

Cloud Utility Rapport Savvy Configuration Beneath Fuzzy Depending on user Inclination

¹Bhanushree N,
PG Student, M.Tech,
Jayam College of Engineering and
Technology,

²Saravana Kumar R,
Asst.Professor,
Jayam College of Engineering and Technology,

Abstract:- Nowadays the customers are not satisfied with the service provided by the single cloud service provider and gestalt of Cloud services is required. The tasks such as exploration, solidarity checking, selection, and unfoldment are included in Cloud service gestalt, which is a complex process and customers find it difficult to select the best one among the possible gestalts available. Diversity of customers with different expertise requiring their applications to be unfolded across difference geographical locations with distinct legal constraints. Selecting a consolidation of virtual appliances and infrastructure services that are compatible and satisfy a customer with vague predilection is the main difficulty lies. Therefore, a framework and algorithms which simplify Cloud service gestalt for unskilled customers is presented.

In this an ontology-based approach (PROMPT) to analyze Cloud service solidarity by applying reasoning on the expert knowledge is developed. And also the customers effort to in expressing their predilection should be minimized, therefore consolidation of evolutionary algorithms and fuzzy logic for gestalt optimization is applied. This helps customers to express their needs in linguistics terms which brings a great comfort to them compared to systems that force customers to assign exact weights for all predilections. In addition, customer experience is another important benchmark for Cloud service providers. It requires evaluating the service efficiently, delegating evaluation tasks to crowds, and calculating the accuracy of the assessments.

Keywords : Cloud computing, Cloud service gestalt, ontology-PROMPT, service level agreement, quality of service

1. INTRODUCTION

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 built of five essential characteristics, three service models, and four unfoldment models

Characteristics

The characteristics of cloud computing include on-demand self service, broad network access, resource pooling, rapid elasticity and measured service.

Service models

The cloud computing service models are Software as a Service, Platform as a Service and Infrastructure as a Service. In a *SaaS* model, a pre-made application, along with any required software, operating system, hardware, and network are provided. In *PaaS*, an operating system, hardware, and network are provided, and the customer installs or develops its own software and applications. The *IaaS* model provides just the hardware and network; the customer installs or develops its own operating systems, software and applications.

Unfoldment of cloud services

Cloud services are typically made available via a private cloud, community cloud, public cloud and hybrid cloud.

Generally speaking, services provided by a *public cloud* are offered over the Internet and are owned and operated by a cloud provider. Example e-mail services, or social networking sites. In a *private cloud*, the cloud infrastructure is operated solely for a specific organization, and is managed by the organization or a third party.

A *hybrid cloud* is a consolidation of different methods of resource pooling (for example, combining public and community clouds). In a *community cloud*, the service is shared by several organizations and made available only to those groups.

2. OVERVIEW

Cloud computing refers to the delivery of computing resources over the Internet. In this the cloud service provider plays the main role of providing services as requested by the customers. To deliver their solutions, application service providers can either utilize the platform-as-a-service (PaaS) offerings such as Google App Engine and OpenShift or develop their own hosting environments by leasing virtual machines from infrastructure-as-a-service (IaaS) providers like Amazon EC2 or GoGrid. However, most PaaS services have restrictions. Such restrictions encourage service providers to build their own platforms using IaaS service offerings.

One of the key challenges in building a platform for unfolding applications is to automatically build, design, and unfold the necessary application that consists of a number of different components. If considered the unfoldment requirements of a web application service provider, it will include security devices (e.g. firewall),

load balancers, web servers, application servers, database servers, and storage devices. Setting up such a complex consolidation of appliances is costly and error prone even in traditional hosting environments let alone in Clouds.

The selection of the best gestalt is a complex task and none of the providers provide any ranking system to choose the best instance type and software solution for the unfoldment. In addition, the best choices found for individual appliances cannot be simply put together as they may not be compatible with the hosting environment. For example, if an appliance format is OVF it cannot currently be unfolded on AmazonEC2. Moreover, there exist legal constraints imposed by countries such as the USA on importing and exporting of appliances from a provider to another. Dealing with all these complexities is costly and aggravating for unskilled customers and encourages them to seek professional help.

In order to tackle Cloud service gestalt challenges, an ontology-based approach to describe services and their QoS properties, which helps us to build a gestalt with a set of compatible services is described. Additionally this system helps non-expert customers with limited or no knowledge on legal and image format solidarity issues to unfold their services faultlessly.

3. EXISTING SYSTEM

Building a platform for unfolding is a complex task. In the existing system, as there are of too many service providers that the customers are not able to choose the right cloud service provider who will satisfy their requirements. And also the service provided should be compatible, (i.e.,) the incompatibilities that are caused by regulations and other factors that are not related to service functionalities should also be considered. The selection of the best gestalt is a complex task and none of the providers provide any ranking system to choose the best instance type and software solution for the unfoldment.

3.1 VIRTUAL APPLIANCES

Virtual appliances provide an elegant solution for this problem. They are built and designed with a necessary operating system and software packages to meet software requirements of a customer. However, customers have to first select their software solution and then the proper virtual machine configuration to host the software solution. A customer may also require more than one virtual appliance and machine. Furthermore, a customer may require more than one virtual appliance and machine, and a gestalt of them that can meet all the requirements of customers is required. However, the selection of the best gestalt is a complex task and none of the providers provide any ranking system to choose the best instance type and software solution for the unfoldment.

3.2 SERVICE SOLIDARITY

A Cloud service gestalt optimization technique that allows non-expert Cloud customers to set their predilections using high level if-then rules and get customer friendly recommendations on the gestalt

solution's prominence. The majority of end customers avoid systems that incur complexity in capturing their constraints, objectives and predilections. An example of such systems is the one which require customers to assign weights to their objectives.

3.3 SERVICE SUGGESTION

The non-expert customers with limited or no knowledge on legal and virtual appliance image format solidarity issues to unfold their services flawlessly. Thus suggestions based on the number of service providers based on the customers' requirements should be provided such that, people with limited knowledge will be able to select the best cloud service provider in an easy way. The information about the solidarity of the appliances should be suggested to improve the Quality Of Service.

4. PROPOSED SYSTEM

Our proposed architecture offers a unified solution that uniquely applies state of the art technologies for Service Suggestion from multiple numbers of existing available Services. The main goal of the architecture is to provide: ease of use for non-experts, more precise exploration and Suggestion. In order to realize and evaluate the proposed approach, a number of components and technologies are utilized.

A technique to optimize the service Suggestion based on customer predilections such as cost, and reliability. The approach exploits the benefits of the fuzzy logic to handle vague predilections of customers. A web service for cloud service information retrieval, instead of retrieve information from cloud providers. Results show that for the proposed case study, it can effectively help an unskilled customer to identify the Service are closest to their predilections

An ontology-based approach to analyze Cloud service solidarity by applying reasoning on the expert knowledge. In addition, to minimize effort of customers in expressing their predilection, a apply consolidation of evolutionary algorithms and fuzzy logic is applied for gestalt optimization.

This lets customers express their needs in linguistics terms which brings a great comfort to them compared to systems that force customers to assign exact weights for all predilection. In addition, customer experience is another important benchmark for Cloud service providers.

5. SYSTEM MODEL

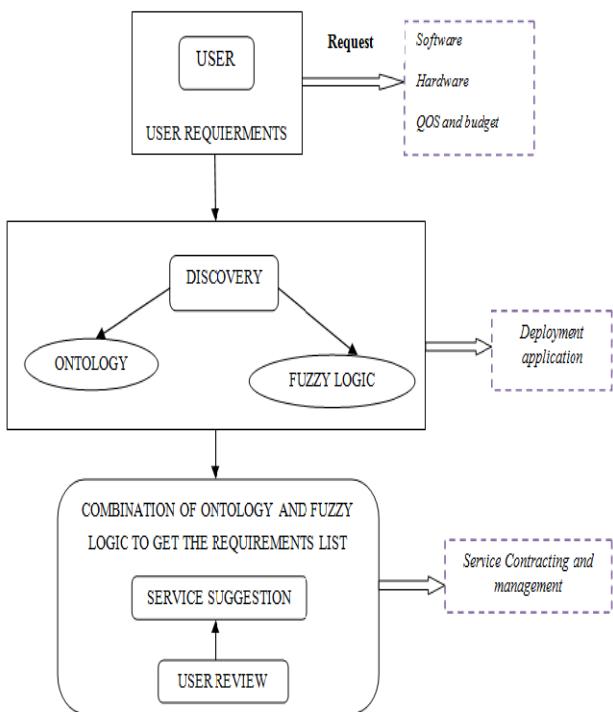


Fig. 5.1 System Model

6. SYSTEM DESIGN

6.1 ADMIN

6.1.1 Admin Login and View

This module allows admin to access Admin Home Page by using their unique customer name and password. This is been created by the administrator for the security purpose. When the customers get registered, list of registered customer with their time and date of registration can be viewed by the administrator for the verification purpose.

6.1.2 Account list – Unauthorized access

This module displays the list of registered Customer ID which was tried to access by the unauthorized customers by using invalid password. This helps to maintain the privacy of the registered customers.

6.2 CUSTOMER LOGIN

When the customer gets registered a customer name and password will be provided to authenticate the customer. If a particular customer name is accessed by invalid password more than one time, this module automatically send alert mail to that particular customer's Mail ID. The mail objective of this module is to provide authorized accessed to the customers profile.

6.2.1 Objective Evaluation

Poorly specified customer requirements are one of the most significant factor behind IT project failure. A successful product or system requires a proper

understanding of both customer and organizational requirements. In this first the customer requirements are evaluated based on there interest and to provide the best service satisfying their requirements. Thus the customer requirements like Maximum Cost and Minimum Cost for retrieve **Infrastructure and Platform** Service information from Web Service.

6.2.2 Pareto Front List

Use of Pareto principle in prioritizing or ranking a range of items which have different levels of significance. The **Pareto front** or **Pareto frontier** is a framework for *partially* evaluating a set of “actions” with multi-dimensional outputs assuming a very weak “desirability” partial ordering which only applies only when one processes is better (or at least as good) for *all* the outputs. It is useful for reducing a set of candidates prior to further analysis.

6.2.3 Service Gestalt Suggestion (IAAS and PAAS)

The service gestalt is of combining multiple existing services. These composite services can be in turn recursively built with other services into higher level solutions, and so on. Such recursive gestalt of services is one of the most important features of SOA, allowing to rapidly build new solutions based on the existing services. As the amount of individual services (and their gestalts) grows, the easier it becomes to implement new enterprise solutions.

The service gestalt suggestion is to provide the customer's the best gestalt of the service based on their requirements. In this the service provided by the service provider will be collected and stored. Thus when the customer's request for the service both the requirements and the service provider details will be compared and the service which can be provided will be listed down to the customer's.

6.2.4 Review The Services

In addition to a critical evaluation, the review's author may assign the work a rating to indicate its relative merit. The reviews are used on shopping sites to give customers an opportunity to rate and comment on products they have purchased, right on the product page. Other consumers can read these when making a purchase decision.

6.2.5 Log Out

Once all the requirements have been negotiated between the customer and the service provider in a detailed manner about the service to be provided and the customer gets satisfied the registered account should be logged out to prevent from unauthorized access.

7. ALGORITHM

7.1 Ontology

Ontology is an explicit specification of a conceptualization. Based on this definition, ontologies are used in the information retrieval field to represent shared and more or less formal domain. By combining it with the

techniques of natural language processing, the documents in the database will be summarized and linked to the ontology concepts. If this step has been properly done, the search would be easier in the future. At the queries reformulation level in order to improve the initial customer queries. The idea is to use ontology to add the semantic dimension to the evaluation process. This can be done by extracting the query terms and their semantic projection using the Word Net ontology on the set of returned documents. The result of this projection is used to extract concepts related to each term, thus building a semantic vector which will be the base of the results classification.

PROMPT is an algorithm that provides a semi-automatic approach to ontology merging and alignment. PROMPT performs some tasks automatically and guides the user in performing other tasks for which his intervention is required. PROMPT also determines possible inconsistencies in the state of the ontology, which result from the user's actions, and suggests ways to remedy these inconsistencies. PROMPT is based on an extremely general knowledge model and therefore can be applied across various platforms.

The PROMPT is an ontology-merging and ontology-alignment algorithm. PROMPT takes two ontologies as input and guides the user in the creation of one merged ontology as output. First PROMPT creates an initial list of matches based on class names. Then the following cycle happens: (1) the user triggers an operation by either selecting one of PROMPT's suggestions from the list or by using an ontology-editing environment to specify the desired operation directly; and (2) PROMPT performs the operation, *automatically* executes additional changes based on the type of the operation, generates a list of *suggestions* for the user based on the structure of the ontology around the arguments to the last operation, and determines *conflicts* that the last operation introduced in the ontology and finds possible solutions for those conflicts.

7.2 Fuzzy Logic

A mathematical logic that attempts to solve problems by assigning values to an imprecise spectrum of data in order to arrive at the most accurate conclusion possible. Fuzzy logic is designed to solve problems in the same way that humans do, by considering all available information and making the best possible decision given the input. It is an approach to computing based on "degrees of truth" rather than the usual "true or false" (1 or 0) Boolean logic on which the modern computer is based. The vague semantic descriptions located in the form of fuzzy rules and create a criterion to measurement of composite services, and then determine and measure the importance of each rule according to customer's clear point of views. In addition this approach exploits the benefits of evolutionary algorithms such as Optimal Multi Objective Particle Swarm Optimization algorithm (OMOPSO), No dominated Sorting Genetic Algorithm - II (NSGA-II), and Strength Pareto Evolutionary Algorithm -

II (SPEA-II) for optimization and fuzzy logic to handle vague predilections of customers.

7.3 Attribute Based Encryption Algorithm

Attribute-Based Encryption (ABE) is a vision of public key encryption that allows customers to encrypt and decrypt messages based on customer attributes. This functionality comes at a cost. In a typical implementation, the size of the cipher text is proportional to the number of attributes associated with it and the decryption time is proportional to the number of attributes used during decryption. Specifically, many practical ABE implementations require one pairing operation per attribute used during decryption. This work focuses on designing ABE schemes with fast decryption algorithms. In this setting, The first key-policy ABE system where cipher texts can be decrypted with a constant number of pairings.

8. CONCLUSION

For the Cloud service gestalt challenges, an ontology based approach is presented to describe services and their QoS properties, which helped us to build a gestalt with a set of compatible services. This system helps non-expert customers with limited or no knowledge on legal and image format solidarity issues to unfold their services faultlessly. A technique to optimize the service gestalt based on customer predilections such as unfoldment time, cost, and reliability is been proposed. The approach exploits the benefits of evolutionary algorithms such as OMOPSO, NSGA-II, and SPEA-II for optimization and fuzzy logic to handle vague predilections of customers.

Results show that for the proposed case study, effectively help an unskilled customer to identify the appliance gestalts which are closest to their predilections. Specifically, defining criteria which are able to model reliability, and trust of a Cloud service are increasingly attractive to customers. Methods to evaluate reliability and trust of providers from customer feedbacks and monitoring services together are been implemented to improve the QoS. This consists of collecting required raw data from trusted sources and statistically analyzing and aggregating them. In addition, customer experience is another important benchmark for Cloud service providers.

REFERENCES

- [1] Aboulnaga A, K. Salem, A. Soror, U. Minhas, P. Kokosiellis, and S. Kamath, "Unfolding Database Appliances in the Cloud," IEEE Data Eng. Bull., vol. 32, no. 1, pp. 13-20, 2009
- [2] Alrifai M, T. Risse, P. Dolog, and W. Nejdl, "A Scalable Approach for QoS-Based Web Service Selection," Service-Oriented Computing— ICSOC 2008 Workshops, Springer, 2009
- [3] Bartalos P and M. Bielikov_a, "Automatic Dynamic Web Service Composition: A Survey and Problem Formalization," Computing and Informatics, vol. 30, no. 4, pp. 793-827, 2012

- [4] Bradshaw R, N. Desai, T. Freeman, and K. Keahey, "A Scalable Approach to Unfolding and Managing Appliances," Proc. Tera-Grid Conference, 2007.
- [5] Coello C A C, G. B. Lamont, and D. A. Van Veldhuizen, Evolutionary algorithms for solving multi-objective problems. Springer, 2007, vol. 5.
- [6] Dastjerdi A, S. Tabatabaei, and R. Buyya, "An effective architecture for automated appliance management system applying ontology-based cloud exploration," in Proceedings of the 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing (CCGrid), 2010.
- [7] Dastjerdi A and R. Buyya, "An autonomous reliability-aware negotiation strategy for cloud computing environments," in Cluster, Cloud and Grid Computing (CCGrid), 2012 12th IEEE/ACM International Symposium on. IEEE, 2012.
- [8] De Bruijn J, H. Lausen, A. Polleres, and D. Fensel, "The web service modeling language wsml: An overview," in Proceedings of the 3rd European conference on The Semantic Web: research and applications (ESWC06), pp.590–604, 2006.
- [9] Deb K, Multi-objective optimization using evolutionary algorithms. Wiley, 2001.
- [10] Natalya Fridman Noy and Mark A. Musen, "Algorithm and Tool for Automated Ontology Merging and Alignment" Stanford Medical Informatics, Stanford University, Stanford, CA 94305-5479
- [11] Jsang A and R. Ismail, "The beta reputation system," in Proceedings of the 15th Bled Electronic Commerce Conference, 2002.
- [12] Kiran M, M. Jiang, D. Armstrong, and K. Djemame, "Towards a Service Lifecycle Based Methodology for Risk Assessment in Cloud Computing," Proc. Ninth Int'l Conf. Dependable, Autonomic and Secure Computing, 2011.
- [13] Kopeck J y, D. Roman, T. Vitvar, M. Moran, and A. Mocan, "Wsmo grounding. wsmo working draft v0. 1", 2007.
- [14] Lecue F and N. Mehandjiev, "Towards Scalability of Quality Driven Semantic Web Service Composition," Proc. IEEE Int'l Conf. Web Services, pp. 469-476, 2009.