An approach to adapt collaborative architectures to cloud computing

Moisés Dutra[a], Minh Tri Nguyen[b], Parisa Ghodous[b, 1] , Youssef Roummieh[b], Chirine Ghedira[b], and Djamal Benslimane[b]

[a] Catarinense Federal Institute of Science and Technology, Brazil.
[b] University of Lyon 1, France.

Abstract. Companies keep trying various strategies in order to reduce the cost of their computer systems. The trend through which “everything is as a service” seems like the most effective solution used. By benefiting from the advantages of Cloud Computing, companies can minimize the cost of their systems and focus on their core businesses, by placing their IT parts into cloud providers. New companies can build their entire systems based on the clouds from the scratch. For legacy systems, however, the adaptation to the Cloud Computing paradigm remais the more effective solution. By proposing a methodology to adapt collaborative architectures to Cloud Computing, we intend to contribute to the first application of this technology in the industrial world. In order to validate the proposed approach, we have adapted our team’s collaborative system to the clouds.

Keywords. Cloud Computing, Collaborative Systems

1 Introduction

The year 2009 saw the emergence of the term “Cloud Computing” in publications. Historically, the term “Cloud Computing” was first used in 2002 by Amazon [13], a leading e-business, who had invested in a vast machinery, sized to handle the heavy load of orders made on their website at Christmas, but unused for the rest of the year. Sub-sizing their fleets would have caused downtime of their website at the time of the peaks, thus jeopardizing their work during the holidays (a big part of their turnover business). Their idea is to open all these unused resources to businesses to hire them on demand. Since then, Amazon is investing heavily in this area and continues to expand its fleet and services.

[1] ghodous@liris.cnrs.fr – Head of Collaborative Modelling Theme, SOC Group
Laboratory of Computer Graphics, Images and Information Systems (LIRIS)
University of Lyon 1, Bâtiment Nautibus, 43, bd. du 11 Novembre. 1918,
69622 Villeurbanne cedex, France
Tel: +33 (0) 4 72 44 58 84
Fax: +33 (0) 4 72 43 13 12

One difficulty is that there is not between providers a single definition of cloud computing [4]. In this paper, we present the most popular and widely understood definition, given by NIST [5]: Cloud computing is a model for enabling 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. This cloud model promotes availability and is composed of five essential characteristics, three service models, and four deployment models.

With many benefits of cloud computing, experts foresee a major trend in this century when companies apply this technology in their system information [1]. Currently, the major actors in IT world have built their own cloud like Azure of Microsoft, IBM Blue House or Amazon Web Services. There are two main ways to apply the technology of cloud computing for businesses. First, the owners of the company may decide to build a new cloud computing-based system. This decision is handy with startup businesses that have not yet built their local system. However, this solution is not suitable for most companies that have already built their own information system. In this latter scenario, the migration of legacy systems into the clouds is considered a better solution. So the problem is how can we adapt existing systems – especially business applications – to the clouds?

The remainder of this manuscript is organized as follows: Section 2 proposes a methodology for migrating legacy systems into the clouds; Section 3 presents the validation of the proposed methodology, by using a classic collaborative architecture scenario; and Section 4 presents the conclusions of this work and some perspectives on this matter.

2 Methodology

Currently, there are not yet neither developed standards for cloud computing, nor effective strategies for moving business applications into the clouds. Throughput in recent research, users are often lost in the clouds development, which makes moving an application to the clouds quite a challenge [2][3]. We consider it is important to have a methodology for systematizing this process. In this section, we present this methodology for transforming legacy systems – especially commercial applications – into cloud-based systems. In order to achieve so, we propose the following steps.

i) Analyzing the existing system

Analyzing the existing system means determining the system's structure by investigating data, business applications and other system components, in order to identify what will be brought into the clouds and what will be kept locally. We should determine an order through which the externalization will be done.

- Data: Have the highest level of priority for migration, because they are usually the first part thought/built in a system. Business applications rely strongly on data. However, depending on the system architecture, this can be considered differently.

- Business Applications: Many people think we should take everything into the clouds, but the reality is quite the opposite. To ensure the success of the externalization, we consider that it is better to start doing it from the simplest to the most complicated strategies.
- Other components: In general, any system component can be migrated to the clouds. Nevertheless, we can also chose to use an already existing cloud service – in the form of SaaS (Software as a Service) – instead of undertaking the migration.

After this step, we will have a mapping of data, business applications and components, representing those which will be taken to the clouds and those which will be kept locally. The next step consists of finding the suitable clouds for migration.

ii) Choosing the suitable cloud configuration

There are a number of clouds available for migration, each one of them suitable for a specific need. Typically, there are three main types of cloud [7]:

- Public Cloud: Available to the general public or large industries, and it is owned by an organization that sells cloud services. With this solution, we put all the data and business applications on a public cloud, such as the Amazon or Microsoft cloud.
- Private Cloud: Exclusively operated by just one organization. This operation can be managed directly by the organization or be outsourced. This model is very suitable for companies that have great resources available and want to have full control of their data, as they can enjoy the benefits of the cloud as a whole.
- Hybrid Cloud: The infrastructure of the hybrid cloud consists of the composition of two or more private or public clouds, which remain single entities and are bounded by a standard – or proprietary technology – that allows the portability of data and applications.

iii) Designing the new architecture

We can now start conceiving a new architecture for our cloud-based system, which will generally have the following structure.

- Databases: databases can set the clouds in the form of DaaS (Database as a Service). Currently, there are available services [12] for simple data provided by Amazon Simple Database [8], and services for relational data, provided by FathomDB [9] and Amazon [10].
- Business Applications: For business applications, we propose the use of the IaaS (Infrastructure as a Service) or PaaS (Platform as a Service) technology. Currently, there are a number of platforms that support multiple languages, such as .Net, Java, Python, PHP, or Ruby on Rail. These platforms provide APIs to facilitate the application deployment. Moreover, they support several tools to manage those applications on the clouds.
- Other components: this architecture provides us with the possibility of using different clouds for different components, according to our needs.

After determining the new system's architecture, we will perform the migration and some tests, which must consist of a detailed process of checking and validation, in order to ensure the system's workability. These processes will be done according to the following steps.

iv) Choosing the clouds for system deployment

During the creation of the new architecture, it is defined what the system's needs are and what technologies (e.g., PaaS, Saas, and DaaS) will be used to build it. The next step consists of choosing the most suitable clouds to deploy the new system. There are some popular commercial clouds that can be used to this purpose, as told before.

v) Building the new system

We will work here with the same order of components depicted in the first stage, namely databases, business applications, and other components.

- Databases: for simple data, such as XML/OWL files, we can put them on the same place as the business applications. However, for specific databases, such as relational databases, we must put them on a DaaS cloud. Normally, the clouds will provide us with Web interfaces to facilitate the import of databases, as well as provide us with specific information for accessing data from another service.
- Business Applications: There are two ways of deploying business applications on the clouds. First, we create a new project on a PaaS cloud. Then we keep carrying out the migration of the required files until we have the whole project deployed on the clouds. Second, we can create services from local applications and make these services available on the clouds. By using the information database, which has already been deployed, we can establish connections between applications on the clouds and databases elsewhere.
- Other components: for modules such as communication modules, which already exist on the clouds in the form of SaaS, we should just pay for these services according to our needs and obtain information from them, in order to make the connections between these services and business applications (before the deployment of these business applications on the PaaS cloud is done).

In order to reduce the appearance of problems, we should systematically test if everything works well after each step taken. Once we have guaranteed the workability of the system, we can continue externalizing the remaining components.

vi) Defining governance strategies

What should we do if some day the service supplier of the clouds is replaced? How could our system be accessed? We can imagine a situation where we do not know what these changes are, neither how they affect our system. In other words, there would be no governance – or the ability to monitor service changes – and service utilization. In the corporate world, governance means controlling the system. On the clouds, we should control our data and our services to ensure the success of the system. We must know who has access to our data and who can create, delete or change data and services. The idea of governance is to provide command, control and surveillance services, including local services and cloud services. We must clarify the responsibilities of governance with our cloud suppliers. Normally, IaaS clouds provide customers with more level of governance than other clouds do. On the contrary, SaaS clouds provide very little level of governance to their customers. As a consequence, customers must deeply investigate contracts between cloud buyers and suppliers, in order to clearly determine what their policies of governance are.

3 Application of the Methodology

Existing system

Our team has developed a generic and synchronous ontology-based collaborative platform [11]. This platform is based on the collaboration of all actors involved in a design project (designers, customers, engineers, vendors, distributors, etc.), and also comprises a multilayer architecture. First, each participant is called an agent. Agents are grouped according to their knowledge and skill levels. Each group of agents is called an agency. All agents share and publish data through a workspace called "Agency Shared Workspace" (ASW). Similarly, all agencies share and publish their data through a workspace called "Project Shared Workspace" (PSW) [11].

The PSW consists of a blackboard with an ontology module and a knowledge-based reasoning module. The ASW has the same structure that the PSW (Figure 1).

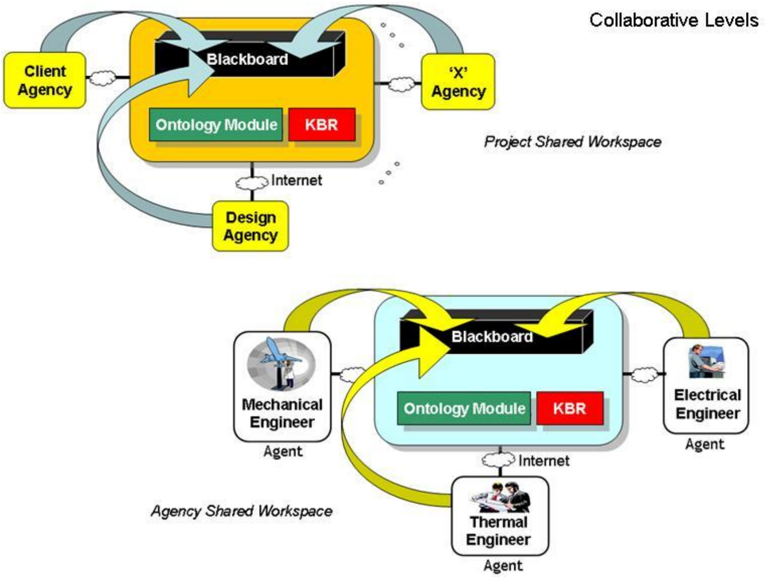

Figure 1. Collaborative platform.

The architecture of this platform is based on a range of agencies collaborating in the process of constructing complex artifacts (e.g., aeronautical products, their processes and services). Each agency is composed of several agents, who work in this process. In the current system, all resources and services are located locally. To take advantage of the cloud computing paradigm, we will put this system on the clouds, by using the methodology presented in Section 2.

New system

We will now apply the proposed methodology to transform our collaborative system into a cloud-based system. As the proposed methodology stands for, we will follow the following steps to build the new system.

1) Analyzing the existing system

The first step during the building of the new system is to analyze the existing system, in order to identify the components to be taken or not to the clouds. In our architecture, we have identified the following components to be taken to the clouds.

- Data: The existing system uses MySQL relational database and OWL-based files. We have then decided to preferentially transfer all these data to the clouds.
- Business applications: business applications consist of the existing applications used by the project manager and users, and the existing applications designed to deal with ontologies. Due to the complexity of the ontology applications, we have decided to transfer the project manager and user applications first. In a second step, further on, after a successful processing of the first one, we will then transfer the ontology applications to the clouds. In the end, the whole collaborative system will have migrated to the clouds.
- Other components: Existing communication services, as the Instant Messaging Service, can be integrated with the clouds as a Service for Enrichment.

2) Choosing the suitable cloud configuration

Currently, we do not have material to build private clouds. Moreover, the cost of construction of clouds remains very expensive. On the other hand, public clouds can perfectly meet our needs at this time. Thus, we have decided to build the new system based on the architecture of public clouds. That is to say, we will put data and business applications onto the clouds and use public cloud services.

3) Designing the new architecture

After selecting the general architecture of the system based on public clouds, we will now associate the system with the new architecture.

- Databases: the relational database will be put on a DaaS cloud, as proposed by our methodology. The OWL-based files will be put on the same place as the business applications, since this kind of data is very specific and this solution will simplify its access by the design application.
- Business Applications: According to the methodology, we will put the business applications on a PaaS cloud. In addition, as the existing system was developed in Java, we have chosen a cloud that contains this feature.

- Other components: In this scenario, modules for communications are the other components. We will use existing SaaS services for this purpose.

4) Choosing the clouds for system deployment

The appropriate clouds must be chosen according to the needs of each system component:

- Database: currently, there are two large DaaS clouds available: the one provided by Amazon and the FathomDB relational database. For this system, we have decided to use FathomDB, due to its simplicity and lower cost.
- Business Applications: for these components, we have decided to choose the PaaS Amazon Web Services cloud, due to three main reasons: it supports Java, it is a large cloud, and it is free for the first use.
- Other components: It has been decided that the communication component will work with mail services, Google chat, Amazon Simple Queue Service, Amazon Simple Notification, and Zoho's "Instant Messaging as a Service".

5) Building the new system

At this stage, we will perform the migration according to the architecture proposed in the previous step.

- Databases: for specific databases such as relational databases, we will put them on a specific DaaS cloud. After seeing details of the information services on the FathomDB website [9], we have decided to choose the "Small Instance" – 150 MB of memory – for the first try. The DaaS will provide us with the information required to access the databases. We now can use the just provided login information to create our database.
- Business Applications: we use Eclipse IDE along with the Eclipse AWS Plugin to implement the business applications and to deploy them onto the Amazon clouds. For the first step, we have chosen to launch an instance of Amazon EC2. We have chosen a small instance for the first try. Through the use of FathomDB database connections, we can establish connections between applications on the cloud and databases outside it.
- Other components: in addition to the existing mail applications, we can integrate Zoho's "Instant Messenger Application as a Service" [6].

After the migration, the new system will work on the clouds as well as they did locally, previously.

6) Security, Governance and elasticity

We have defined very clear security and governance policies to be applied on the chosen clouds. For example, the policies concerning data are very rigid; we can only access data by a specific IP or by an Amazon cloud. Amazon EC2 defines very clear security policies for each specific connection type. Considering governance, clouds provides us with the ability to easily manage our system through Web interfaces. In addition, they provide us with automatic backup services to be used by the system's instance on the clouds.

A big advantage of migration is to extend the system automatically, without human interaction. When necessary, we can also easily expand the database or even pay for more instances of Amazon EC2.

4 Conclusions and Perspectives

Cloud computing technology is rapidly becoming more and more known/used. Possessing benefits such as elasticity and self-service, along with the cost to transfer an entire system to the clouds, companies have now at their disposal a very effective option to reduce the cost of their systems' maintenance. This becomes even clearer in the world's present context, with all the economic difficulties that companies have been facing. In such a context, cloud computing plays a very important role, as it provides a whole new simpler way of designing and managing architectures of enterprise systems. Moreover, building a new system from the scratch is not always an option well seen by companies. Considering this scenario, this work proposes a methodology for transferring companies' legacy systems into the clouds. This methodology is composed of well defined steps of externalization, in a rising level of complexity. We have chosen to validate this methodology by using it during the migration of our team's collaborative system to the clouds.

Next steps of this work consist of filling the existing gaps in the methodology, especially the security and governance issues. Moreover, a prototype that can help transfer the system automatically to the clouds is a goal that we want to achieve as well.

5 References

[1] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia – Above the Clouds: A Berkeley View of Cloud Computing, Armburst, 2009.

[2] A. Chandra and J. Weissman, University of Minnesota – Nebulas: Using Distributed Voluntary Resources to Build Clouds, 2009.

[3] A. T. Velte, T. J. Velte, and R. Elsenpeter – *Cloud Computing - A practical approach*, 2009.

[4] *Twenty-One Experts Define Cloud Computing*: http://cloudcomputing.sys-con.com/node/612375, in February 2011.

[5] Peter Mell and Tim Grance – A NIST Notional Definition of Cloud Computing.

[6] http://chat.zoho.com, in February 2011.

[7] Cloud Computing Use Case Discussion Group – *Cloud Computing Use Cases.*

[8] http://aws.amazon.com/simpledb/, in February 2011.

[9] http://fathomdb.com, in February 2011.

[10] http://aws.amazon.com/rds/, in February 2011.

[11] M. Dutra, P. Ghodous, O. Kuhn, N. M. Tri – *A Generic and Synchronous Ontology-based Architecture for Collaborative Design*. Concurrent Engineering, Research and Applications 18(1):65-74, Sage, ISSN 1063 293X, 2010.

[12] Luis Felipe Cabrera, Christopher Kurt, Don Box (Microsoft Corporation) – *Introduction à l'architecture de services Web et ses spécifications WS*. Octobre 2004.

[13] http://www.cloudconsulting.fr/index.php/fr/le-cloud-computing/historique, in February 2011.