Recent Advances in Technology and Engineering (RATE-2017)
6th National Conference by TJIT, Bangalore

International Journal of Science, Engineering and Technology
An Open Access Journal

Bursting of Service-Based Applications Efficiently in Hybrid Clouds

[1]Navya Uppar, [2]Indudhar B H, [3]Pratheek, [4]Mamatha K

Abstract

Enterprises are extensively using hybrid clouds to deploy and run applications. It provides and manages software and hardware resources within the enterprise and additional resources whenever required is provided externally by public clouds. In this later case, deployment of new applications consists in choosing in choosing a placement of some components in the private cloud and some others in the public cloud. In this paper, we propose a new efficient algorithm for service-based applications modelled that can be not only be described as behavior-based but also as architecture-based compositions of services.

Keywords: Bursting, Hybrid clouds, Cloud computing, Forward Procedure, Backward procedure, and Refinement procedure.

1 Introduction

Cloud computing is a new delivery model for IT services. It typically involves infrastructure, platform and software levels. It addresses fundamentals like virtualization, scalability, interoperability, quality of service and failover mechanism. Cloud computing differs from traditional environments on the fact that it (1) is massively scalable, (2) can deliver different levels of services to customers outside the cloud, (3) economics of scale, (4) dynamically configured and (5) can be delivered on demand.

Cloud environments can be public, private and hybrid. A public cloud is a cloud that provides cloud resources and services to the public. A private cloud is a cloud that is owned or leased by an enterprise. A hybrid cloud is a composition of two or more clouds. In this paper, a hybrid cloud is a composition of one public and one private cloud. Such cloud, is an environment in which an enterprise has its own private cloud that provides and manages some internal resources and only uses public cloud when needed. In cloud environments, the diversity of requirements for hosting services in the cloud makes the management of cloud applications and resources a challenging task. In this paper, we are interested in application division in clouds and particularly service-based application in hybrid clouds in order to attain cost efficiency. SBAs consists in composition of components providing business services. The composition cab be architecture-based or UML component diagram or behavior-based or business process model and notation. When deploying SBAs in a hybrid cloud, their cloud, their division also known as bursting in public and private clouds consists in choosing components/service to be deployed in the public cloud and those to be kept in the private cloud. Cloud bursting is used when there are spikes in workload in private cloud. Several parameters would be involved to make the division decision. In this paper, we are mainly interested in communication costs and we do not deal with security, privacy and other related properties. While we believe that these issues are important, the communication and hosting costs criteria discussed here are complex enough to deserve a separate treatment.

The existing work on placement of resources as different levels such as software as a service (a.k.a. SaaS), platform as a service (a.k.a. PaaS), infrastructure as a platform (a.k.a. IaaS). They are not suitable to the considered problem of this paper. They do not consider communication between services or hybrid situation. In this paper, we propose a new and more efficient algorithm that provides solutions that are very close to cost optimization.

2 State of the Art

The issue of placement of resources (virtual machines, web servers, software components and data) in clouds has been tackled from different points of views (IaaS, PaaS, and SaaS), while taking into account different types of clouds (private/public, hybrid and federation of clouds) and different types of conditions (hosting, communication, QoS, etc.).

2.1 Placement of Resources in Private/Public Clouds

We present some approaches for placement at the IaaS level [2], at the PaaS level [3], and at the SaaS level [4], [5].

In [2], the authors provide algorithms to allocate platform resources for service placement. These algorithms take into account communication resources, hosting resources and QoS. The considered resources are mainly instances of services that should be deployed on top of web servers.

In [3], the authors present an approach for optimizing the placement of virtual resources at the level of infrastructure across multiple clouds. They developed a broker that allows the deployment and management of virtual resources based on the placement results.

In [4], the authors present a penalty-based genetic algorithm for the placement of components and data that optimize the SaaS performance.

In [5], the authors consider a placement problem for SaaS components in a multi-tenant architecture. Components are placed in a set of available platform nodes while optimizing the resource usage in each node.

All these approaches are suitable for the placement in private or public cloud but not for hybrid clouds. In hybrid clouds cost of services depends on placement of services in which public and private communications must be differentiated. The problem is to minimize the communication in the placement. The communication is of the same type whereas in our situation and context the communication is of two types hybrid communication and public communication. Here its formulation and its cost are different from those of the above-mentioned approaches.

2.2 Bursting of Resources in Hybrid Clouds

Here we present approaches that optimize cost of resources in hybrid clouds.

Scheduling model for optimizing virtual cluster placements across clouds was proposed in [6] and [7]. These placements are based on different points such as optimization (average price), user constraints (performance and environment) and environmental conditions. These approaches are not applicable as they do not consider communication costs.

3 Deployment of in Hybrid Clouds

In this section, we define service-base application that we consider in this paper.

3.1 Service-Based application

SBAs consists in assembling of a set of services using appropriate service composition that can be architecture-based or behavior-based. Here we define these two types of compositions.

3.1.1 Behavior-Based Composition

This is also known as structured process which consists of a set of nodes and transitions between them.

Definition 1 (structured Process). A structured process is inductively defined as follows:

- A process consisting of a single service is a structured process. This service is both initial and final.
- Let X and Y be structured processes. The concatenation of these processes where the final node X has a transition to initial node Y, then also is a structured process. The initial node of this process is the initial node of X and its final node of Y.
- Let X_1.........X_n be structured process and let j be an Or-join and an Or-split. The process with a initial node s and final node j and transition between s and the initial node X and between the final node of X and j is then also a structured process. The initial node of this structured process is s and its final node is j.
- Let X_1.........X_n be structures process and let j be an And-join and s an And-split. The process with a initial node s and final node j and transition between s and the initial node of X and between the final node of X and j is then also a structured process. The initial node of this structure process is as and its final node is j.

3.1.2 Architecture-Based Composition

A SBA composed using an architecture-based composition can be described as a set of linked components.

The example is a composition of four services. The store service provides the interface for the online store. The catalog service provides the item prices for the catalog items. The currency convertor service provides the currency convertor for the catalog service. The shopping cart service is used to include items chosen from the catalog service.

4 Problem Statement

We are interested in this paper in deployment of SBAs in hybrid clouds that here we consider is a composition of one private and one public cloud.

In this paper, we are deciding about services to be deployed in the public cloud based on some parameters we have given below. The need of resources from public cloud can be quantified in terms of amount of hosting units. This quantity refers as hosting quality and noted HQ. When it is necessary to use the public cloud, one has to decide about the services that should be deployed in public cloud and private cloud.

In addition to their hosting cost, communication cost is generated by the services to be deployed in the public cloud and communication between them and services of applications deployed in the private cloud. These hosting and communication costs have to be supported by the enterprise.

We do not consider here the communication and hosting costs within the private cloud. We are interested in the costs to be paid to the public cloud provider. We consider that almost ass the private resources are consumed. They have to be supported in any case. Hence decide about the services to be deployed in the private cloud and the public cloud is to minimize costs of hosting in public cloud and communications between the private and the public cloud and in the public cloud. But this minimization problem is subject to a constraint.

5 Algorithms for An Efficient Bursting

We propose two algorithms for the bursting of service-based applications in hybrid clouds. These algorithms are based on three procedures called Forward procedure, Backward Procedure and Refinement procedure.

5.1 Forward, Backward and Refinement Procedures

The Forward procedure consists in choosing services from the private cloud to be moved to the public cloud. The Backward procedure tries to repair the result of the first step by moving services from the public cloud back to the private cloud. The refinement procedure tries to repair the result for the second step by moving services again from the private to the public cloud.

Algorithm 1. Forward procedure

Require: (S,E,h,c,l): SBA graph
Ensure: Public: set of application nodes in the public cloud
Ensure: Private: set of application nodes in the private cloud
1: Private $\leftarrow$ S;
2: Public $\leftarrow \varnothing$;
3: While h(Public) < HQ do
4: mincost $\leftarrow +\infty$;
5: for all $s_i \in$ Private where s_i is a service do
6: $cost(s_i) \leftarrow \beta_1 \times icc(s_i) + (\beta_2\text{-}\beta_1) \times occ(s_i) + \alpha \times h(s_i)$;
7: if $cost(s_i)$ < mincost then
8: mincost $\leftarrow cost(s_i)$;
9: $s_{min} \leftarrow s_i$
10: end if
11: end for
12: Private $\leftarrow$ Private \ $\{s_{min}\}$;
13: Public $\leftarrow$ Public $\cup \{s_{min}\}$
14: end while

Algorithm 2. Backward procedure

Require: (S,E,h,c,l): SBA graph
Ensure: Public: set of application nodes in the public cloud
Ensure: Private: set of application nodes in the private cloud
Ensure: Public: set of application nodes in the public cloud
Ensure: Private: set of application nodes in the private cloud
1: MoveBack $\leftarrow$ true;
2: while MoveBack do
3: mincost $\leftarrow \infty$;
4: MoveBack $\leftarrow$ false;
5: for all $s_i \in$ Public where h(Public) – h(si) $\geq$ HQ do
6: $cost(s_i) \leftarrow (\beta_1\text{-}\beta_2) \times icc(s_i) - \beta_1 \times occ(s_i) - \alpha \times h(s_i)$;
7: if $cost(s_i)$ < mincost then
8: mincost $\leftarrow cost(s_i)$;

$S_{min} \leftarrow s_i$
end if
end for
if mincost < 0 then
Private $\leftarrow$ Private $\cup \{s_{min}\}$;
Public $\leftarrow$ Public$\setminus\{s_{min}\}$;
MoveBack $\leftarrow$ true;
end if
end while

Algorithm 3. Refinement procedure

Require: (S,E,h,c,l): SBA graph
Ensure: Public: set of application nodes in the public cloud
Ensure: Private: set of application nodes in the private cloud
Ensure: Public: set of application nodes in the public cloud
Ensure: Private: set of application nodes in the private cloud
Move $\leftarrow$ true;
while Move do
mincost $\leftarrow +\infty$;
Move $\leftarrow$ false;
for all $s_i \in$ Private do
cost(s_i) $\leftarrow \beta_1 \times icc(s_i) + (\beta_2-\beta_1) \times occ(s_i) + \alpha \times h(s_i)$;
if cost(s_i) < mincost then
mincost $\leftarrow$ cost(s_i);
$s_{min} \leftarrow s_i$
end if
end for
if mincost < 0 then
Private $\leftarrow$ Private $\setminus \{s_{min}\}$;
Public $\leftarrow$ Public $\cup \{s_{min}\}$;
Move $\leftarrow$ true;
end if
end while

Algorithm 4. FB procedure

Require: (S,E,h,c,l): SBA graph
Ensure: Public: set of application nodes in the public cloud
Ensure: Private: set of application nodes in the private cloud
Public, Private $\leftarrow$ Forward (Graph)
Public, Private $\leftarrow$ Backward (Graph, Public, Private)

5.2 Forward-Backward (FB) Algorithm

The Forward-Backward(FB) algorithm is a first algorithm for bursting that calls Forward procedure first and then calls Backward procedure.

5.3 Forward-Backward-Refinement (FBR) Procedure

Algorithm 5 is an efficient approach known as Forward-Backward-Refinement algorithm also known as FBR in short. Here it calls Forward procedure first, Backward procedure second and Refinement procedure third. Finally, our algorithm calculates the optimized costs.

Algorithm 5. FBR algorithm

Require: (S,E,h,c,l): SBA graph
Ensure: Public: set of application nodes in the public cloud
Ensure: Private: set of application nodes in the private cloud
Public, Private $\leftarrow$ Forward(G)
$Public_1$, $Private_1$ $\leftarrow$ Backward(G, Public, Private)
$Public_1$, $Private_1$ $\leftarrow$ Refinement(G, $Public_1$, $Private_1$)
$Public_2$, $Private_2$ $\leftarrow$ Refinement(G, Public, Private)
Public, $Private_2$ $\leftarrow$ Backward(G, $Public_2$, $Private_2$)
$Private_3$ $\leftarrow$
$Public_3$ $\leftarrow$ Graph.S
$Public_3$, $Private_3$ $\leftarrow$ Backward(G, $Public_3$, $Private_3$)
$Public_3$, $Private_3$ $\leftarrow$ Refinement(G, $Public_3$, $Private_3$)
choose i s.t. cost($Public_i$, $Private_i$) $\leq$ cost($Public_j$, $Private_j$), J=1, 2, 3
Public $\leftarrow$ $Public_i$
Private $\leftarrow$ $Private_i$

Conclusion

Here we have solved a problem related to optimal costs of a service-based applications to be deployed in hybrid clouds. This consists in determining what services are put up in public and in private cloud in order to minimize hosting and communication costs. In private cloud, private services are deployed and in public cloud, public services are deployed. In [1], we have the FB algorithm and we have suggested in this paper a new algorithm with acceptable complexity. We have developed a graph generator that gives us with different realistic graphs that cover architecture-based compositions and structured process compositions.

Results show that our FB algorithm has a good behavior which represent service-based applications described as behavior-based compositions. They show that our FRB algorithm has a good behavior for structured process and not only for architecture-based composition. Compared to the FB algorithm, our new FBR algorithm reduces the average of the percentage of loss by a half for graphs that represent architecture-based compositions and by two-thirds for graphs that represent behavior- based compositions. The work we achieved is very promising and several perspectives are under study. In this paper, we consider hosting and communications costs as criteria for service bursting of SBAs in hybrid clouds. In our future work, we will consider additional parameters such as security and privacy.

Acknowledgements

We acknowledge that the content in this paper is prepared by doing a survey of different thesis and papers on conjunctive keyword search with Hybrid clouds, Cloud Computing and service based application.

References

[1] F. B. Charrada, N. Tebourski, S. Tata, and S. Moalla, "Approximate placement of service-based application in hybrid clouds", in Proc. IEEE 21st Int. Workshop Enabling Techol. Infrastructure Collaborative Enterprises, 2012, pp. 161-166.

[2] J. Famaey, T. Wauters, F. D. Turck, B. Dhoedt, and P. Demeester, "Towards efficient service placement and server selection for large-scale deployments," in Proc. 4th Adv. Int. Conf. Telecommun.,2008, pp. 13–18.

[3] J. Tordsson, R. S. Montero, R. Moreno-Vozmediano, and I. M.Llorente, "Cloud brokering mechanisms for optimized placementof virtual machines across multiple providers," Future Gener. Comput. Syst., vol. 28, no. 2, pp. 358–367, 2012.

[4] A. Sailer, M. R. Head, A. Kochut, and H. Shaikh, "Graph-based cloud service placement," in Proc. Int. Conf. Serv. Comput., 2010, pp. 89–96.

[5] T. Kwok and A. Mohindra, "Resource calculations with constraints,and placement of tenants and instances for multi-tenant saas applications," in Proc. 6th Int. Conf. Serv.-Oriented Comput., 2008, pp. 633–648.

[6] J. Lucas-Simarro, R. Moreno-Vozmediano, R. Montero, and I. Llorente, "Dynamic placement of virtual machines for cost optimization in multi-cloud environments," in Proc. Int. Conf. High Perform. Comput. Simul., Jul. 2011, pp. 1–7.

[7] J. Lucas-Simarro, R. Moreno-Vozmediano, R. Montero, and I. Llorente, "Scheduling strategies for optimal service deployment across multiple clouds," Future Gener. Comput. Syst., vol. 29, pp. 1431–1441, 2012.

Author's details

1Pratheek L UG student, CSE, T John Institute of Technology, Karnataka, India, pratheeklingaraju@gmail.com

2Navya Uppar UG student, CSE, T John Institute of Technology, Karnataka, India, upparnavya@gmail.com

3Indudhar B H, UG student, CSE, T John Institute of Technology, Karnataka, India, indudharbh19@gmail.com

4Mamatha K, Assistant professor, CSE, T John Institute of Technology, Karnataka, India, mamathak@tjohngroup.com