to represent the beneficiaries of the brokering system
	SERA [5] is using a multi-agent system with agents representing the beneficiaries of the brokering system augment with special duties of scheduling or controlling the resources or monitoring, registering or recovery.
Uniform APIs	SAGA [16] dynamically allocate resources via a job interface and bind sub-jobs to these resources
Search engines	Cloudle [10] is a Cloud service search engine based on a specific Cloud ontology.
Benchmarks	CloudCmp [14] is a set of benchmarking tools for comparing services from elasticity, persistence of storage, intra-cloud and WAN communications.

based on multiple criteria is expected. Comparison criteria can vary from cost, policies, performance and so on. For the performance measurements independent observer services need to be built.

Other problems that should be solved in Markets of Clouds are related to:

– Automated approaches for deploying virtual appliances are expected to emerge.
– Deployment description languages should target application run-time aspects. Key requirements are stated by [13].
– Service aggregator that combines services from different Clouds, including dashboards or smashups should be build.
– Expert systems for recommending systems are expected to emerge. Artificial intelligence techniques, from reasoners to evolutionary computing or even multi-agent systems can found interesting applications in this field.
– Multi-Cloud governance - high level management. The paper [29] are proposing autonomic approach based on a governance model where a high-level manager dynamically adapts the behaviors of the low-level managers by fine-tuning their policies.
– Portability in this context is the ability to migrate applications between different Clouds (subject of the next section). Standardized and open interfaces and protocols to manage Cloud services are required.

4. Case study of a support for Markets of Clouds: mOSAIC, an open-source Platform-as-a-Service. We have recently contributed to the development of the open-source platform-as-a-service named mOSAIC. It is designed to allow the portability of applications on top of different infrastructure-as-a-services. The applications are expected to be built from components and to communicate via a message passing system. An event-driven approach should be adopted when dealing with the Cloud resources that are interfaces through the vendor-independent API.

mOSAIC (Open source API and Platform for Multiple Clouds) is developed in the frame of a multi-national collaborative project funded by the European Commission in the period 2010-2013, and it involves more than forty persons, software engineers and programmers, as well as application designer and developers.

The open-source middleware, currently in a stable version, is deployable and available at https://bitbucket.org/mosaic. Details about the proposed API can be found in the article proposed by [25], in on-line documentations (http://developers.mosaic-cloud.eu) or demos (YouTube, key-phrase mOSAIC Cloud computing), and the project site (http://www.mosaic-cloud.eu).

mOSAIC system has a complex architecture (Figure 3) that includes:

(a) core platform services: from scheduler, load balancer, deployer, provisioner, scaler, monitor, component discoverer, specific virtual appliances, and so on, that are independent from the Cloud services;
(b) market services: broker based on multi-agent technologies, service discoverer, semantic engine based on Cloud ontology to match the functionality of the system and the services with the user requirements,

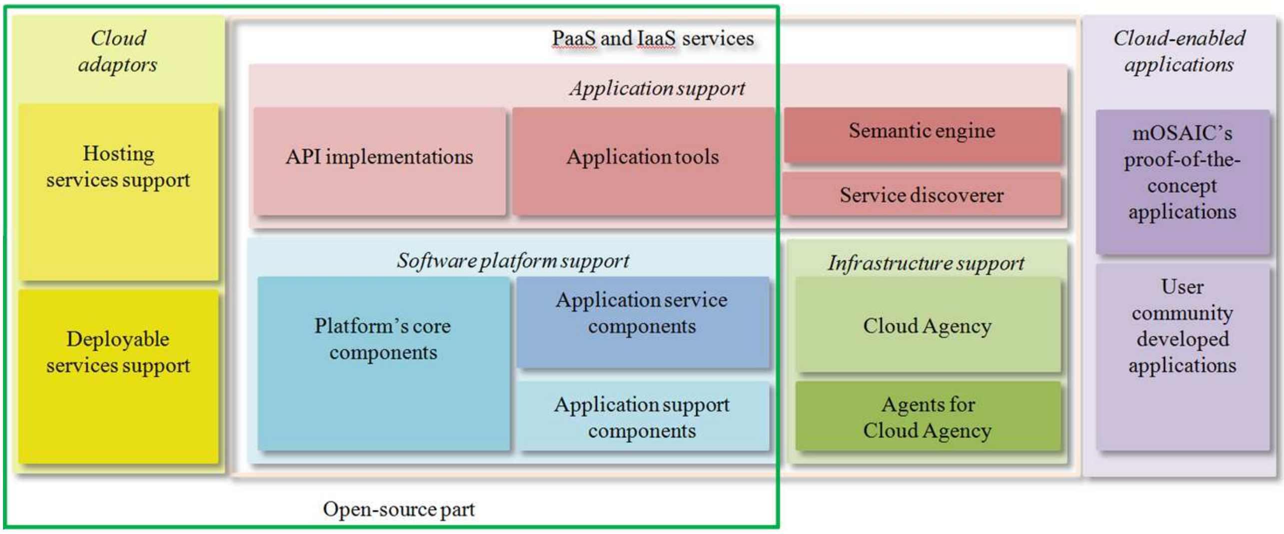

Fig. 4.1: General overview of the component architecture of mOSAIC solution

as well as service-level-agreements negotiation mechanisms;

(c) Cloud connectors and agents: to Public Cloud services as well as support to deploy open-source Cloud technologies (from more than ten providers);

(d) application development support: from a Desktop Cloud allowing the debugging of the developed application on desktops, to web interfaces of the platform to control the life-cycle of the components.

Similar efforts in providing open-source PaaS are undertaken currently by companies like VMWare (Cloud Foundry) or RedHat (OpenShift). While these are oriented mainly to web applications, mOSAIC intends to support also other types of applications, like scientific ones, or even business processes.

In this section we present the compliance of mOSAIC features with the ones requested for Federations and Market of Clouds, which we identified in the previous section. Table 4.1 synthesizes the results of this analysis. We consider that the criteria exposed in Table 4.1 can be further used to compare several solutions for Federations or Markets of Clouds.

5. Following the trends. We presented in this paper a particular view on the status of Cloud services and the current efforts to build Federations and Markets of Clouds expected to be the next step in the development of the Cloud services. We have also pointed towards to the potential support for building Markets of Clouds coming from a new open-source and deployable platform as a service, namely mOSAIC. Moreover, we consider that the usage in Cloud computing of open-source software can be a signal for a level of maturity of the Cloud technologies. When the diversity of the open-source stack will be reached the Cloud will expand beyond its current limitations, like the vendor lock-in problem.

The lessons learned in the development of mOSAIC embrace several topics. We mention here only few general ones. The diversity of the Cloud services has reached a certain degree to which the finding a common denominator is almost impossible and therefore the proposal of new standards in the field should be complemented by frameworks that are leaving the door opens for innovation in a competitive market. The availability of deployable solutions enables the fast development of new technologies and the most mature ones can be considered embeddable and trustable bricks in building a solid platform for building applications consuming Cloud services. The degree of automatisation that is expected from any group of Cloud services is hardly faced by the current mechanisms (like schedulers, auto-scalers, resource provisioners and so on) and the development of new solutions tailored for the case of Federations and Markets of Clouds are needed.

Several European collaborative projects, partially funded by the European Commission, and involving tens of research and development teams of Cloud technologies as well as users of Cloud services, are currently working to realize the vision of Federation of Markets of Clouds. We remind here only few on them, beyond the one already mentioned, mOSAIC (details about these projects and other similar ones can be found at least in the book [26]). Contrail (www.contrail-project.eu) is providing a solution for the Federation of Clouds. TClouds

Table 4.1: Compliance of mOSAIC with the requirements of Federations or Markets

Subject	mOSAIC solution
Federation manager/Broker	The Cloud agency augmented with the broker, vendor agents and the SLA mechanism are ensuring the selection of one or more services that are satisfying the requirements of the application (based on an application descriptor, the final result being the provisioning of the resources)
Live migration of VMs	Not supported at the level of the platform. But live migration of application components (not encountered in other middlewares), yes.
Interoperability framework	The semantic engine assist the developer of the applications to find the right functionality of the API and the Cloud services, based on a Cloud ontology.
Network virtualization techniques	A naming service was designed based on DNS service extensions
Uniform APIs	The APIs are vendor-independent
Search engines	Under development, architecture and services already established
Benchmarks	The benchmark framework allows to setup a custom benchmark which measures the performances of the target application under well known workloads
Meta-schedulers	Based on genetic algorithms for multi-criteria optimizations
Integration of security technologies	Credential service and an Intrusion-detection-as-a-Service
Monitoring meta-system	Not supported
Automated operations	Scaler and scheduler based on agent technologies and rules. Self-adaptability is in research phase
Integration-as-a-Service	The platform uses application descriptors, call for proposals (of resources) and application deployment descriptors that are matching the Cloud blueprinting idea
Deployment description languages	The above mentioned descriptors are described in a kind of XML based language
Automated deployment of VA	Virtual appliances are prepared on the fly (virtual machines with the platform controllers and deployable Cloud technologies) and deployed
Service aggregator	Aggregator should be part of the deployed application. Aggregation at the platform level is resumed to the component discovery mechanisms
Recommending system	Not supported
Portability	Possible if the component-based applications are compliant with the rules related to communications, architectural style (event-driven) and programming languages (currently Java and Python)
Multi-Cloud governance	Under development, architecture and services already established

(www.tclouds-project.eu) is offering security, privacy and resilience mechanisms for Federations and Markets of Clouds. 4CaaSt (4CaaSt.morfeo-project.eu) is proposing a BluePrint for registering the Cloud services in an e-Market. Optimis (www.optimis-project.eu) is providing brokerage mechanisms. Innovative technologies that are enabling the design of Federation and Markets are developed in the frame of: Vision Cloud (www.visioncloud.eu) which is looking in details to the issues of data management in Clouds; Cloud4SOA (www.cloud4soa.eu) which is dealing with semantic based interoperability at platform level; Remics (www.remics.eu) which is dealing with migration of legacy applications to Clouds; Cloud-TM (www.cloudtm.eu) proposing a new programming paradigm for Clouds. The integration of the research results in the production lines of the commercial partners of these projects are expected to happen in a range of two-there years.

Acknowledgments. This work was partially supported by the grant of the European Commission FP7-ICT- 2009-5-256910 (mOSAIC) for the Sections 4-5 and Romanian National Authority for Scientific Research, CNCS UEFISCDI, PN-II-ID-PCE-2011-3-0260 (AMICAS) for Sections 1-3.

REFERENCES

[1] D. Bernstein, E. Ludvigson, K. Sankar, S. Diamond, M. Morrow, *Blueprint for the Intercloud - protocols and formats for cloud computing interoperability*, Procs. ICIW '09 (2009), pp. 328–336.
[2] R. Buyya, R. Ranjan, R. Calheiros, *Intercloud: utility-oriented federation of cloud computing environments for scaling of application services*, LNCS 6081 (2010), pp. 13–31.
[3] H. Cai, K. Zhang, M. Wang, J. Li, L. Sun, X. Mao, *Customer centric cloud service model and a case study on commerce as a service*, Procs. IEEE Cloud'09 (2009), pp. 57–64.
[4] A. Celesti, F. Tusa, M. Villari, A. Puliafito *How to enhance cloud architectures to enable cross-federation*, Procs. 3rd IEEE Cloud (2010), pp. 337–345.
[5] J. Ejarque, R. Sirvent, R. Badia, *A multi-agent approach for semantic resource allocation*, Procs. 2nd CloudCom (2010), pp. 335–342.
[6] E. Elmroth, J. Tordsson, F. Hernandez, A. Ali-Eldin, P. Svard, M. Sedaghat, W. Li, *Self-management challenges for multi-cloud architectures*, LNCS 6994 (2011), pp. 38–49.
[7] T. Harmer, P. Wright, C. Cunningham, R. Perrott, *Provider-independent use of the cloud*, Procs. Euro-Par'09 (2009), LNCS 5704, pp. 454–465.
[8] M. Hogan, F. Liu, A. Sokol, J. Tong, *Nist Cloud computing standards roadmap-version 1.0*, Special Publication 500-291 (2011).
[9] I. Houidi, M. Mechtri, W. Louati, D. Zeghlache, *Cloud service delivery across multiple cloud platforms*, Procs. SCC'11 (2011), pp. 741–742.
[10] J. Kang, K.M. Sim, *Cloudle: a multi-criteria cloud service search engine*, Procs. APSCC '10 (2010), pp. 339–346.
[11] K. Keahey, M. Tsugawa, A. Matsunaga, J. Fortes, *Sky computing*, Internet Computing 13 (5) (2009), pp. 43–51.
[12] G. Kecskemeti, M. Maurer, I. Brandic, A. Kertesz, Z. Nemeth, S. Dustdar, *Facilitating self-adaptable inter-cloud management*, Procs. PDP '12 (2012), pp. 575–582.
[13] A. Lenk, C. Danschel, M. Klems, D. Bermbach, T. Kurze, *Requirements for an iaas deployment language in federated clouds*, Procs. IEEE SOCA'11 (2011), pp. 1–4.
[14] A. Li, X. Yang, S. Kandula, M. Zhang, *CloudCmp: comparing public cloud providers*, Procs. 10th Conf. Internet measurement (2010), pp. 1–14.
[15] N. Loutas, E. Kamateri, K. Tarabanis, *A semantic interoperability framework for cloudplatform as a service*, Procs. 3rd IEEE CloudCom (2011), pp. 380–387.
[16] A. Luckow, L. Lacinski, S. Jha, *SAGA bigjob: an extensible and interoperable pilot-job abstraction for distributed applications and systems*, Procs. CCGRID'10 (2010), pp. 135–144.
[17] A. Mandal, Y. Xin, I. Baldine, P. Ruth, C. Heerman, J. Chase, V. Orlikowski, A. Yumerefendi, *Provisioning and evaluating multi-domain networked clouds for hadoop-based applications*, Procs. 3rd IEEE CloudCom (2011), pp. 690–697.
[18] H. Mearns, J. Leaney, A. Parakhine, J. Debenham, D. Verchere, *An autonomic open marketplace for inter-cloud service management*, Procs. UCC'11 (2011), pp. 186–193.
[19] R. Moreno-Vozmediano, R. Montero, I. Llorente, *IaaS cloud architecture: from virtualized data centers to federated cloud infrastructures*, Computer 99 (2012).
[20] F. Moscato, R. Aversa, B. Di Martino, D. Petcu, M. Rak, S. Venticinque, *An Ontology for the Cloud in mOSAIC*, In Wang L. et al, Cloud Computing: Methodology, Systems, and Applications, CRC Press (2011), pp. 467–486.
[21] D.K. Nguyen, F. Lelli, Y. Taher, M. Parkin, M.P. Papazoglou, W.J. van den Heuvel, *Blueprint template support for engineering cloud-based services*, LNCS 6994 (2011), pp. 26–37.
[22] J. Nimis, A. Anandasivam, N. Borissov, G. Smith, D. Neumann, N. Wirstrom, E. Rosenberg, M. Villa, Sorma - business cases for and open grid market: Concept and implementation, LNCS 5206 (2008), pp. 173–184.
[23] M. Pathirage, S. Perera, I. Kumara, S. Weerawarana, *A Multi-tenant Architecture for Business Process Executions*, Procs. ICWS'11 (2011), pp. 121–128.
[24] D. Petcu, *Portability and interoperability between clouds: challenges and case study*, LNCS 6994 (2011), pp. 62–74.
[25] D. Petcu, G. Macariu, S. Panica, C. Craciun Portable cloud applications - from theory to practice, Future Generation Computer Systems, doi: 10.1016/j.future. 2012.01.009 (2012).

[26] D. Petcu, J.L. Vazquez-Poletti (eds.) *European Research Activities in Cloud Computing*, Cambridge Scholars Publishing, UK (2012).
[27] P. Riteau, *Building dynamic computing infrastructures over distributed clouds*, Procs. IPDPS'11 (2011), pp. 2097–2100.
[28] L. Schubert, K. Jeffery (eds.), *Advances in Clouds Research in Future Cloud Computing*, Expert Group Report, Public version 1.0 (2012).
[29] M. Sedaghat, F. Hernandez, E. Elmroth, *Unifying cloud management: towards overall governance of business level objectives*, Procs. CCGrid'11 (2011), pp. 591–597.
[30] D.M. Smith, *Hype cycle for cloud computing*, Gartner Research G00214915 (2011).
[31] S. Sotiriadis, N. Bessis, N. Antonopoulos, *Towards inter-cloud schedulers: A survey of meta-scheduling approaches*, Procs. 3PGCIC'11 (2011), pp. 59–66.
[32] M. Tsugawa, A. Matsunaga, J. Fortes, *User-level virtual network support for sky computing*, Procs. 5th IEEE e-Science (2009), pp. 72–79.

Edited by: Enn Õunapuu and Vlado Stankovski
Received: Dec 28, 2012
Accepted: Jan. 10, 2013