Sumedha Sood

Department of Computer science and applications M.D. University, Rohtak, Haryana, India

Automatic Selection and Ranking of Cloud Providers using Service Level Agreements

Preeti Gulia, Department of Computer science and applications M.D. University, Rohtak, Haryana, India

ABSTRACT

The popularity of cloud computing has increased by tremendous amounts within last few years. By replacing huge amount of traditional IT infrastructure in very short time, cloud computing has brought itself to a supreme position in IT industry. Various factors such as easy availability, pay as you use model and cost effective nature of cloud computing have helped it to achieve this position. However, with the rising popularity of cloud a large number of providers have readily invested in the same. As a result of this the numbers of providers offering cloud services have increased rapidly. Thus, it becomes extremely difficult for a user to select the best cloud manually. Thus keeping the above issue in the mind, this paper proposes a framework that dynamically selects the best cloud as per user requirements and thereby removes the overhead of cloud selection from the user. The selection and ranking of clouds is done by matching user requirements with Service Level Agreements offered by different clouds according to user assigned weights.

General Terms

Automatic selection of clouds

Keywords

Business Level Objectives, Cloud Computing, Service Level Agreement

1. INTRODUCTION

Cloud computing is a newly developed computing model which permits a service provider to conveniently rent access to fully developed software applications, software development and deployment environment and to various computer infrastructural resources such as storage, databases, etc. [1] Many companies such as Google, Amazon, GoGrid, etc offer services via clouds. Service level agreements form an important part of cloud computing paradigm. Service level agreement can be defined as short legal documents that contain various promises regarding performance of services that a provider intends to provide. [1][2] The importance of service level agreement in cloud computing can be clearly illustrated from the fact that in cloud computing a consumer has to outsource his entire data to the provider. As a result of this he becomes completely dependent on the provider. Thus for a customer to entirely trust the provider, there should be some kind of legal relationship between the two. Service level agreements define the required legal relationship between provider and consumer. They contain in detail commitment about various services that a provider offers. This legal binding thus helps a consumer to trust a provider. [2]

1.1 Term of service level agreement

Following points list some important factors that must be considered at the time of defining terms of SLA [2] [3]

- Business Level Objectives The cloud consumer before selecting any cloud service should first specify the main objective behind using cloud services. This would help in deciding which services he precisely requires and in how much quantity.
- 2) Responsibilities of both parties There should be a balance between responsibilities of cloud consumer and cloud provider. Both the consumer and provider should be fully aware of their responsibilities. These responsibilities should be thoroughly discussed at the time of defining terms of SLA. Those areas where a provider does not want to hold himself responsible for any violation should be also clearly specified in SLA.
- 3) Disaster Recovery Since the cloud consumer has to risk his entire data on the cloud hence he must make sure that appropriate measures taken by the provider to ensure disaster protection. These measures should be considered at the time of defining SLA.
- 4) Redundancy In order to ensure the reliability of consumer's data the provider must ensure appropriate backups for it. The provider can keep redundant systems for the same. The customer should be well aware of it.
- 5) Maintenance It is provider's responsibility to ensure that all his systems are in good working conditions. Thus time to time maintenance of clouds is a must. Hence, the provider must take appropriate maintenance measures to ensure proper service delivery to customer. The customer should also be aware of those measures so that his business does not get affected. If the customer is well informed he can easily adjust his work beforehand.
- 6) Brokers and resellers If the contract between cloud provider and customer is established by means of a broker, the customer should be well aware of all the services offered by actual cloud provider. He should be well aware and satisfied with all the policies of the actual cloud provider at the time of defining SLA.
- 7) Data location If there is a requirement that the customer's data can be stored at a specific location, the provider must be fully aware of it. The provider must fulfil the above requirement by guarantying to store data at a specific location as required and also in addition providing the customer an ability to audit the system.

Since Service level agreement of a cloud contains all the details of various services that a provider has to offer, hence a consumer can easily select the most suitable cloud for their business by matching the service level agreements with their requirements. However, as the numbers of providers offering cloud services are increasing rapidly in the market, this

manual process of matching becomes very tedious and time consuming to be performed by a user. [2][4] With absence of appropriate standards, it is quite possible that a user may waste money on a cloud that does not satisfy his requirements in the best possible manner. Thus keeping this issue in mind this paper proposes a dynamic algorithm that would select the cloud for a user that would be best as per his requirements. It is referred to as dynamic since the selection would be done dynamically on the basis of user requirements.

The rest of paper is organized as follows. In section 2, we describe related work in the field of automatic cloud provider selection for provider. Section 3 describes the proposed work with the help of proposed algorithm. Section 4 discusses in tabular form data used in algorithm. Section 5 discusses the results. The paper finally concludes in section 6.

2. RELATED WORK

Similar work of selecting the best cloud has been done previously also. Recently (2011) an algorithm was developed by Tejas Chauhan et al for the purpose of selecting the best provider automatically by matching user's requirement (requirement model) with cloud's Service Level Agreement (Cloud Capability Model). The above process of matching two models was done on the basis of various service level agreement parameters. Total nine parameters namely Virtual machine, Storage Capability, Memory capability, Ethernet, Availability, Processor speed, response time Server reboot, Service Credit were considered. [5].It was developed by using Jena API of JAVA.

Then, a framework named SMICloud was developed by Saurabh Kumar Garg, Steve Versteeg and Rajkumar Buyya in their paper titled "SMICloud: A Framework for Comparing and Ranking Cloud Services". In this framework various clouds were ranked on the basis of service measurement index using AHP based ranking mechanism [6].

3. PROPOSED WORK

The main motive of this work is to provide the user with that cloud that delivers best services according to his requirements. This approach works by taking difference between how much provider offers and how much user requires. The difference will tell how much a cloud provides in addition to minimum user requirement. The more is the difference, the greater is the benefit of user. The difference is then raised to power with weight assigned to the parameter which is contained in user weight table.

3.1 Algorithm

This work has been implemented in JAVA using My SQL as backend. Regarding this work three tables have been used: cloud provider table [4], user requirement table [4] and user weight table.

The algorithm works as follows:

- Find the eligible clouds by matching services offered by clouds (as contained in cloud provider table) with user requirements (as contained in requirement table). The clouds that provide the service equal to or above the user requirements are eligible and remaining are not eligible.
- 2. Calculate the points for each cloud using following formula:

• Points(cloud₁) =
(cloud₁ - Requirement_x) =
$$(C_{11} - R_{X1})^{w_1}$$
 +

 $(C_{12} - R_{X2})^{w_2} + \dots + (R_{xc} - C_{1c})^{w_c} + \dots + (C_{1m} - R_{xm})^{w_m}$

- $\begin{aligned} Points(cloud_{2}) &= \\ (cloud_{2} Requirement_{x}) &= (C_{21} R_{X1})^{w_{1}} + \\ (C_{22} R_{X2})^{w_{2}} + \cdots + (R_{xc} C_{2c})^{w_{c}} + \cdots + \\ (C_{2m} R_{xm})^{w_{m}} \end{aligned}$
- Points(cloud₃) = (cloud₃ - Requirement_x) = $(C_{31} - R_{X1})^{w_1} +$ $(C_{32} - R_{X2})^{w_2} + \dots + (R_{xc} - C_{3c})^{w_c} + \dots +$ $(C_{3m} - R_{xm})^{w_m}$.
- $Points(cloud_n) =$ $(cloud_n - Requirement_1) = (C_{n1} - R_{X1})^{w_1} +$ $(C_{n2} - R_{X2})^{w_2} + \dots + (R_{xc} - C_{nc})^{w_c} + \dots +$ $(C_{nm} - R_{xm})^{w_m}$

Where,

 $x = 1 \dots p$ (There are total p requirements)

 $Rx_1 =$

value of 1^{st} parameter of x^{th} requirement

 $Rx_2 =$ value of 2nd parameter of x^{th} requirement

 $\dots Rx_c =$ value of cost parameter of x^{th} requirement

 $Rx_m =$

value of m^{th} parameter of x^{th} requirement

(Total there are m parameters)

 $cloud_1 = 1^{st} eligible cloud$

 $cloud_2 = 2^{nd}$ eligible cloud

 $\dots cloud_n = n^{th} \ eligible \ cloud$

 $C_{11} =$

value of 1^{st} parameter of 1^{st} eligible cloud

 $C_{12} =$

value of 2^{nd} parameter of 1^{st} eligible cloud

 $...C_{1c} =$ value of cost parameter of 1^{st} eligible cloud

 $\ldots C_{1m} =$

value of m^{th} parameter of 1^{st} eligible cloud

 $C_{21} =$ Value of 1st parameter of 2nd eligible cloud

 $C_{22} =$ value of 2nd parameter of 2nd eligible cloud

 $... C_{2c}$ = value of cost parameter of 2^{nd} eligible cloud

 $...C_{2m} =$

value of m^{th} parameter of 2^{nd} eligible cloud

 $C_{n1} =$ $value of 1^{st} parameter of n^{th} eligible cloud$ $C_{n2} =$ $value of 2^{nd} parameter of n^{th} eligible cloud$ $\dots C_{nc} =$ $value of cost parameter of n^{th} eligible cloud$ $\dots C_{nm} =$ $value of m^{th} parameter of n^{th} eligible cloud$ (Total there are n eligible clouds) $W_1 = Weight of 1^{st} parameter$ $W_2 = Weight of 2^{nd} parameter$ $\dots W_m = Value of m^{th} parameter$

- Arrange these points in descending order and rank them. The cloud that gets the highest points gets the 1st rank and is the best cloud for given x requirement. The step 2 will be repeated for all x requirements, where x=1...p.
- 4. Repeat above steps for all requirements.

Every parameter in requirement table $(R_{x1}...R_{xm})$ represents the minimum value that a user requires. However, the cost is the maximum value expected by a user. For example \$2200 value of a cost parameter in a given requirement means that maximum value a user is willing to pay is \$2200 and every value below this will be accepted. Hence, for every parameter except cost, user requirement value is subtracted from cloud provider value (since, cloud provider value > user requirement value). However, for cost parameter, cloud provider value is subtracted from user requirement value (since, user requirement value > cloud provider value). If we subtract user requirement value from cloud provider value for the cost parameter, it will result in negative results and would decrease the points of given provider and thus produce wrong results. Hence, cloud provider value is subtracted from user requirement value when the parameter is cost.

The above point is explained more clearly in following example: A user requires minimum 100 GB storage and 20

GB RAM and his maximum budget is \$3 (per hour). Cloud A offers 500 GB and 32 GB RAM at \$1.8 and Cloud B offers 600 GB and 40 GB RAM at \$2.8. Thus, points would be calculated by adding (500 - 100) GB and (32- 20) (and cost) for cloud A and (600 - 100) GB and (40-20) (and cost) for cloud B. However if we do the same thing for the cost parameter also i.e. (1.2 - 3) and (2.8 - 3), it would provide us with negative answer and if these cost points are added, cloud B which is more expensive than cloud A will be benefited. This is so because the cloud with maximum points wins and if above is done less points will be subtracted from B as compared to A and it will be benefited more. Thus for the cost parameter user requirement is subtracted from cloud provider cost to produce correct answers(Each difference is raised to the power of user weight and then added)

Following service level agreement parameters are considered:

- 1) Security
- 2) Availability
- 3) Processor cores
- 4) Processor speed
- 6) RAM
- 7) Cost (hourly/monthly basis)
- 8) Storage
- 9) Service credit

4. TABLES USED IN ALGORITHM

This algorithm makes use of three tables: cloud provider table, requirement table and user weight tables. All these tables are stored in a database created by using MySql

1) Cloud provider table

Cloud provider table [4] contains the amount of service that each provider guarantees to provide for the above mentioned SLA parameters. All the above information is collected from websites of the providers. Since no information about security was provided, hence it is assumed. 10 cloud provides have been used namely Google compute engine [7][8], Rackspace[9][10], HP[11][12], GoGrid[13][14], Opsource[15][16], Nephoscale[17][18], Bitrefinery[19][20], Windows azure[21][22], saavisdirect[23][24], Joyent[25][26].

Cloud	Security	Availab	Processor	Processor	Cost (per	Cost (RAM	Storage	Service
Provide		ility	speed(per	Cores	hour basis)	monthly			credit
r			core)*(approx)			basis)			
Google	22 hours	99.95%	Not	8	\$1.06	Not	30 GB	3540GB	50%
Comput			Mentioned			Mentione			
e Engine						d			
Rackspa	23 hours	100%	2.3 GHz	8	\$1.20	\$876.6	30 GB	1228GB	100%
ce									
Нр	22 hours	99.95%	2.7 GHz	8	\$1.12	\$817.6	32 GB	960 GB	30%
GoGrid	24 hours	100%	2.9 GHz	24	\$1.92	\$870	24 GB	1228 GB	10,000
									%
OpSour	22 hours	100%	2.1 GHz	8	\$2.17	\$1584.10	64 GB	2500GB	100%
ce									
Nephosc	22 hours	99.95%	2.4 GHz	8	Not	\$1499	144 GB	1000 GB	25%
ale					Mentioned				
Bitrefin	23hours	100%	2.1 GHz	4	Not	\$246.20	8 GB	150 GB	100%
ery					Mentioned				
Window	22 hours	99.95%	1.6 GHz	8	\$1.80	\$1399.00	56 GB	2040 GB	25%
s Azure									
Savvisdi	22 hours	99.9%	2.67GHz	8	Not	\$329.37	8 GB	500 GB	20%
rect					Mentioned				
Joyent	22 hours	100%	Not	16	\$2.80	\$2044	80GB	2048GB	100%
			Mentioned						

Table I (cloud provider table) [4]

2) User requirement table :-

User requirement table [4] lists the minimum service each user requires for each SLA parameter. Only the cost parameter denotes the maximum budget of the user. Only those clouds that provide service above that mentioned in requirement table are considered as eligible while others are considered as not eligible. Even if a cloud fails to provide service above that mentioned in requirement table for a single parameter, it would be considered not eligible. The fields that contain not required means that the user does not require service for that parameter.

For example, In Requirement 1, 20 Hours of security means minimum security required by user is 20 hours. Similarly 99.5% Availability means user requires minimum 99.5% availability and hence clouds that offer this much or more availability are considered eligible. However, \$2000 cost means that a user is willing to pay cost below the given value.

Requirements	Security	Availabili	Processor	Processor	Cost	RAM	Storage	Service credit
		ty	cores	speed(per core)* ^{(approx}				
Requirement1	20 hours	99.5%	4	2.1 GHz	\$2000	25 GB	800 GB	25%
Requirement 2	20 hours	98.5%	8	2.4 GHz	\$2200	8 GB	400 GB	20%
Requirement 3	Not Required	95%	4	1.6 GHz	\$2500	Not Required	500 GB	Not Required
Requirement 4	Not Required	90%	4	2 .1GHz	\$1500	16 GB	800 GB	Not Required
Requirement 5	15 Hours	100%	4	2.1GHz	\$3(Per Hour)	30GB	800GB	10%
Requirement 6	22 Hours	90%	4	1.0 GHz	\$2200 (Per month	16 GB	Not Required	10%
Requirement 7	Not Required	95.5%	Not Required	Not Required	\$3(Per hour	Not Required	Not Required	50%
Requirement 8	20 Hours	90%	4	1.8 GHz	\$2500 (Per month)	8 GB	100 GB	20 %
Requirement 9	20 hours	99.5%	8	1.7GHz	\$3 (Per hour	20 GB	400 GB	25%
Requirement 10	20 hours	Not Required	Not Required	Not Required	\$2200 (Per month)	10 GB	400 GB	Not Required
Requirement 11	Not Required	Not Required	Not Required	Not Required	\$2(Per hour	Not Required	600GB	20 %
Requirement 12	Not Required	Not Required	8	Not Required	\$3 (Per hour)	Not Required	1000 GB	Not Required
Requirement 13	Not Required	100%	8	Not Required	\$2500 (Per month)	16 GB	100 GB	Not Required

Table II (User requirement table) [4]

3) User weight table

User weight table contains the weights that a user assigns to different parameters in order of their importance. The users have to assign weights in the range of 0 to 4.

- a. If a given parameter is the most important parameter, it would be assigned weight 4. Highest preference will be given to these parameters at the time of selecting clouds.
- b. If a given parameter is important (but not as important as the parameters assigned weight 4), it would be assigned weight 3.
- c. If a given parameter is moderately important it would be given weight 2,

- d. If a give parameter is required but not important, it would be assigned weight 1,
- e. If a parameter is not required it would be assigned weight 0,

Any number of parameters can be assigned same weights. For example, in the 1st Requirement, all parameters: security, availability and processor speed are most important parameters hence, they are assigned weights 4, next highest important parameter is processor speed (weight = 3), then RAM and storage are important (weight = 2) and the least important parameter are service credit and cost (priority=1).

Requirements	Security	Availabi lity	Processor cores	Processor speed(per core)	RAM	Storage	Service credit	Cost
Requirement 1	4	4	4	3	2	2	1	1
Requirement 2	3	2	4	4	1	1	4	1
Requirement 3	0	4	1	4	0	2	0	3
Requirement 4	0	4	4	4	1	3	0	2
Requirement 5	1	3	3	3	2	4	4	1
Requirement 6	3	4	4	4	2	0	1	3
Requirement 7	0	3	0	0	0	0	3	4
Requirement 8	4	4	2	4	4	1	4	3
Requirement 9	3	4	4	2	1	3	3	2
Requirement 10	1	0	0	0	2	4	0	3
Requirement 11	0	0	0	0	0	2	4	3
Requirement 12	0	0	4	0	0	3	0	4
Requirement 13	0	3	2	0	1	4	0	4

Table III (user weight table)

5. RESULTS

The following table shows the results of the above algorithm.

REQUIREM	Google	Racksp	HP	GoGrid	Opsour	Nephosca	Bit	Windows	Savvisd	Joyent
ENTS	Compute	ace			ce	le	Refiner	Azure	irect	
	Engine						У			
Requirement	Not	2 nd	4 th	Not	1 st	3 rd	Not	Not	Not	Not
1	Eligible			Eligible			Eligible	Eligible	Eligible	Eligible
Requirement	Not	Not	2 nd	1 st	Not	3 rd	Not	Not Eligible	4 th	Not
2	Eligible	Eligible			Eligible		Eligible			Eligible
Requirement	Not	4 th	2 nd	3 rd	7 th	6 th	Not	5 th	1 st	Not
3	Eligible						Eligible			Eligible
Requirement	Not	2 nd	4 th	1 st	Not	3 rd	Not	Not	Not	Not
4	Eligible				eligible		Eligible	Eligible	Eligible	Eligible
Requirement	Not	2 nd	Not	Not	1 st	Not	Not	Not	Not	Not
5	Eligible		Eligibl	Eligible		Eligible	Eligible	Eligible	Eligible	Eligible
			e							
Requirement	Not	3 rd	1 st	2 nd	6 th	5 th	Not	4 th	Not	Not
6	Eligible						Eligible		Eligible	Eligible
Requirement	5 th	2 nd	Not	1 st	3 rd	Not	Not	Not	Not	4 th
7			Eligibl			Eligible	Eligible	Eligible	Eligible	
			e							
Requirement	Not	5 th	4 th	1 st	7 th	6 th	2 nd	Not	3 rd	Not
8	Eligible							Eligible		Eligible
Requirement	Not	3 rd	4 th	1 st	2 nd	Not	Not	Not	Not	Not
9	Eligible					Eligible	Eligible	Eligible	Eligible	Eligible
Requirement	Not	5 th	7 th	4 th	1 st	6 th	Not	3 rd	Not	2 nd
10	Eligible						Eligible		Eligible	
Requirement	3 rd	2 nd	5 th	1 st	Not	Not	Not	4 th	Not	Not
11					Eligible	Eligible	Eligible		Eligible	Eligible
Requirement	1 st	6 th	Not	5 th	2 nd	Not	Not	4 th	Not	3 rd
12			Eligibl			Eligible	Eligible		Eligible	
			e							
Requirement	Not	4 th	Not	3 rd	1 st	Not	Not	Not	Not	2 nd
13	Eligible		Eligibl			Eligible	Eligible	Eligible	Eligible	
			e							

For Requirement 1 Google compute engine, GoGrid, Bit Refinery, Windows Azure, Saavisdirect and Joyent are ineligible. The reasons are explained as follows:

Google Compute Engine is ineligible since it offers does not offer services on monthly basis as required by requirement1. It also has no mention of processor speed and Requirement 1 demands 2.1 GHz processor speed. GoGrid is not eligible for the requirement 1 since it does not provide the required (25 GB) RAM. Bit refinery does not provide required RAM and Storage (25 GB and 800 GB). Windows Azure dies not meet criteria for required processor Speed (2.1 GHz). Savvisdirect provides an 8 GB RAM whereas Requirement1 demands 25 GB RAM. It also does not provide required storage (800 GB) and service credit (25%) Joyent has no mention of processor speed in its SLA.

Thus the only eligible clouds are Rackspace, HP, Nephoscale, and Opsource. These clouds fulfil all the requirements. On the basis of the proposed algorithm Rackspace gets 1,84,664.48 points, HP gets 27,092.658 points, Opsource gets 28,92,268.0 points, Nephoscale gets 54,918.07, The ranks are provided to these clouds on by arranging points in descending order, on

the basis of which Opsource gets highest points and is thus assigned 1^{st} rank.

In Requirement 2, Google Compute Engine, Bitrefinery, Rackspace, Opsource, Windows Azure and Joyent are ineligible. The reasons for the same are explained as follows: Opsource, Windows Azure and rackspace fulfil all the requirements, except the required processor speed (2.4 GHz). Google compute engine has not mentioned Processor speed in its SLA. And, it also does not provide services on monthly basis. Bitrefinery does not provide required 8 cores, 400 GB storage and 2.4 GHz processor speed. And Joyent has not mentioned processor speed in its SLA.

Hence, the eligible clouds are HP, GoGrid, Nephoscale, and Savvisdirect. HP gets 11976.511 points, Gogrid gets 9.9202395E15 points, Nephoscale gets 2072.1025 points, and Saavisdirect gets 1980.0953 points. Ranks are provided to these clouds on by arranging points in descending order, on the basis of which GoGrid gets highest points and is thus assigned 1st rank. Similarly the results for all other requirements are obtained. From the above ranks Go Grid can be considered as best cloud since it achieves highest number of 1st positions.

6. CONCLUSIONS

In this paper an approach is developed that provides the cloud consumer with the most suitable cloud according to his requirements and priorities. The same is done with the help of service level agreements offered by various clouds. The main objective of the work is to ease task of selection of a provider for a cloud customer. As the numbers of providers are increasing rapidly, this manual selection of clouds becomes quite challenging task for a user. Hence, this approach reduces the overhead of selection from the consumer by automatically selecting the best cloud for the user. The selected cloud is the one that meets all the user requirements in the best possible way. The difference between what user demands and what provider offers tells how much the cloud provider provides in addition to the minimum user requirement. For example, a provider that offers 75% more than what user demands is much better than one that offers just 10% or 5% more. Thus, all the clouds are ranked on basis of this difference. This approach also helps a provider to compare their services with other clouds and further improve and thus provide better services. We hope that this effort of ours would help those working in the area of cloud computing with their future works.

7. REFRENCES

- Lee Badger, Tim Grance, Robert Patt-Corner, Jeff Voas *Cloud Computing Synopsis and Recommendations* ,Recommendations of the National Institute of Standards and Technology, US Department Of Commerce, Special Publication 800-146
- [2] Preeti Gulia and Sumedha Sood, "Comparitive analysis of present day clouds using service level agreements", To be published in International Journal of Computer Application, 26th June 2013
- [3] Dustin Amrhein et all, Review and summary of cloud service level agreements, Cloud Computing Use Cases, White paper Version 4, IBM Corporation 2010. Retrieved from http://www.ibm.com/developerworks/cloud/library/clrev2sla.html
- [4] Preeti Gulia and Sumedha Sood, "Dynamic ranking and selection of cloud providers using service level agreements", To be published in International Journal of Advanced Research in Computer Science And Software Engineering, 26th June 2013
- [5] Tejas Chauhan, Sanjay Chaudhary, Vikas Kumar, and Minal Bhise. "Service Level Agreement parameter matching in Cloud Computing, Information and communication technologies (WICT)," 2011 World congress on IEEE, 2011
- [6] Saurabh Kumar Garg, Steve Versteeg and Rajkumar Buyya, "SMICloud: A Framework for Comparing and Ranking Cloud Services", 2011 Fourth IEEE International Conference on Utility and Cloud Computing
- "Google Compute Engine Service Level Agreement (SLA) - Google Compute Engine — Google Developers"https://developers.google.com/compute/docs /sla [online ; accessed June2013]

- [8] "Google Compute Engine Pricing Google Cloud Platform — Cloud Platform" https://cloud.google.com/pricing/compute-engine[online ; accessed June2013]
- "Cloud SLA | Rackspace Legal | Rackspace Hosting" http://www.rackspace.com/information/legal/cloud/sla [online; accessed June2013]
- [10] "Cloud Servers Pricing by Rackspace Cloud Computing &Hosting"http://www.rackspace.com/cloud/servers/ pricing/ [online ; accessed June2013]
- [11] "Service Level Agreements for All Products" https://www.hpcloud.com/SLA [online; accessed June2013]
- [12] "HP Cloud Pricing" https://www.hpcloud.com/pricing [online ; accessed June2013]
- [13] "Service Level Agreement (SLA) | GoGrid" http://www.gogrid.com/legal/service-level-agreement-sla http://www.gogrid.com/products/pricing, [online ; accessed June2013]
- [14] "Pricing|GoGrid" http://www.gogrid.com/products/pricing[online; accessed June2013]
- [15] "Cloud Hosting Service Level Agreement| OpSource|OpSource"http://www.opsource.net/Services/ Cloud-Hosting/Service-Level-Agreement[online; accessed June2013]
- [16] "Cloud Computing Cost Estimation & amp; Pricing|OpSource"http://www.opsource.net/Services/Clo ud-Hosting/Pricing [online ; accessed June2013]
- [17] "NephoScale Service Level Agreement NephoScale"http://www.nephoscale.com/service-levelagreement [online ; accessed June2013]
- [18] BareMetal Dedicated Servers | NephoScale "http://www.nephoscale.com/dedicated-servers" [online; accessed June2013]
- [19] "TOS" http://bitrefinery.com/tos [online; accessed June2013]
- [20] "Cloud Services Pricing" http://bitrefinery.com/cloudhosting/pricing[online; accessed June2013]
- [21] "Download Windows Azure Cloud Services, Virtual Machines, and Virtual Network SLA from Official Microsoft Download Center" http://www.microsoft.com/enus/download/details.aspx?i d=38427[online;accessed June2013]
- [22] "Windows Azure Pricing Calculator | Cloud Offers| Cloud Pricing" http://www.windowsazure.com/enus/pricing/calculator/ [online;accessed June2013]
- [23] "SLAs|savvisdirect"http://www.savvisdirect.com/slas [online; accessed June2013]
- [24] "Cloud Servers Pricing | savvisdirect" http://www.savvisdirect.com/cloud-servers/pricing [online; accessed June2013]
- [25] "Cloud Hosting Service Level Agreement Company-Joyent" http://joyent.com/company/policies/cloudhosting-service-level-agreement [online;accessed June2013]
- [26] "Joyent Cloud Compute Pricing Products Joyent" http://joyent.com/products/joyent-