Building a Mosaic of Clouds

Beniamino Di Martino[1], Dana Petcu[2], Roberto Cossu[3],
Pedro Goncalves[4], Tamás Máhr[5], and Miguel Loichate[6]

[1] Second University of Naples, Italy
[2] Institute e-Austria Timişoara, Romania
[3] European Space Agency, France
[4] Terradue SRL, Italy
[5] AITIA International Inc., Hungary
[6] Fatronik Tecnalia, Spain

Abstract. The current diversity of Cloud computing services, benefic for the fast development of a new IT market, hinders the easy development, portability and inter-operability of Cloud oriented applications. Developing an application oriented view of Cloud services instead the current provider ones can lead to a step forward in the adoption of Cloud computing on a larger scale than the actual one. In this context, we present a position paper exposing the concepts behind a recent proposal for an open-source application programming interface and platform for dealing with multiple Cloud computing offers.

Keywords: Cloud programming model, Federation of Clouds, Cloud-based applications.

1 Introduction

Cloud computing, currently used mainly in commercial settings and focusing on on-demand provision of services, has a clear potential to play a significant role in a variety of areas including innovations or e-business, virtual worlds or social networks. Cloud computing offers until date have been developed without addressing a common programming model, open standard interfaces, adequate service level agreements or portability of applications. Neglecting these issues current Cloud computing offers force people to be stranded into locked, proprietary systems. Developers making an effort in Cloud-ifying their applications cannot port them elsewhere. Moreover, users put in the hands of commercial providers applications and data without negotiable quality of service agreements. From these points of view, Cloud computing is still in its early stages with consistent experimentation to come.

In order to respond to the above described community needs, the mOSAIC project has been initiated in the frame of FP7-ICT programme. It intends to create and promote an open-source Cloud application programming interface and a platform targeted for developing multi-Cloud oriented applications. This early position paper about mOSAIC offer describes its concepts and expectations. It is organized as follows. Section 2 is dedicated to mOSAIC's manifest.

The user requirements motivating the concepts implementation are described through several scenarios in Section 3. Section 4 identifies the main challenges and components of the proposed software developments. Finally, Section 5 comments the implementation status and the benefits of mOSAIC offer usage.

2   Manifesto

The mOSAIC proposal is motivated by existing weaknesses of the current research and practice in Cloud computing: (a) lack of common programming model for Cloud-oriented applications; (b) lack of tools for easy deployment of scalable applications and (multi)-Cloud-based service compositions; (c) lack of standard interfaces for resource virtualisation; (d) lack of adequate service level agreements and their dynamic negotiation; (e) platform dependability and non-portability due to different APIs for different types of resources. To these weaknesses mOSAIC intends to respond by providing the followings:

1. Design a language- and platform-agnostic application programming interface for using multi-Cloud resources and Cloud usage patterns.
2. Build an open-source and portable platform for using Cloud services based on the proposed API and Cloud usage patterns.
3. Design a generic agent skeleton for representing various stakeholders, e.g. vendors and their resources, users of various types, and collection of modules that can be used to adapt agent skeleton to support needed functionalities.
4. Design user-centric service level agreements, a Cloud ontology and mechanisms for dynamic negotiation of resources based on multi-agent technologies and semantic data processing.
5. Build proof-of-concept applications with emphasis on data intensive applications.

The open-source platform will be a proof-of-the-concept prototype ready to be tested, exploited or extended by its users. It will include instances of the APIs for at least two programming languages and applications tools. Its semantic engine that, based on the Cloud ontology, will express the application's needs for Cloud resources in terms of SLAs and QoS requirements that are the inputs of a negotiation module. This module initiates a bid to the agents representing different Cloud resources providers. Cloud resources can vary from software services (including virtual appliances) or data services to hardware services.

In the current context of the Cloud market, mOSAIC is expected to offer the freedom of choice at programming level as well as at the resource level.

3   Usage Scenarios and Proof-of-the-Concept Applications

3.1   An Entreprise Usage Scenario

In order to describe an example of use of the mOSAIC API and the related framework and platform, we refer to a common Enterprise-to-Cloud use case

reported in [1] which involves an enterprise using Cloud services for its internal processes. Suppose that the context is the one of a project with limited duration and that will be a need which will stress the internal resource, reducing the SLA for all the other tasks of the enterprise. The enterprise has a fixed project fund for computational use and will acquire external computational resources from Cloud. Actual Clouds offer a large set of solutions able to solve this kind of problems: as an example the enterprise may acquire computational resources from EC2, customize the machines, deploy its data on the Amazon S3 storage system, setup their software on the target resources, define a set of procedures for accessing the Cloud resources from internal ones and viceversa, and only after that they will proceed to the target project. Even if the procedure remain the same for every kind of Cloud resources (e.g. EC2 or GoGRID), they should be completely re-executed from scratch, changing Cloud provider: different machine images, needing a new customization, new setup of software, probably different procedure to access the external resource from internal ones, and so on. Moreover, as side effect, once the project started using a given Cloud provider, even if a cheaper one arises, e.g. after more detailed evaluation, another Cloud proves to be cheaper than the chosen one, the change will probably be too expensive. Furthermore, if two different Cloud providers offer different features at different prices, and the best solution (in term of quality/cost ratio) is to use both for the different tasks, the cost of setting up the procedure on both will be very high. The overall cost may be not acceptable for a project with limited time and budget.

We expect that in this context the enterprise will use mOSAIC solutions to describe the kind of resources, to request them, and setup internal-external communication. Thanks to the mOSAIC the enterprise will have access to all the (supported) Cloud providers independently. No cost in changing the provider or difficulties in accessing different providers are foreseen. Moreover the solution developed may be easily reused for different applications and projects (the overall development cost and time are reduced). Furthermore is possible to use the mOSAIC framework in order to enrich the needed resource description with an expected quality/cost ratio or the expected cost, so that the framework, in an autonomous way setup a different set of resources or stop using them when the costs grows too high.

3.2 Application Scenarios

A special attention will be given to the validation of the API and platform through data-intensive applications and simulations. The motivation for data-intensive applications is the emergency of the fourth paradigm of scientific and technological discovery, the data-intensive science: the availability of diverse data is shifting scientific approaches from hypothesis-driven scientific method to science based on exploration. The following Gray' laws were recently postulated in the book [2]: scientific computing is becoming increasingly data intensive; the solution is in a "scale-out" architecture; bring computations to the data, rather than data to the computations. In this context Cloud computing offers consists in: allowing groups to host, process, and analyze large volumes of data;

consolidating computing and storage in very large data centers as an economical efficient solution; offering hosting facilitates for long-term data preservation.

On another hand the current scientific and business simulations can scale to a point where they are either computationally too expensive, or require too much memory to be able to run on simple computers, servers, or supercomputers. Simulations are therefore obvious candidates as Cloud applications, their extra needs being satisfied by customized resource allocations.

The mOSAIC's proof-of-the concept applications described in what follows have different requirements and intend to validate not only the usage of Cloud services and the proposed APIs but also the semantic engine for the selection of Cloud resources vendors.

Earth Observation Scenarios. Due to its intensive data processing and highly distributed organization, the multidisciplinary Earth Science (ES) applications community is uniquely positioned for the validation and exploitation of Cloud computing infrastructures. Petabytes of already acquired data are presently under-exploited (under 10%), because for getting the results in a reasonable time not enough computing resources are available. However, even if they were to be made available, an efficient infrastructure to handle and treat very large data sets is still missing. In particular mOSAIC will focused on Earth Observation (EO), a specific discipline of ES that well represents their needs and challenges.

Storage and data distribution. With the growth of network bandwidths and local storage capacities, the media for the distribution of EO data has a new approach. Users are able to discover, select and download data eventually combined with processing services. While the access to near-real time isi currently addressed, the EO systems are not able to respond to peaks of demand. Moreover, providing on-line access to huge amounts of data is challenging and different data policies and controlled access are issues that need to be addressed. Cloud computing for EO on-line data access can split the archiving/preservation from the on-line data access while providing a number of clearly identified benefits: (a) controlled access: access to data can be made private or public and specific rights can be granted to specific users; (b) several data access protocols: http/https and Bittorrent are common among Cloud computing providers; (c) high uptime rates, protected by SLAs; (d) controlled and simple cost model.

EO mission reprocessing. The EO mission re-processing targets improvements of the EO data quality. These improvements can be achieved with the development of new and enhanced algorithms, tuning of auxiliary parameters, processor re-design, instrument calibration or threshold and scaling factor corrections. Huge volumes of data of an EO mission need to be processed. The input data can then be pushed into Cloud storage for the duration of the processing and the reprocessed data can be published on Cloud storage with the extra benefit of possibly being used for on-line data access and distribution.

Routine production. EO routine production includes the generation, archiving and eventually the distribution of high-level products following data- or date-driven scheduled services. These products have different goals and purposes, but

a common point is that it is the same data being processed over and over again during specific moments of the production cycle and then no longer used. With this in mind, Cloud storage could host the amount of input data required for the generation of these products using Cloud computing processing power.

Fast data access for crisis situations. EO data has proven to be an excellent source of information for damage assessment for both natural and man-made disasters. There are several services fully dedicated in providing EO-based crisis mapping. The applications behind it rely on an on-line archive continuously feed with new data acquired taken from rolling-archives and on a number of computing resources to do the processing. The common issues for these applications are the fast access to post-crisis data in near-real time and to archived historical data, and the fact that the computing resources have to be "on-call", i.e. continuously available for processing in case of a crisis event. Moreover, the interest for related data is concentrated in a short time span; this situation may put strain on the on-line data distribution and eventually lead to its unavailability. Since the Cloud provides scalable storage capacity, the EO historical archive and near-real time data could be hosted on such infrastructure. Furthermore, the processing resources needed to provide the higher-level products and maps could also be provided by Cloud computing thus removing the need for maintaining "on-call" computing resources. Furthermore, Cloud storage provides high transfer rates even when numerous users access the same data at the same time.

Distributed Intelligent Maintenance. Another scenario involves the development of a distributed intelligence maintenance tool based on advanced data processing techniques, more precisely based on data mining and artificial intelligence paradigms. A generic platform is intended to be build to target industrial systems and processes in the fields of energy and transport where data-intensive tasks must be performed for analysis, diagnosis, anomaly detection and resource optimization. The data load is usually non-uniform and it presents peaks and stagger increments during intensive workloads or when the park of monitored units increases; for these data-intensive telemetry scenarios, on-demand and distributed storage allocation and computing is needed. Fast deployment on Clouds using mOSAIC's platform will be possible without a considerable investment in application re-writing.

Agent-Based Simulations: from Cluster to Cloud. Agent-based models usually depend on several input parameters, which in turn have several possible values. When the parameter space is explored, the number of simulations required runs easily out of hands. A simulation running tool named MEME [3] is extended in the frame of mOSAIC to allow users to run their parameter sweep distributed using Cloud services. Currently the prototype can utilize a local or remote cluster of machines to distribute the parameter sweep operation on the machines. Similar speed-up can be achieved by accessing Cloud services with the extra advantage of more flexible resource provisioning. Typical business cases for running simulations can be explored, such as asap/money does not matter, minimize cost/ no deadline, anything in between.

An important function of the mOSAIC platform will be that it will be able to compose virtual services by matching applications' requirements to service descriptions provided by Cloud vendors. In iterative experiment designs, in particular, the simulations periodically need several parallel VMs to run experiments, followed by en evaluation phase using a single VM, which determines further computations to carry out. The resource need of such experiments therefore fluctuate between a single VM and several parallel VMs. A semantic description of an actual simulation service requested can describe the varying resource needs of this type of simulations, and will help the platform to find a best possible composition of cloud service providers.

The sweeping through the parameter space of distributed simulations that use several machines themselves requires a special service profile where VMs that cooperate in running one simulation should have a fast Internet connection, while other instances of the same simulation can be run anywhere. That is, parameter sweeping of distributed simulations require several clusters of computer, where there is fast network connection within a cluster, but can be slower connections between the clusters. Semantic description of such simulations will be provided by the users who write the simulations to enable the mOSAIC platform to compose appropriate service considering the different processor, storage, and communication needs of different parameter settings.

4   Main Challenges for the Technical Solutions

From the five main challenges identified in [4] for Cloud, mOSAIC will address application and data portability and interoperability. The use of standard interfaces could allow the flexibility to create new solutions enabled by applications and data that interoperate with each other regardless of a specific Cloud type; in this context vendor-independent application interfaces (as mOSAIC's one) are emerging. Moreover currently the term of virtualisation is understood differently in the context of different technologies such as storage, processing, networking, and a unified resource representation is needed to be considered. Furthermore, mOSAIC investigates the Cloud usage patterns in order to expose them through the proposed API.

The selection of Cloud providers for a particular application is an intricate issue due to the complex business model associated with such computing systems. RESERVOIR project [5] is the first initiative intending to provide open source technology to enable deployment and management of complex services across different administrative domains. mOSAIC proposes a complementary solution, based on software agents and semantic data processing, for Cloud resource negotiations and service level agreements.

In what concerns the data services, data lock-in (lack of standardized APIs) is one of the important issues to deal with. The solution proposed by mOSAIC is to use semantics to identify the application requirements in terms of Cloud data services. We further consider that service requirements of applications can change over time and thus may require amendments of original service requests.

mOSAIC proposes Cloud specific SLAs and QoS requirements that allow negotiation and re-negotiation solutions during run-time.

The starting point in order to build a comprehensive API is to consider the emerging standards and to build the missing pieces. The current emerging standards to be considered are OCCI, UCI, OVF and CDMI. For building the Cloud ontology the starting point is the design of a Cloud taxonomy - the recent proposals for taxonomies (e.g. [8]) are taken into consideration. One of the most challenging goals of the semantic engine is to design and develop semantic-based Cloud services discovery; a prototypical tool will be built based on syntactic and structural schema matching.

The mOSAIC platform has two parts: Resource Broker and Application Executor. The Resource Broker, responsible with resource negotiation and booking, has also two sub-systems: Client interface and Cloud agency. The first one uses an application specification document for describing application resources needs and supplementary resource specification document for requesting supplementary resources by the Application Executor. The second one includes a monitor, a negotiator, a mediator, a service registry and a client semantic engine, provider semantic engines represented by agents, and uses a Cloud ontology and QoS parameters. Moreover, it validates the application specifications and generates a SLA document for resource negotiation and booking, as well as a resource contract used by the Application Executor to access the physical resources for application execution. The Application Executor, in charge with application execution using the resources booked and stated in Resource contract document, has also several sub-systems like: API Execution Engine that is the user's API for accessing the physical resources; Virtual Cluster including the booked resources; Providers wrappers as special connectors ensuring a uniform interface to the Clouds resources available in resource contract; Resource manager ensuring resource availability and management, including Resource scheduler and Resource monitor, and handling supplementary resources request.

Agents will represent different Cloud resources providers and users; e.g. each agent representing a vendor that offers resources understands the requirements specified in the unified representation of resources and translates them in vendor specific requirements. A core set of agents will implement the basic services provided by the Cloud agency. Some agents will be in charge of interacting with users and providers in order to negotiate and to broker the needing resources. The agents of the selected resources during the negotiation will further represent the Cloud resources during the execution of an application, as they will understand the application requests and translate them into vendor specific requests (acts as wrappers). Starting from the MAGDA toolset [6], mOSAIC platform component related to the agent layer will provide facilities to design, develop and deploy agents-based services.

A special platform component is the Virtual cluster (named so by folowing the proposal from [7]), an agent-based resource management facility. The Resource contract will refer this cluster build for an application. The platform will build up a Vc on the basis of a given SLA established within a certain application.

5   Conclusions

The main benefit of using the mOSAIC solutions will be a transparent and simple access to heterogeneous Cloud computing resources and the avoidance of lock-in proprietary solutions. The open-source platform will enables applications to negotiate Cloud services as requested by their users. Using a specific Cloud ontology, applications will be able to specify their service requirements and communicate them to the platform via the innovative API. The platform will implement a multi-agent brokering mechanism that will search for services matching the applications' request, and possibly compose the requested service if no direct hit is found. Cloud-application developers and maintainers will be able to postpone their decision on the procurement of Cloud services until runtime, while end-user applications will be able to find best-fitting Cloud services to their actual needs and efficiently outsource computations. The platform will also facilitate competition between Cloud providers, who, in return, will be able to reach customers they could not reach before.

The developments scheduled in the mOSAIC project have recently started and there is a long way until its promises will become a reality. The first open-source stable version of the API will be publicly available in autumn 2011 and the full platform and proof-of-the concept applications in two years. Until this early date of the project, the Cloud computing offers, emerging standards, usage patterns and semantic solutions were tested or analyzed to identify the inputs for the developments that are scheduled in the near future.

References

1. Cloud Computing Use Case Discussion Group, Cloud Computing Use Cases, White Paper v0.1 (August 2009)
2. Hey, T., Tansley, S., Tolle, K. (eds.): The fourth paradigm: data-intensive scientific discovery, Microsoft Research (October 2009)
3. Gulyás, L., Szemes, G., Kampis, G., de Back, W.: A Modeler-friendly API for ABM Partitioning. In: Procs. ASME (2009)
4. Open Cloud Manifesto (Spring 2009), `www.opencloudmanifesto.org`
5. Rochwerger, B., et al.: The RESERVOIR Model and Architecture for Open Federated Cloud Computing. IBM J. Research & Development 53 (4) (2009)
6. Aversa, R., Di Martino, B., Mazzocca, N., Venticinque, S.: MAGDA: A Mobile Agent based Grid Architecture. J. Grid Computing 4(4), 395–412 (2006)
7. Keahey, K., Tsugawa, M., Matsunaga, A., Fortes, J.A.B.: Sky Computing. IEEE Internet Computing 13(5), 43–51 (2009)
8. Lenk, A., Klems, M., Nimis, J., Tai, S.: What's Inside the Cloud? An Architectural Map of the Cloud Landscape. In: Procs. ICSE, pp. 23–31. IEEE Computer Press, Los Alamitos (2009)