A Reference Architecture for Interoperability in a Multi Cloud Environment

Submitted By:

Sadia Siddiqa

368-FBAS/MSSE/F12

Supervisor: Dr. Sheheryar Malik

Head of Graduate Programs & Research Assistant Dean

Co-Supervisor: Zakia Jalil

Assistant Prof. IIUI

Department of Computer Science & Software Engineering

Faculty of Basic and Applied Sciences

International Islamic University Islamabad

2016

1. Cloud computing [illegible]
2. Multi [illegible]

Department of Computer Sciences and Software Engineering

International Islamic University Islamabad

Final Approval

Dated: 7-11-2016

It is certified that we have read the project report submitted by Miss Sadia Siddiqa (368-FBAS/MSSE/F12), and it is our judgment that this project is of sufficient standard to warrant its acceptance by the International Islamic University, Islamabad for the MS Degree in Software Engineering

Committee

External Examiner

Dr Huma Hayat Khan
Assistant Professor
Department of Software Engineering
NUML Islamabad

Internal Examiner

Salma Imtiaz
Assistant Professor
Department of Computer Science
And Software Engineering
IIUI- Islamabad

Supervisor

Dr Shehervar Malik, PhD
Associate Professor
Department of Computer Science & Engineering
Inha University, Incheon Korea
IUT Tashkent Campus

Co-Supervisor

Zakia Jalil
Assistant Professor
Department of Computer Science
And Software Engineering
IIUI- Islamabad

Table of Contents

ABSTRACT

Cloud technology is emerging computer paradigm which allows users to access computing resources by using internet As soon as cloud adoption is more evolving, cloud users has more demand of availing services from cloud service providers One cloud service provider can provide limit services which does not fulfil customers full requirements so customers intended to have service from more than one cloud service provider at same time Access to multi-cloud environment fulfills these requirements if those interoperate with each other As per our literature review Multi-cloud have an inherent problem of interoperability, as different cloud service providers use different standards for their services Some interoperability libraries have been produced to handle this situation Most of these libraries are in development process Currently different cloud service providers are platform dependent environment which locks in the customer to interoperate with other cloud service providers and their structure and services stop the portability process The European commission and ENISA has highlighted as a high risk of vendor lock-in problem because it does not meet the multi-cloud environment usage requirements for a single customer to conduct specific tasks in specific infrastructure

In order to enable interoperability between different cloud service providers and to eliminate the vendor-lock in problem, interoperability reference architecture shall solve those issues We have proposed the cloud interoperability reference architecture which solves the vendor lock-in issue

Our research methodology was literature review where we have studied number of research papers, conference and journals After that we have proposed a reference architecture and conducted an expert review from cloud industry researchers and professionals Based upon that we have conducted the analysis and recorded our analysis in later chapter The identification of cloud interoperability components is a very challenges task In the process of identification from the literature survey, we had identified number of components which are being used at present level and also in also future clouds

Keywords: *Cloud Federation, Cloud Interoperability, Integration, Portability*

Acknowledgement

In the Name of Allah the most merciful and kind who made whole universe I am really blessed by Allah who made me able to accomplish this thesis.

I am obliged to my Supervisor, Dr Sheheryar Malik who guided me and highly encouraged me at each stage of my thesis to complete my thesis work His great ideas, suggestions and recommendation during my thesis were quite useful in order to accomplish my work His knowledge, expertise and continuous help was quite helping

I am also obliged to Ms Zakia Jalil for her valuable guidance, valuable discussions and encouragement during my thesis work. At last, I am thankful to my family, my parents who encouraged me and helped me financially to complete my work Indeed without their help and prayers, it was not possible to accomplish my work

Dedication

This thesis is dedicated to my beloved parents, respected teachers and all those who prayed for my success

Sadia Siddiqa

368-FBAS/MSSE/F12

Dissertation

A dissertation Submitted To

Department of Computer Science,

Faculty of Basic and Applied Sciences,

International Islamic University, Islamabad

As a Partial Fulfillment of the Requirement for the Award of the

Degree of

MS Software Engineering (MSSE)

Chapter 1
Introduction

1. Introduction

This chapter gives an introduction to this research work, it provides overview of cloud computing, cloud federation and cloud interoperability and introduces the problem, motivation and methodology for this study

1.1 Background

The aim of cloud computing is to provide its resources as a services in accordance with pay-per-use paradigm The resources provided by cloud are different with respect to its type and level of abstraction [1] Cloud can provide its resources at the level of infrastructure which includes Infrastructure as a Service (IaaS), e g virtual storage, at the platform level Platform as a Service (PaaS) e g programming libraries, application templates and components and at the software level SaaS e g full fledge application like Google documents [2]

In cloud computing environment, resources are pooled in a central location under the supervision of single cloud service provider and consumers access these resources through web browser [3] Cloud computing services models consist of three models i e Infrastructure as a Service (IaaS) Platform as a Service (PaaS) and Software as a Service (SaaS) These models are different for providing capabilities and how cloud customers can access these services and further it can be used IaaS provides basic computing capabilities to customers In PaaS environment, users can deploy their applications and control their hosting environment to some extent SaaS enables to run some applications on cloud infrastructure for its clients [4]

There are several cloud service providers from IT industry such as Amazon Web Services (AWS) [5], Rackspace [6], and Flexiant's FlexiScale [7], are based on their own interfaces technology and semantics [8] Thus, leads to isolated environments This isolation prevents the enterprises to fulfill all their business requirements under the supervision of single cloud service provider There is a need of cloud federation which will enable the clouds to interoperate with each other and federation is not possible without interoperability [9]

This research focuses on "Architecture of Interoperability Manager" The main contribution of this research is to develop a framework which comprises of guidelines related to the internal structure of interoperability manager All the existing work based on the cloud federation architecture and interoperability is one of its modules The internal components and elements for interoperability manager has been designed which aims to provide communication between multiple cloud providers using some interaction mechanism [10]

1.2 Cloud: Overview

Although numerous definitions of cloud computing have been proposed but one delivered by the NIST (National Institute of Standards and Technology) is very proper definition which includes all the elements used in cloud community [11]

Cloud computing is a model for enabling convenient, ondemand 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 [11]

1.3 Essential Elements

These definitions will describes the architecture, security and deployment strategies of cloud Specifically written about five important elements of cloud computing

1.3.1 On-demand self-service

A consumer can avail instant services on need basis such as CPU time, software, network use or storage automatically without physical interaction of human [11]

1 3.2 Broad network access

These resources are being provided over the network and can be used in heterogeneous environments (such as mobile devices, PDAs, laptops, mobile phones) by different client applications [11]

1.3.3 Resource pooling

A cloud service provider's resources are "pooled" in order to deliver services to multiple consumers It is being done either use of *multi-tenancy* or *virtualization* model in which multiple resources in terms of physical and virtual are assigned dynamically and as per consumer demand [12] There are two factors involved in motivation of setting up the pool-based computing.

economies of scale and *specialization* Consumer doesn't know where their data is being processed in cloud due to invisibility of computing resources from cloud service providers [11]

1.3.4 Rapid elasticity

Computing resources became immediate rather than persistent

There are not pre committed contract with consumers that they can use them to increase resources whenever they need and they can release when not required Moreover, provisioning of resources are infinite, consumption of resources are increasing rapidly in order to meet the consumer's requirements at any time [11]

1.3.5 Measured services

Although computing resources has been pooled and more than one consumer is using them at a same time (i e multi-tenancy), cloud infrastructure can measure the proper usage of resources between the consumers by using their metering system [11]

1.4 Service Model

Except those five important characteristics, cloud computing community has used below three service models extensively in order to categorize the cloud services

1.4.1 Infrastructure as a service (IaaS)

The services provided to consumers to deliver processing power, storage, networks and other required resources where a consumer can deploy their required software such as operating systems and applications Cloud (EC2) web service provided resizable compute size For example amazon simple storage service (S3) web service provides a simple web service interface in order to save and populate data when required over the web [13].

1.4.2 Platform as a service (PaaS)

The services provided to consumers to deploy applications been created by using programming languages, Integrated Development Environment (IDEs), Application Programming Interface (APIs), libraries and services onto the cloud infrastructure which are supported by the provider For example, google provides an engine named as Google APP It allows developers to run web applications on Google's infrastructure It is easy to build and maintain and easy to grow its data storage when required [14]

1.4.3 Software as a service (SaaS)

The service provided to consumers is to use applications provided by providers which are running on cloud platform [15] For example there is an online customer relationship management provided by salesforece com and Google Maps Inter-cloud operations is widely used in the industry since customer's requested services might involve multiple clouds

1.5 Deployment Model

Currently there are four cloud deployment models are introduced in cloud community

1.5.1 Private cloud

The infrastructure of cloud which is operated within a single organization managed by organization itself of managed by 3rd party irrespective of its deployment within its premises or outside premise There are several reasons to deploy private cloud within an organization for example security concerns, including data privacy and maximization and optimization of existing in-house resources [16]

1.5.2 Community cloud

Numerous companies and organizations deploy cloud infrastructure and they share the same infra, deployed policies, requirements and other requirements The cloud computing environment can be deployed either by third vendor or it can be deployed within an organization [17]

1.5.3 Public cloud

Public cloud computing is a foremost term used in cloud environment where a cloud computing company which is hosting cloud has full ownership on infrastructure, policies, profit, costing and billing model For example google AppEngine and Amazon Elastic Compute 2 (EC2) [1]

1.5.4 Hybrid cloud

The cloud environment which is combination of two or more clouds (includes public, private and hybrid) those are sole entities but followed by standardized technologies Organizations use hybrid cloud in order to increase their efficiency and optimize the resourcing to increase their core capabilities Organizations controls their core business functions and activities on premises through private cloud [18]

1.6 Cloud Federation

The concept of cloud federation is used to reduce the cost of subcontracting with to other regions By applying this, security requirements can be fulfilled by using some techniques such as fragmentation Its benefit is to use services in a single pool by support of interchangeability basic features resource migration, aggregation of complementary resource and resource redundancy Cloud federation includes provision engine, distribution manager, and deployment manager, configuration manager, data distribution manager, transformation, resource manager and monitoring components [3]

1.7 Cloud Interoperability

In computer world, this property has the concrete meaning of exchanging information and use of the information that has been exchanged between two or more systems or components It is a property of a product or system, whose interfaces are completely understood, allowing it to work with other products or systems The best way to elaborate the concept of cloud computing interoperability is its mottos "avoid vendor lock-in" "develop your application one deploy anywhere", "enable hybrid clouds", or even "one API to rule them all" [19]

1.7.1 Interoperability Motivation

Interoperability is lacking in cloud federation which improves the communication between customers and cloud service providers In specifically, a customer will be able to choose among different cloud service providers which are offering services with different characteristics without having their data or application at risk [20] Moreover, interoperability cloud market will help small and medium enterprises to stand and strengthen theirselve in cloud market They can interoperate and built new cloud models without having interoperability problems [21]

Although different provided interoperability standards can have different solutions to interoperability problems which are not interoperable to each other Therefore industry professionals, researchers should agree upon common principles which can provide solution to interoperability and all interoperability solutions can adhere to it [22]

1.7.2 Cloud Interoperability Challenges

This section enlighten some of the challenges encounter in the way of interoperability

Portability and Mobility

An interesting question " can I deploy existing cloud assets (VM images, software, apps) on other service providers environment without having changes to artifacts Urquhart has divided application, software, VM images, and servers in to two categories portability and mobility [23] According to Urquhart definition of portability "the ability to move an image in down stat from source to destination and then boot it from new location" According to Urquhart definition of mobility is "ability to move a host's workload from one to another without losing client's connection or in-flight state" Portability or mobility give major indication of cloud interoperability between different clouds Mobility is one of the factor of interoperability of cloud It will require the advancement in VM technologies and APIs for supporting migration of live virtual machines, Internet Protocol (IP) and other services from one cloud to another [24]

Cloud-Service Integration

In order to accomplish and maintain controls over the mission-critical operations, SMEs might need to integrate both on-premises and SaaS applications in order to complete the business requirements The current practice need sufficient coding which can integrate software by using APIs as well as frequent updates [25]

Security, Privacy and Trust

Cloud adopters also require security from cloud service providers such as access control of users to PII [26] This requires the industry standard best solutions such as multi-layer security, role based access control, logging and monitoring To reduce the cost with security requirements potentials, cost can be reduced through consolidation and optimization of resources, cloud computing has introduced set of security and privacy issues in order to make cloud computing successful [27] In addition to that, cloud computing service providers have to provide sufficient level of security in cloud environment to customers [28] Security issues are strongly associated with the cloud administrations such as session management, user management, access control monitoring, security policies and authentication

Management, Monitoring, and Audit

Cloud users need the assurance of compliance to security and privacy policies consistency and SLAs should be met when cloud services are being migrated from one cloud to another Automated tools should be monitoring the compliance and reporting to violation of security policies set by cloud service providers [25]

1.8 Research motivation

A single cloud service provider can provide limited services which does not fulfil customer s full requirements so customers intended to have services from more than one cloud service provider at same time Access to multi-cloud environment fulfills these requirements if those interoperate with each other As per our literature review Multi-cloud have an inherent problem of interoperability, as different cloud service providers use different standards for their services Even their service architecture may differ for the same sort of service Some interoperability libraries have been produced to handle this situation But these libraries discuss the interoperability at a quite abstract level Moreover, most of these libraries are in development process [34] [35] [36] Currently different cloud service providers are platform dependent environment which locks in the customer to interoperate with other cloud service providers and their structure and services stop the portability process [37] This dominating battle resist the small and medium organizations (SMEs) to adopt the cloud services because of vendor lock-in issues [12] The European commission and ENISA has highlighted as a high risk of vendor lock-in problem [39] because it does not meet the multi-cloud environment usage requirements for a single customer to conduct specific tasks in specific infrastructure

1.9 Research Question

On the basis of above discussion following questions raise for research as

RQ1 what are the components related to interoperability?

RQ2 how interoperability issues can be resolved when two service providers federate with each other?

1.10 Research Objectives

As the interoperability between multiple cloud platforms provides the consumer a single place where they are enabled to achieve all the resources from different cloud service provider The objective of this research is to develop an explanatory reference architecture for multi cloud environment in order to resolve the issues Those issues (e g vendor lock-in) arises when two cloud service providers are being federated with each other This reference architecture will be developed through study of different federation architectures

1.11 Research methodology

The research comprises of literature survey From the literature we extract the interoperability components on the basis of most referred and relevant components Data selection is based on the keywords and synonym and the quality of conference and journals After getting data we design a questionnaire to get the answer of our research questions Expert review is conducted to evaluate the proposed reference architecture For this purpose we choose the experts from cloud background and send them the questionnaire We have analyzed the results based upon the experts review so that it can be observed that how much effective is the proposed architecture It is not possible to include all the literature architecture in our research but most of them are discussed

1.12 Thesis Outline

This research draft is comprises of six chapters They are organized as follows

- Chapter one is an introductory chapter
- Second chapter presents the literature review related to this thesis all the scenarios provided in literature
- The literature regarding the methodology of the work will also be discussed in chapter three
- Fourth chapter introduces the proposed architecture that is Reference Architecture for Interoperability in a Multi Cloud Environment
- Chapter five discusses the theoretical evaluation of the proposed reference architecture on the basis of comparison between the literature and the thesis work
- Last chapter is about the concluding part of the thesis which defines the summary and the

contribution of the research It summarizes the work its applications and possible direction for future work, in the next section references were written

Chapter 2

Literature Review

2. Literature Review

This chapter briefly gives an introduction of problem domain and existing state of art solutions dealing with the federation of cloud which discusses interoperability as a minor component

2.1 Cloud Federation in Contrail

The federation acts as a bridge between users and cloud providers The *federation support* offers to users, in a uniform fashion, resources belonging to different cloud providers A Contrail federation can exploit two kind providers, those based on the Contrail cloud infrastructure and the ones based on other public and commercial infrastructures As shown in Figure 1 the federation architecture *is* composed of three layers Every layer is in turn composed by modules where each module addresses a well-defined commitment [31]

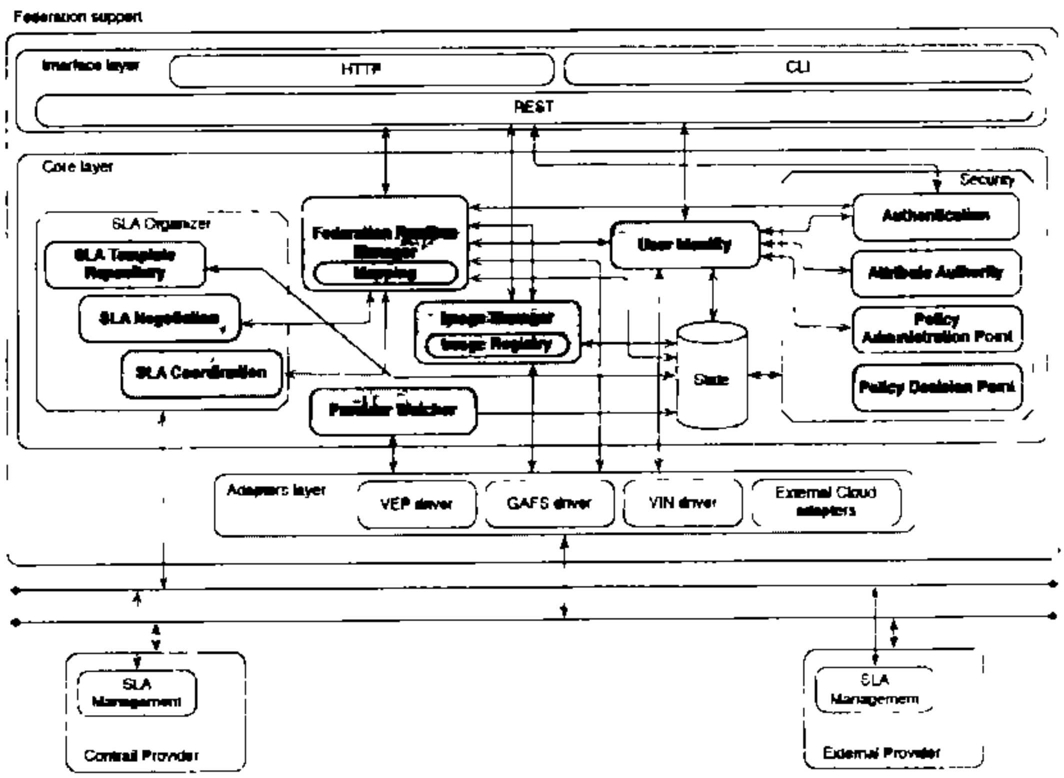

Figure 1 Contrail Federation-support Architecture [31]

The architecture of the European project Contrail [31] is fabricated around a central entity that performs its activities as a single entry point for federation of cloud providers Its responsibilities include continuous observation of state of cloud service providers and facilitating federation-wise SLA It also provides the service of SSO so every user should authenticate only once and they can work with whole federation There is a vital architecturally designed component known as Federation Runtime Manager (FRM) which is responsible to make comparison between cloud resources and user's requirements Cost and performance related problem solving techniques are designed by the FRM and it also provide all the information about cloud service providers The responsibility of FRM module is to provide services according to SLA by minimizing costs

The Adapter Layer contained all the adapters related to each cloud service provider The function of each adapter is to build a connection between cloud and federation manager There are two type of adapters listed below

- Internal adapters – these adapters are based on contrail software that is why they are called contrail clouds In contrail clouds, all the resources are organised by the federation manager [31]
- External adapters – this adapter is for those clouds which are not responsible for running contrail software

Federation is uncertain in case of using external adapters But the contrail architecture shows the federation pattern is Multi-Cloud using external adapters Internal adapter federation is called centralised federation In fact, two types of architectures supported by the contrail environment

2.2 The Federated Cloud Management Architecture

Figure 2 shows the Federated Cloud Management (FCM) architecture and its connections to the corresponding components that together represent an interoperable solution for establishing a federated cloud environment

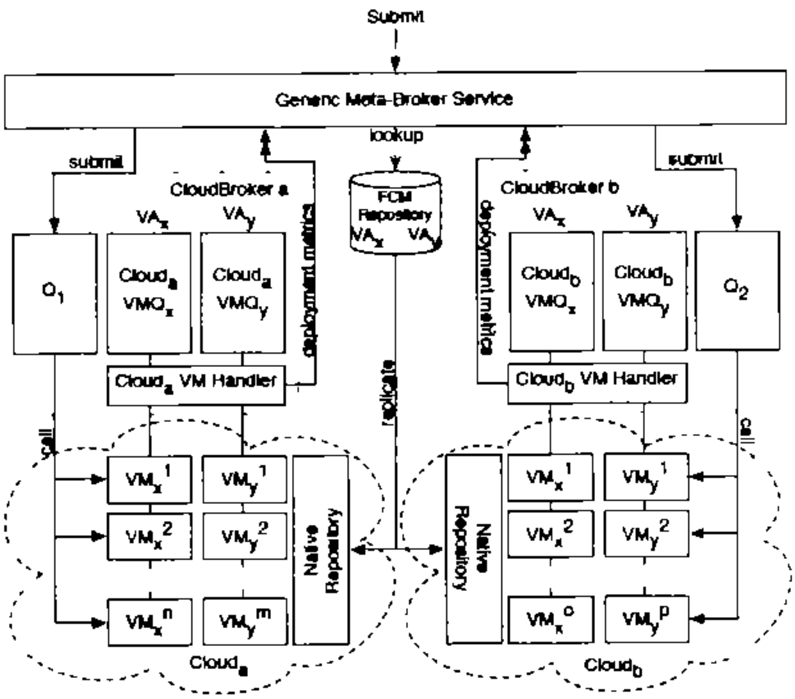

Figure 2 The Federated Cloud Management Architecture [30]

The architecture Federated Cloud Management (FCM) trusts on a general repository which is known as FCM repository for storing virtual appliances for federated services [30] This is mirrored on native repositories of IaaS providers The client's deals with Generic Meta – Broker Service (GMBS) only so that it can describe the services requested and further provisioning as well as scheduling is transparent if they are concerned further [30]

In architecture, there is a communicator broker in every IaaS provider known as CloudBroker which manages the allocation and VMs and de-allocation as well It sends applications calls which are incoming to concerned virtual machines

GMBS can talk to CloudBroker components which are part of federated cloud because it has access to repository of FCM When request is submitted to GMBS it then execute the matchmaking between the concerned CloudBroker and request received

This matchmaking is grounded by the information provided by repository of FCM and runtime metrics of CloudBrokers Every CloudBrokers maintains a database queue for all application calls which is received and maintains a database of priority queue separately for each virtual appliance VM queues exposes those resources which can be serve those calls related to virtual appliance

The priority of VM queue is set on the bases of all requests available in queue, the total VMs running and the execution times in log On basis of VM queues, CloudBroker will allocate or de-allocate the virtual machine The CloudBroker will also managing the above mentioned calls which it is receiving It forwards those calls to concerned VM which are created by CloudBroker In FCM, application brokering is managed transparently at client side Nothing is said about the user-level control used resources locations as per definition of architecture [30] In addition to that, those location specific requirements can be a part of the SLA between GMBS and clients

2.3 Reference Architecture for Semantically Interoperable Clouds

The Reference Architecture as described in Figure 3 will introduce a scalable, reusable, modular extendable and transferable approach for facilitating the design, deployment and execution of resource intensive SOA services on top of a semantically interoperable Cloud

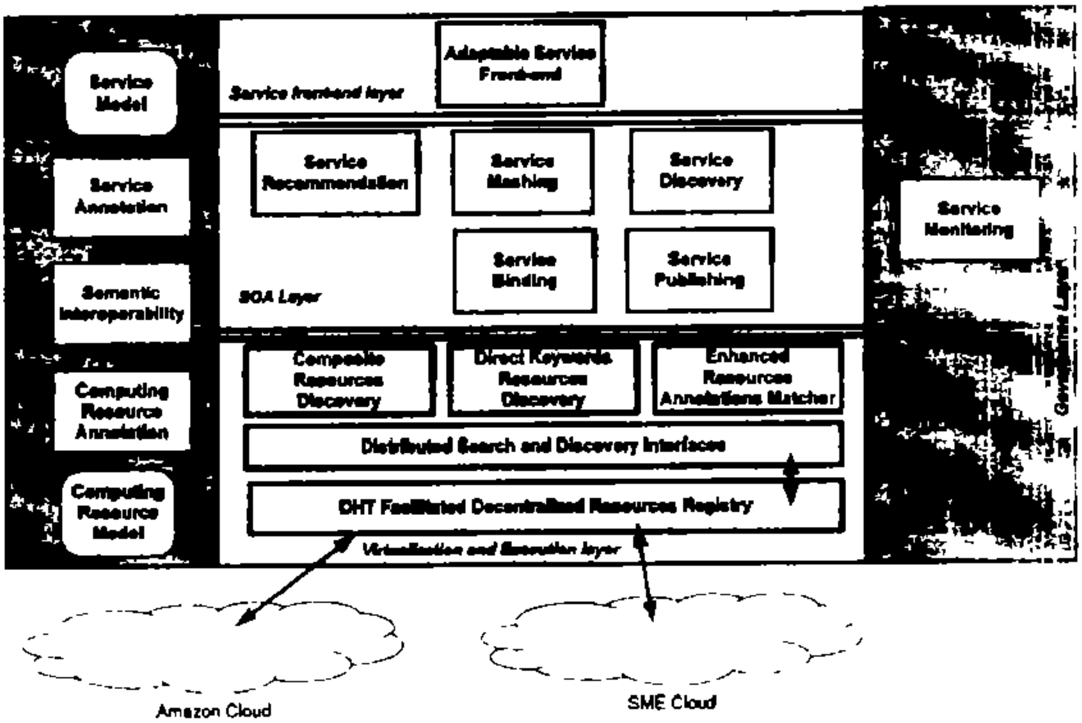

Figure 3 The Reference Architecture for semantically Interoperable Clouds (RASIC) [12]

Reference Architecture for Semantically Interoperable Clouds (RASIC) [12] is an extendable and modular approach highlighted by Nikolaos Loutas et al It is the mediator among different clouds infrastructure For data transformation, standardized common cloud API is designed by the authors RASIC has three horizontal layers (1) Service front-end layer, (2) SOA layer, (3) virtualization and execution layer And two vertical layers (1) Semantic layer (2) Governance layer The switching cost from one cloud provider to other and the vendor lock-in problem can be minimized by using RASIC and common cloud API approach

2.4 Intercloud Architecture Framework

The Intercloud Architecture labeled in figure 4 address the interoperability and integration issues in the current and emerging heterogeneous multi-domain and multi-provider clouds that could host modern and future critical enterprise and e-Science infrastructures and applications including integration and interoperability with legacy campus/enterprise infrastructure

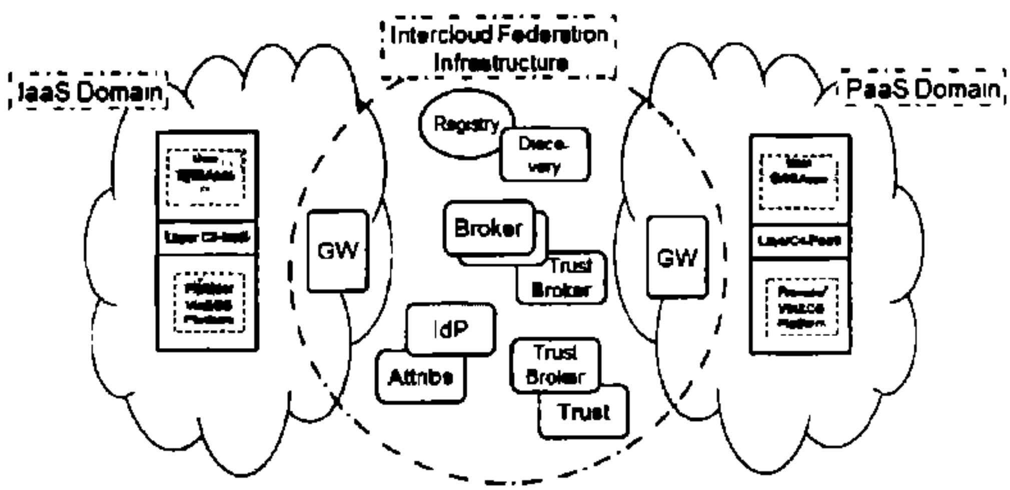

Figure 4 Intercloud Federation Framework [40]

In [40], author proposed Intercloud Architecture Framework that discourse the problems regarding interoperability and integration in the mixed cloud environment This architecture is for the cloud based architecture that allows the provisioning of network, computing and storage resources The ICAF comprises of four corresponding components

2.4.1 Multilayer Cloud Service Model (CSM)

Multilayer Cloud Services Model (CSM) for vertical cloud services interaction, integration and compatibility that defines both relations between cloud service models (such as IaaS, PaaS SaaS) and other required functional layers and components of the general cloud based services infrastructure

2.4.2 Intercloud Control and Management Plane (ICCMP)

Intercloud Control and Management Plane (ICCMP) for Intercloud applications/infrastructure control and management, including inter-applications signalling, synchronization and session management, configuration, monitoring, run time infrastructure optimization including VM migration, resources scaling, and jobs/objects routing

2.4.3 Intercloud Federation Framework (ICFF)

Intercloud Federation Framework (ICFF) to allow independent clouds and related infrastructure components federation of independently managed cloud based infrastructure components belonging to different cloud providers and/or administrative domains, this should support federation at the level of services, business applications, semantics, and namespaces assuming necessary gateway or federation services

2.4.4 Intercloud Operation Framework (ICOF)

Intercloud Operation Framework (ICOF) that contains abilities to support more than one provider infrastructure operation that includes workflows of business, management of service level agreement and accounting Intercloud Operation Framework (ICOF) explains the mandatory actors roles and inter relations in terms of ownership, operations and management Intercloud Operation Framework (ICOF) needs interactions with both ICMP and ICFF It requires support with ICMP and ICFF as well

2.5 Service-oriented Cloud Computing Architecture

Service Oriented Cloud Computing Architecture is a layered architecture shown in Figure 5

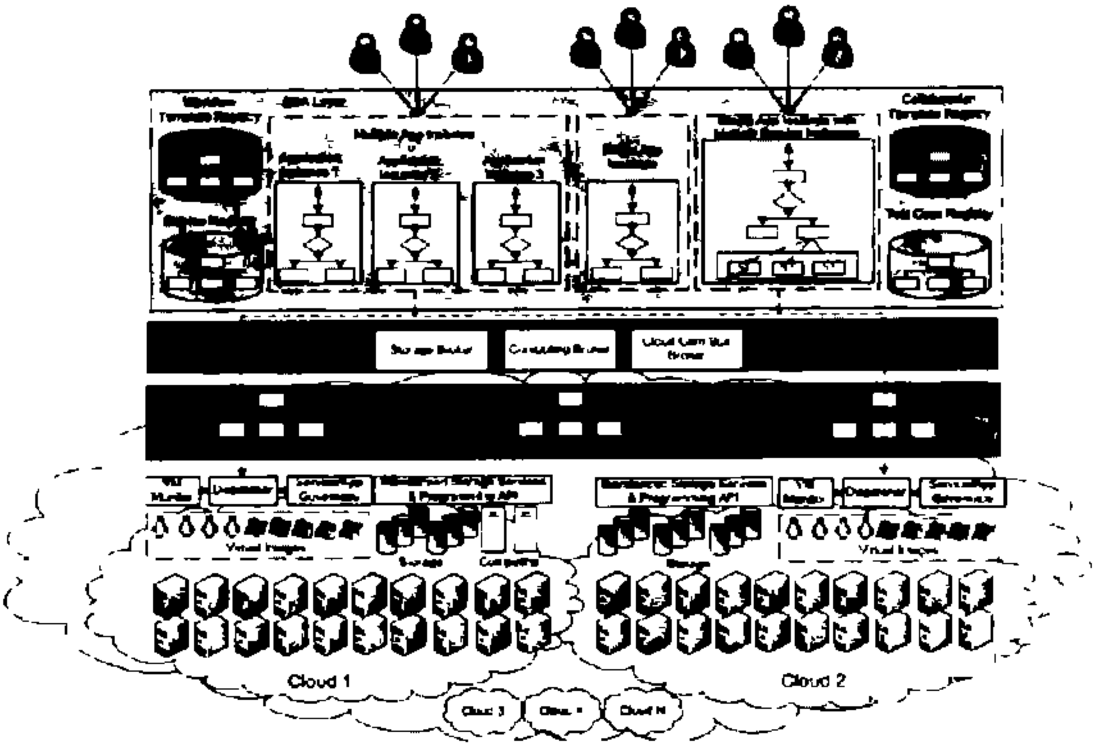

Figure 5 Service Oriented Cloud Computing Architecture (SOCCA) [33]

Wei- Tek Tsai et al [33] conducted a survey of existing cloud computing architecture and they exposed many cloud interoperability issues with current clouds To overcome these issues authors proposed a service oriented cloud computing architecture (SOCCA) that gives the mixture of cloud computing and service oriented architecture technologies This architecture is based on 4 layers that enable cloud applications to run on different clouds Cloud providers can build their own data centers and offer services on individual cloud provider layer The unique thing in this architecture is that they can be accessed via open standard interfaces and this is the only way through which multiple clouds provider can combine their offered services If any cloud provider implemented extra features which are not defined in the standards then cloud ontology mapping layer is used SOCCA has cloud broker layer which serves as manager between two clouds The responsibility of cloud broker is to publish cloud provider information and rank the services according to their reliability, availability, price and security The SOCCA also support multi-tenancy feature of cloud computing Google App Engine and Azure real platforms are used to deploy this architecture An application service is developed using Google App Engine that utilized Web Service Connector (WSC) WSC is a code generation tool that generates generic code using WSDL to access web services offered by Azure

2.6 Architecture and Protocol for Intercloud Communication

Jaime Lloret et al [32] described an architecture and protocol, which allows swapping information, data and applications among all interconnected clouds It is favorably flexible and authorizes to append new clouds easily The suggested architecture is compose of three layers (a) Organization layer is consist of Onodes, that organize the connection between Dnodes of clouds, (b) The responsibility of distribution layer is to transfer services and resources among clouds and the nodes involved in this layer are termed as Dnodes, (c) Access layer is established by the cloud nodes Authors also explained the analytical model of proposed architecture Suitability parameter is the most important part of Neighbor Selection method, which allowing the load balancing of nodes to achieve best quality of services To recover node failure author included fault tolerance procedure Results are generated performing simulation in a controlled test bench Results show that architecture is more viable and could work in all situations without overloading the system

2.7 CompatibleOne

In Figure 6 below, a high-level overview of the CompatibleOne platform architecture is provided in four quadrants These quadrants represent the four steps of the functional cycle of the platform and are also consistent with the four agents or cloud computing stakeholders identified in the NIST report

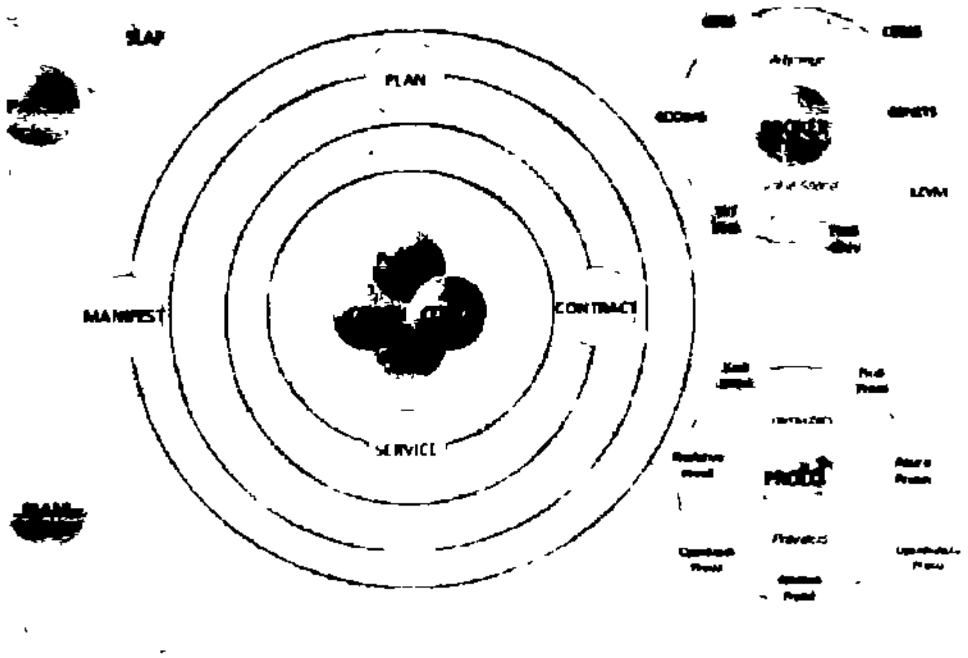

Figure 6 CompatibleOne Architecture [34]

The CompatibleOne is open source project and it is perceived that open innovation of cloud computing environment is enabled by open source and embracing the open standards That is why the CompatibleOne project is based on open standards Series of open standard have recognized by the project partner that deliver the background of the project [34]

Comprised standards are

- Distributed Management Task Force (DMTF)
- Cloud Data Management Interface (CDMI)
- Storage Networking Industry Association (SNIA)
- Open Cloud Computing Interface (OCCI)
- Open Grid Forum (OGF)

CompatibleOne has explains high level depiction of CompatibleOne architecture in four parts

The existing four parts portrays the steps of architecture's functional cycle It is also compliant with the NIST cloud computing stakeholders or the four agents

2.7.1 Step One: Handling the user's requirements

The first phase of the handling user's requirement is user interactivity CompatibleOne Resource Description Schema (CORDS) component will document the requirements of the users, unfold their required infrastructure All the services which is to be supplied, specification of limitations and technical norms are the concern of this document The output of CORDS document is the Manifest

2.7.2 Step Two: Validation and provisioning plan

In the second slice, SLAP MASTER allocation engine is responsible for collecting the Manifest document Transactional event management is recruited by the SLAP MASTER and intended to be used by the billing system The manifest is passed from the CORDS parser, concern of CORDS parser is parsing of manifest, infrastructure feasibility, validating the XML syntax and the plan conformance In feasibility process, firstly the document will examined by the CORDS parser and limit the service providers inside the CORDS parser community which is able to apprehend and satisfy the terminologies written in the manifest document After the successful completion of this process, the XML description will translated into certified provisioning plan Provisioning plan is a proposal which is comprised of information regarding infrastructure labeled by the user in the manifest document.

2.7.3 Step Three: Execution of the provisioning plan

The validated provisioning plan document will be passed to the CORDS Broker which will actually starts the provisioning process CORDS Broker containing all the CompatibleOne service provider Following is the list of services implemented by the CompatibleOne project

- COOBAS (CompatibleOne Ordering Billing and Accounting Services)
- COES (CompatibleOne Elasticity Scalability Services)
- COMONS (CompatibleOne Monitoring Services)
- CONETS (CompatibleOne Network Services)
- COEES (CompatibleOne Energy Efficiency Services)

- EZVM (VM Creation of System Services)
- UniData (Unification of Data storage and block storage for the EZVM)
- PaaS4Dev – OSGi Interface Management

The process written in the plan document will be implemented by the Broker with the assistance of these above stated services and very essential for the formation of control graph All the discrete components managed by the framework provided by the control graph

2.7.4 Step Four: Delivering the cloud services

Fourth chunk describes about delivering the cloud services In this section, CORDS Procci will allocate the resources negotiated by the Broker, under the supervision of Control Graph The resources are settled by the broker with the help of CORDS Procci The process is expressed in the Manifest document and succeed by the plan This is the required service which will be delivered to user Client/server interfaces are used to communicate between different components which is defined using OCCI standard and implemented inside the CORDS Procci OpenStack and Open Nebula are working on the basis of CompatibleOne

2.8 Interoperability in mOSAIC

The collection of individual components that represent mOSAIC's proof-of-concept prototype solutions are depicted in Figure 7 Individual components, like API implementations, application developing tools, vendor modules and so on are part of the integration platform

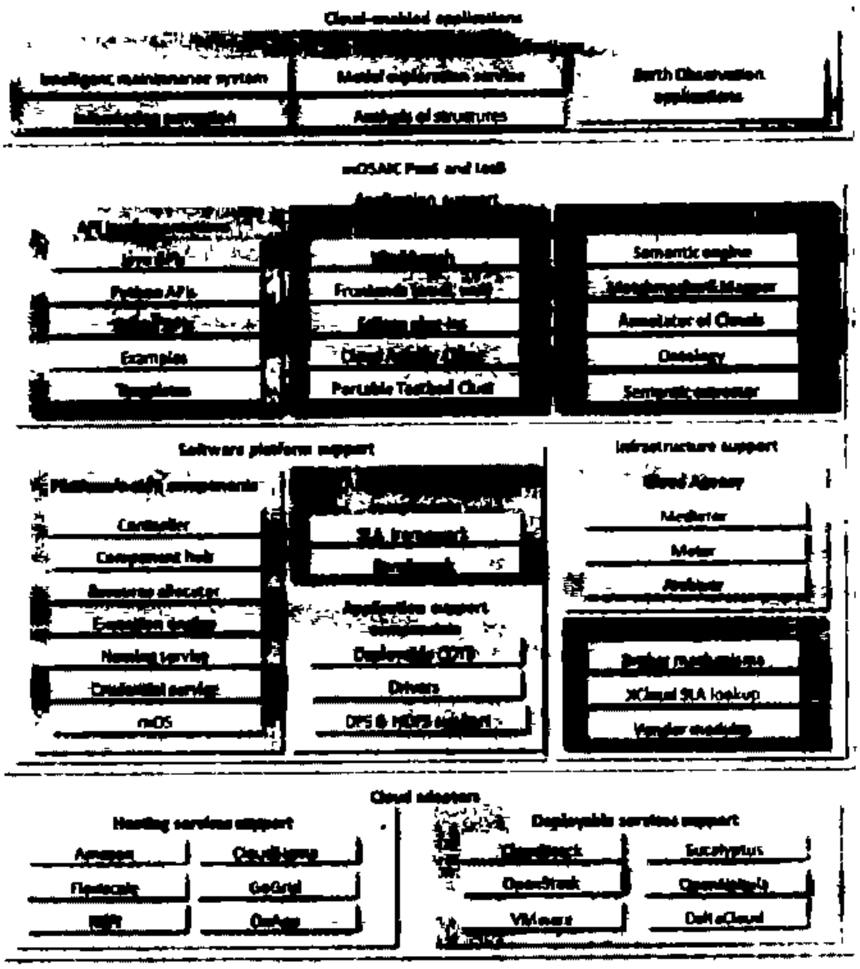

Figure 7 Component view of mOSAIC [41]

The mOSAIC is a project which aims to develop the open source platform so that it can serve as an environment for competition between cloud service providers In this direction mOSAIC will develop two main components The Resource Broker and Application Executor The resource broker consist of client interface and cloud agency which is responsible for resource negotiation The resource broker describes the need of application The application executor describes the set of tools which includes the methods of validation of application specifications and generates the SLAs accordingly The application executors looks after the execution of applications in accordance with the defined SLAs [41]

2.9 Cloud Storage Abstraction Layer (CSAL)

Cloud storage abstraction layer (CSAL) supports portable cloud applications and three types of storage abstractions provided by the storage layer i e Blobs, Tables and Queues Across the platforms there is unified view of CSAL layer provided by the implementation [35] Metadata is managed to exploit the storage service in multiple clouds Assimilation of storage on higher level is not very useful because data has different perspective for different types of users Data can be mentioned as XML, Binary data, SQL database tables. In cloud computing, four data storage mechanisms are used

2.9.1 BLOB Storage

It is stands for binary large object Images and other binary structures are stored in blob storage It is mostly useful for distributed and remotely accessible files The range of storage of such files is in Gigabytes and Terabytes Examples are Azure Blob Storage and Amazon S3

2.9.2 Table Storage

For large data items the storage is used is SQL Amazon SimpeDB is its best example Common SQL-based database is incapable of relational query and strong transactional semantics which deliver feeble consistency But CSAL providing the high availability storage for huge data

2.9.3 Queue storage

Queuing semantics is vital part of cloud storage which is used for definite item delivery Customer-facing interactive services i e websites are decoupled by these queues Examples for such type of storage is Azure Queue Storage and Amazon SQS

2.9.4 Relational table storage (SQL)

Transactional SQL database in contrast with Table storage is much more secure because it provide transactional guarantees and rich query semantics. Some SQL database services are often offered by the vendors to deal with such cases

2.10 Open Cloud Computing Interface (OCCI)

The Open Cloud Computing Interface contains a course of action of open gathering lead determinations passed on via Open Grid Forum It's a Protocol as well as an API for an extensive variety of Management endeavors OCCI was at first begun to make a remote organization API for IaaS model based Services, mulling over the change of interoperable services for typical assignments including deployment, scaling and monitoring It has resulting to form into a versatile API with portability, integration, interoperability and innovation while so far offering an abnormal state of extensibility OCCI is suitable to serve not only IaaS but it can serve many other models such as PaaS and SaaS The working group of OCCI is also working on some additional specifications which are in progress such as OCCI XML Rendering, OCC JSON Rendering, OCCI Billing and Monitoring, OCCI PaaS extension, OCCI SLAs extension [36]

2.11 Comparative Analysis

It has been observed from the literature that there exist many architectures which are providing the cloud federation These architectures includes different aspects such as communication libraries and security mechanisms Some architectures provides only federation among cloud service providers and other provides interoperability at a quite abstract level All the aspects reviewed from the literature categorized as general aspects Following is the summary of related aspects of federation and interoperability from the literature in *Table 1*

Table 1 Comparison between Frameworks/Libraries

Aspects \ Frameworks	Contrail	FCM	RASIC	ICAF	SOCCA	CO	mOSAIC	OCCI	CSAI
Service Monitoring	YES	N/A	YES	YES	N/A	YES	YES	YES	N/A
Communication Libraries APIs	YES	N/A	YES	N/A	N/A	N/A	YES	YES	YES
Federation Support	YES	YES	N/A	YES	N/A	N/A	N/A	N/A	N/A
SLA Management	YES	N/A	YES	YES	N/A	N/A	YES	YES	N/A
Mapping Component	YES	YES	N/A	N/A	N/A	N/A	YES	N/A	YES
Metadata	YES	N/A	YES	N/A	N/A	N/A	N/A	N/A	YES

Identity Management	YES	N/A	N/A	YES	N/A	YES	N/A	N/A	N/A
Single Sign On	YES	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Authentication	YES	N/A	N/A	N/A	N/A	YES	N/A	N/A	N/A
Authorization	YES	N/A	N/A	N/A	N/A	YES	N/A	N/A	N/A
Brokering Mechanism	N/A	YES	N/A	YES	YES	YES	YES	N/A	N/A
Service Deployment	N/A	YES	N/A	N/A	N/A	N/A	YES	N/A	N/A
Performance	N/A	YES	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Open Standards	N/A	N/A	YES	YES	YES	YES	N/A	YES	N/A
Maintenance	N/A	N/A	YES	N/A	N/A	N/A	YES	N/A	N/A
Ontology	N/A	N/A	YES	N/A	YES	YES	YES	N/A	N/A
Interoperability Engine	N/A	N/A	YES	N/A	N/A	N/A	N/A	YES	N/A
Service Publishing	N/A	N/A	YES	N/A	N/A	N/A	N/A	N/A	N/A
Reporting	N/A	N/A	YES	N/A	N/A	N/A	N/A	N/A	N/A
Resource Scaling	N/A	N/A	N/A	YES	N/A	N/A	N/A	N/A	N/A
Multi-tenancy	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Metering	N/A	N/A	N/A	N/A	N/A	N/A	YES	N/A	N/A
Portability Support	N/A	N/A	N/A	N/A	N/A	N/A	YES	YES	N/A
Billing	N/A	N/A	N/A	N/A	N/A	YES	N/A	YES	N/A

All the listed architectures and frameworks focusing on different aspects Interoperability Engine is the major component of interoperability for interoperation between multiple clouds but after the literature comparison this component got less evidence

2.12 Limitations

It has been analyzed from literature that existing libraries/architectures are providing solutions for cloud federation But they discussed interoperability at a very abstract level Furthermore these architectures are not mature enough to provide the details about interoperability There is a need to develop comprehensive reference architecture for interoperability module

In this chapter we have studied the related work to our research topic where interoperability has been discussed as minor component In next chapter we will describe our research methodology that how research will be conducted Initially we have searched research papers, journals and conference papers based upon the synonyms, strings and keywords

Chapter 3

Research Methodology

3 Research Methodology

In this chapter we will briefly describe the methodology which is being used in order to accomplish the thesis As said by Creswell (2013) research is a systematic inquiry which objects to analyze the theory or a research objective The main purpose of doing a research is to suggest new interpretation of data, which directs to new questions for future research, contribution of knowledge [42]

The analytic approach for this research depicted in Figure 8, is to formulate a research objective then identifying the interoperability issues such as vendor lock-in between different services provider by studying literature Literature review is the method which is used to conduct the research [43]

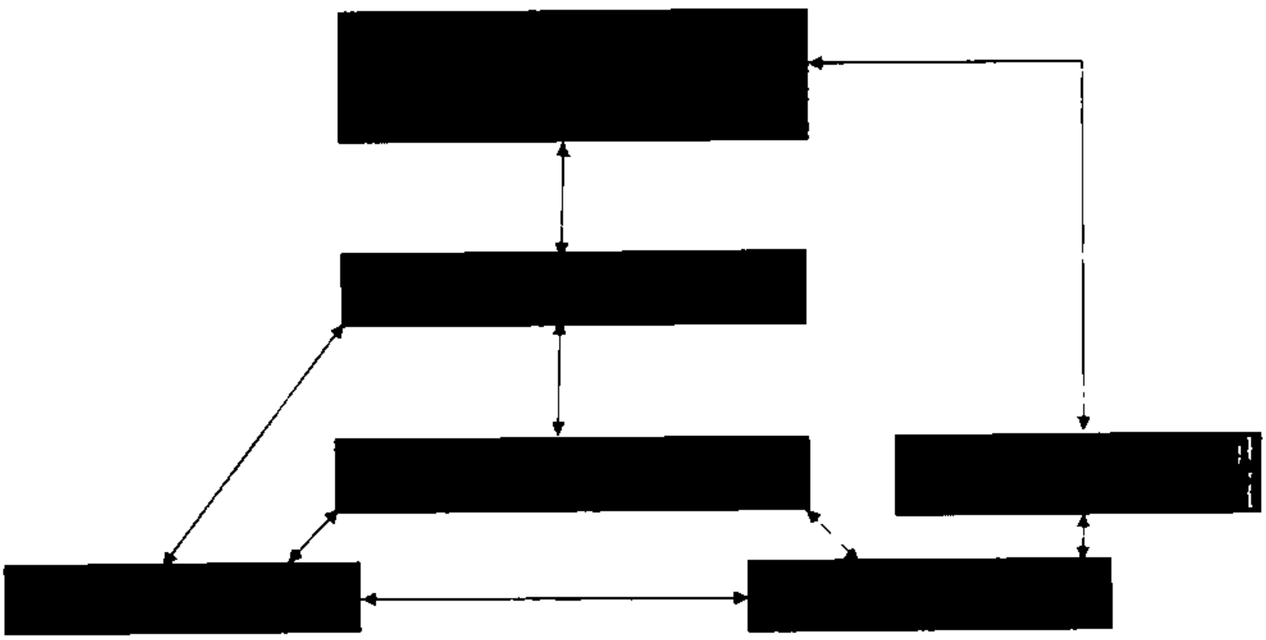

Figure 8 Research methodology used

3.1 Search process

The search method initiated by the systematic reading to and scanning of different research papers, journals, conference papers published in IEEE, springer, ACM, Elsevier, CiteXeerS and ScienceDirect Apart from that, research papers published by the technology giant such as Microsoft, IBM, google and Amazon

Different keywords and terminology such as cloud federation, cloud interoperability, cloud portability, mediator between clouds, cloud integration and synonyms such as mixing of two clouds, cloud cooperation, cloud alliance, cloud association and cloud partnership with combination of strings were selected in order to search the most relevant research papers about interoperability Some of the synonyms and keywords are rephrased back and forwarded straight and by using the OR operator in order to utilize the efficient searching mechanism [44]

The research papers which are published in digital library i e IEEE, Springer, ACM Elsevier CiteXeerS and ScienceDirect were accessed by using International Islamic University IP Other internet resources were also used in order to search the articles, blog posts and papers published by cloud community and cloud vendors [45]

3.2 Reading

The research papers, journals and articles we found which is related to our topic (interoperability in multi cloud environment) and the subject areas like (problems in obtaining services from multiple cloud service providers) We started reading from abstract of that research paper We have tried to find the research problem statements, research methodologies being stated in that research paper Apart from that, if researcher has mentioned any clear solution to that problem that is described in abstract section of that research paper After reading the abstract as we get to know that we are able to extract more information related to our topic, we further decided to proceed reading the introduction, related works, and methodology, discussion and conclusion sections of those research papers This provide us a clear overview what the researchers has contributed or either add any knowledge [46]

Gaps in the research area of cloud computing We also get to understand if there are any missing information or there requires further works which need to be implemented in the near future

3.3 Evaluation

To investigate the research objective we read the articles that were related to our research topic The assessed articles provided information about interoperability between multiple cloud service providers Most of the research papers provided more information about federation rather than interoperability Interoperability has more components which should be discussed in the papers but having fact that they are giving very least information The evaluation was done based on content of the assessed literature We have extracted and collected the data from difference research papers, journals and conference papers After that we designed a questionnaire in order to get evaluation of proposed architecture We found cloud experts from industry for getting feedback on evaluation Selection criteria of cloud expert was that they shall be from cloud industry and have expert level experience We selected two cloud experts and sent them our questionnaire On the basis of their feedback, we analyzed the answers and we got our results regarding efficiency and appropriateness of our proposed architecture [43]

Interoperability in multi cloud environment is the new era and much of the research is going in the lab to the technical giants and papers are being published both from those companies as well as the employees blogs and other sides which we critically evaluated the content of those readings as well as authors authentication and credibility

In this chapter we have described our research methodology that how we have searched the research papers, journals and conference papers We have collected data from different papers Based upon the research methodology, we have proposed a new reference architecture

Chapter 4

Proposed Reference Architecture

4 Proposed Reference Architecture

In this chapter, we have proposed a reference architecture for interoperability in cloud computing The Cloud Interoperability Reference Architecture (CIRA) Its association with other components together provides better solutions for interoperability in a multi cloud environment It provides a bridge between multiple cloud service providers Below are the components in *Table 2* which have been selected and categorized in terms of core components, support components and general components These components are referred from different research papers, journals and conferences by conducting literature review as part of our research The detailed reference id given below

Table 2 Reference of Interoperability Components

Sr. No	Components of Interoperability	References
1	Interoperability Engine	[12] [36]
2	Rule Engine	[47]
3	Billing	[34] [36]
4	Metering	[41]
5	Service Ontology	[12] [33] [34] [41]
6	Service Monitoring	[12] [31] [34] [36] [40] [41]
7	Auditing & Reporting	[48] [49]
8	Incident Reporting	[50]
9	Communication Libraries	[12] [31] [35] [36] [41]
10	Resource Change	[51]
11	Elasticity Rule	[54] [55]
12	Pricing and Rating	[56]
13	Identity Management	[31] [34] [40]
14	Authentication	[31] [34]

15	Authorization	[31] [34]
16	Single Sign On	[31]
17	Privacy	[51] [52]
18	SLA Management	[12] [31] [36] [40] [41]

The CIRA targets the problems outlined in introduction so that many providers can adapt such architecture to fulfill maximum interoperability requirements Figure 8 below, shows the Cloud Interoperability Reference Architecture

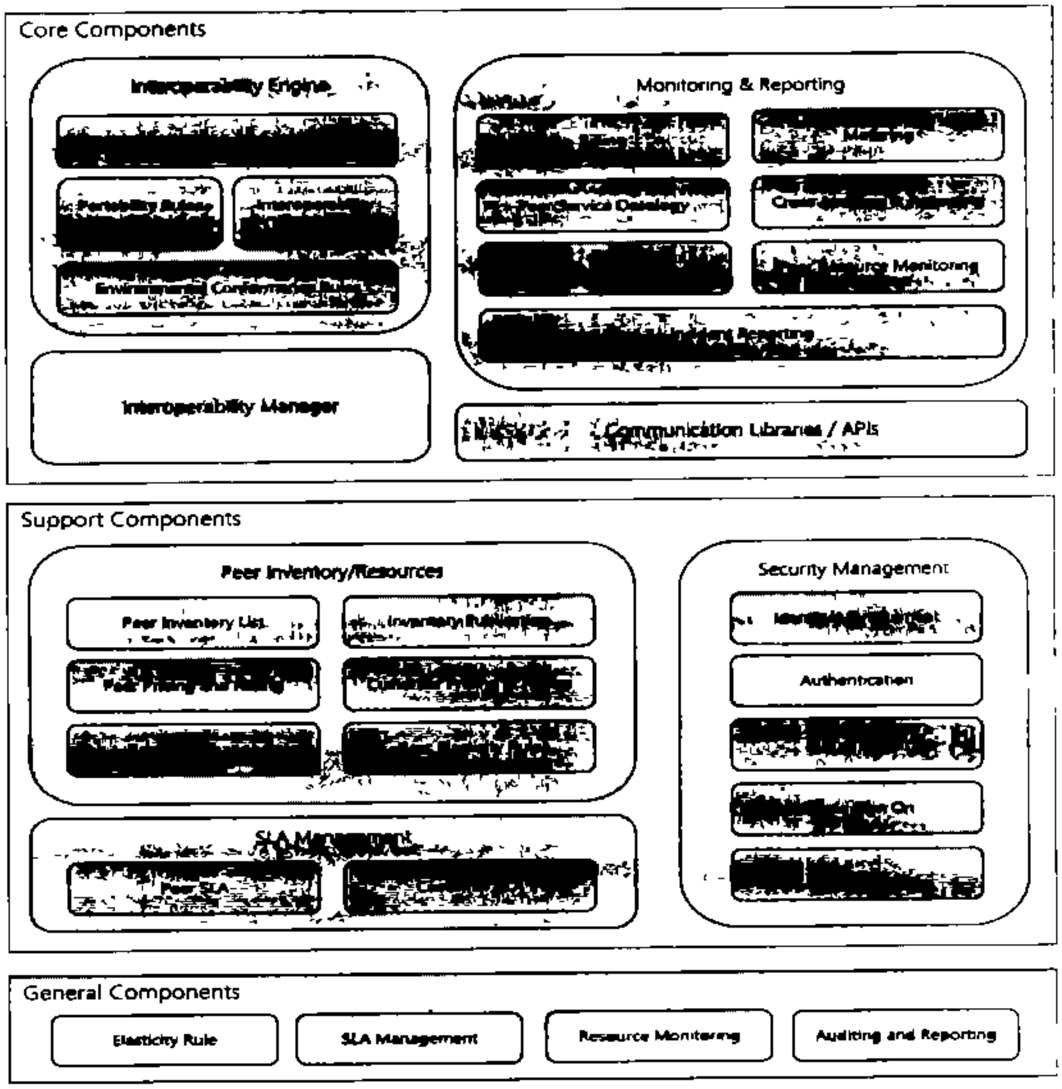

Figure 8 Cloud Interoperability Reference Architecture (CIRA)

Interoperability architecture consist of core components of Interoperability, support components and general components

A. Functional view

The architectural functional view describes the working of single component independently

4.1 Core Components

Core components are those which are directly related to the cloud interoperability These components together explains the process of interaction of multiple cloud service providers Without core components interoperability activity cannot be performed

4.1.1 Interoperability Engine

Interoperability engine allows seamless data exchange between entire components It provides integration between all components through a single platform that can be easily maintained [12] [36]

Rule Engine

In Cloud computing, there are many cloud service providers offering diverse services The essential part of planning and designing the use of cloud services is to comprehend the interoperability and portability If the interoperability and portability rules are described well then it will be beneficial to pinpoint the best fitted solution [47]

Interoperability Rules DB

Generally interoperability is the aptitude of two or more systems or applications to share information seamlessly From the perspective of cloud computing, interoperability means exchange data at different layers between cloud service providers In cloud eco-system interoperability intends to provide the compatibility between private cloud, public cloud and other systems to understand the each other's application interfaces, authentication and

authorization mechanisms, configurations and data formats etc Interoperability rule DB comprises of the rules which will allow the two systems to interoperate with each other
The supreme concern of interoperability is the interaction between cloud service provider and customer This interaction is fulfilled by using the prescribed interface or API There are different interfaces for different aspects of cloud services The perfect interoperability is the one which comprises of standardized interfaces which make it easy for customer to shift from one cloud service provider to other without any fault

Portability Rules DB

Portability rule DB consist of rules that make it easy to migrate data between two cloud service providers The term portability implies that the ability to move an entity and its allied data between one or more cloud service providers so that it is functioning on other system Lack of portability leads to the problem that it require more strength to convert the entity from its existing format to the target system's format Further portability has two concerns application portability and data portability
Data portability: it facilitates to simply transfer, move or copy the data among different cloud application or environments with no need of re-entering the data It might be accomplished when both the source and target services have exactly the same data formats If both does not matches the formats then commonly available tools are used to accomplish the transformation
Application Portability application or application components are transferred from one cloud service to other cloud service also run in target environment It is about, application is developed in one environment and run in other environment It will help to remove the gap between the development and deployment settings The application code does not need to make substantial changes but it might entail relinking or recompiling for the target environment

Environmental Conformance Rule

Environmental compliance means conformance with the legal, regulations and laws, standards such as ISO PCIDSS, HIPPA or any other technical requirements such as environment permits to operate in

4.1.2 Monitoring and Reporting

SLA lifecycle is monitored by the monitoring service It governs the completion of the SLA and then report it to the billing service Cloud systems are monitored by some services and then these services produce its performance report Which further monitor the other services It assembles the matrix and delivers a best analysis and repot [12] [31] [34] [36] [40] [41] 48]

Billing

The concept of cloud billing is to generate bill for customer by using pre-defined policies as they have used from available resources Billing module supports port type so that it can fulfill all functional requirements It generates the bill and suspends account It stores bill billing repository for later use [34] [36]

Metering

Metering concept is when a customer has unlimited access to available resources but it pays whenever it utilizes any resource from available pool This service is very commonly used in current era of information technology This service enables service providers to make a big infrastructure for customers and charge them whenever they uses resources Metering is becoming common in consumers market Now a days ISPs are widely using this services and charge the customers upon usage of services [41]

Peer Service Ontology

Ontology is termed as the data semantics are described by the Meta information To bridge the gap between the computer agent and the human, ontology provides the common understanding of a realm It specifies the set of concepts and association between the concepts and it also useful for the information retrieval when user request a query [12] [33] [34] [41]

Cross Auditing and Reporting

Cross auditing means an auditor is auditing the department or function where they are not working In terms of interoperability, an auditor can audit the cross platform or peer service provider [48] [49]

Service Monitoring

Service monitoring component monitors the performance, continuity and efficiency in virtualized and on demand environment Also checks the security of all the components that supports service delivery hardware, software and services in the data center [12] [31] [34] [36] [40] [41]

Peer Resource Monitoring

Peers confirms the optimum performance, effectiveness, security and stability in virtualized manner by monitoring service Peers not only ensures the performance of network infrastructure running applications and all the components but also organize the better provisioning of resources

Incident Reporting

Grobauer, B et al [50] described that incident is basically the loss of integrity which affects job of the core service of network and information system If these incidents are reported to concerned person then it have deemed significant impact So Incident reporting is the technique by which the cloud service provider informs the authority about the occurring incidents Incident reports provide the validation on the efficacy of the security measures, and the security of the acquired cloud services It includes the incidents such as hacking incident, system RAM overflow etc

4.1.3 Interoperability Manager

Interoperability Manager is a central component which accomplishes and organizes the operations of all the components inside the model When any cloud service provider want to do interoperability / portability first interact with this component A unified API is used to manage the resources among different clouds which provides seamless interconnection across different platforms A data management issue in cloud federation is also resolved by the Interoperability manager

4.1.4 Communication Libraries/API's

Communication libraries works between different modules as an intermediate It is a set of code protocols and tools which builds software applications APIs some time comes in the form or a library which includes the specifications for structure, variables, objects and classes APIs or communication libraries work as a specification or remote calls which are exposed to module or consumer of that API [12] [31] [35] [36] [41]

4.2 Support Components

Support components are not crucial for the main function but these components ensures the efficiency of the system

4.2.1 Peer Inventory and Resources

The record of services and all the other resources provided by the peers are described in this section

Peer Inventory List

It consists of the services list which are any cloud service provider is providing for customers

Inventory Publications

Inventory publication is what how to resources, services are made available to outer world publically

Peer Pricing and Rating

Peers have detailed pricing structures that is based on pay-per-use model This makes it easy for peer to track the particular price of running resources There are many aspects of pricing structures i e monthly storage space allocation Some peers have concealed service charges which are mentioned in their service level agreements (SLAs) [56]

Unit Price Plan cost for one unit is constant for one resources but it varies for different resources and usage of that resources [56]

Period Based Plan units utilized per day, week, month and year Time of the day is also under consideration Exclude the holidays [56]

Customer Pricing and Rating

There are 3 models used by cloud service providers (customer perspective) Tiered pricing, unit pricing and subscription pricing Tiered pricing is when service provider charges upon each different specification like CPU and RAM usage Per unit price means when a service provider charges upon exact resource usages Subscription based usage is common in software – as-a-service in which customer is being charged upon subscription of services which costs per user per month

Resource Change

Resource change means a service provider is changing resources in terms of upgrade or downgrade any CUP resources, RAM resources, joining new nodes, deleting nodes or can make any kind of changings in terms of configuration [51]

Peer Elasticity Rule

Elasticity is the mechanism which specify the rules for allocation and de-allocation of additional capacity at deployment time According to these rules when some event occur it triggers the automated actions which alter the service capacity Commonly elasticity could be achieved by three dimensions, such as Cost, Quality and Resources As to satisfy the customer requirements it enables to increase or decrease the Cost, Quality and available resources [54] [55] The capacity changes comprises

- **Scaling up** increase or decrease the reserved storage capacity of the service
- **Scaling out service** components are added or eradicated

4.2.2 Security Management

This section involves some important security challenges when using cloud services, including

Identity Management

Identity management is the combined model of process, procedures and technologies that allows the system to perfectly recognize entities and control the use of information among them It labels the management of individual identities, their access rights, and privileges within or out of the system It involves the several components' role management, access management, federated identities, password administration and directory services etc [31] [34] [40]

Authorization

After successfully logged in to the system, authorization delimits the access rights of user that what the resource requester is permitted to do On the basis of identity, system can recognize the user's access rights but for some situations system may require some additional traits such as user's role or title along with identity The most important security attribute, referential integrity is maintained by the authorization [31] [34]

Authentication

Authentication is a process which should be applied to confirm the identities of user and service provider For the sensitivity of application and information assets, authentication assurance levels are suitable This process permits the system to first identify the user by its username then secondly user is validated by providing password [31] [34]

Single Sign On

Single sign on (SSO) [31] is a user authentication technique in which user is authorized to access multiple applications by providing name and password once During a certain session, user has been given rights to all the applications and does not prompts for authentication when user shift among two applications It works on central service technique, which creates cookies when you log on first time Therefore admittance to second application may not require credentials again it means you are already logged in

Privacy

Privacy [51] [52] is the high priority security attribute It is more critical because the customer's data and the business logic exist in the suspected servers, which are sustained by the cloud provider Consequently the personal profile and the confidential data is revealed to community and professional competitors Other security aspects which effect the privacy may include

- **Integrity** privacy is somehow preserved by the integrity, which satisfies that data is not corrupted

- **Confidentiality** confidentiality is necessary to keep the private data save from being revealed
- **Accountability** it is contrary to privacy if customer's private data leaked out then provider is not accountable for such loss

4.2.3 SLA Management

The level of service is defined by the service level agreement which is basically the bond between customer and the service provider It outlines the guaranties, warranties, priorities of services and understanding about services It is necessary utility for cloud computing to regulate the usage pattern [12] [31]

Peer SLA

The services provided by the peers for cloud customers are followed by the contract called peer SLA For example, a telecom service provider offers other basic services to all the customers and also provide assured maintenance as a part of deal with universal charging [40] [41]

Customer SLA

A customer SLA is an agreement with a distinct customer group, it deals with all the services using by that particular customer For instance, SLA among cloud service provider and the HR department of a large corporate system for the services such as HR planning, training and development, talent management, recruitment selection system etc Customer concerns regarding service level agreements [36] includes

- Customer will opt that service provider is going to provide how much service
- Delivery of IT services are the customer's concerns
- The start stage of new product or technology decided by the customer like development or test stage

4.3 General Components

General components comprises of the information for SLA such as capacity availability and performance, elasticity rules, resource monitoring, and auditing and reporting

4.3.1 Elasticity Rules

From service provider's perspective, elasticity is the aptitude to vary the capacity of system for instance storage, I/O bandwidth, CPU etc offered for a given service It allows their customers to make changes in their service capacity according to their requirements [54]
In the customer's view, elasticity is the automated means to swiftly demand a service and later on freed as many resources as needed Preferably, it is the provisioning of unlimited space for the customer so that customer can be acquired as much quantity at any time Elastic resources are able to run on the changed environment easily [55]

4.3.2 SLA Management

Service Level Agreement (SLA) [12] [31] [36] [40] [41] is a contract between Cloud Service Provider and its customers which specify the terms and conditions and customers' expectation for service provision
SLAs examine the service provider s performance and quality in many ways Some aspects that SLAs may state consist of

- Availability availability of service in percentage
- Number of the customers can be attended simultaneously
- On the basis of standard criteria, service provider's performance will be evaluated
- Application response time
- Response time for the assistance of problems
- Service usage figures will be mentioned

In addition to these aspects, SLA may also specify the strategy for the occurrence of a contract breach Once the contract founded, should be revised and modified to reveal the changes in the technology

4.3.3 Resource Monitoring

Resource monitoring component gather information of system resources and assist them in decision making process Many vital cloud computing operations such as, load balancing network analysis, fault detecting, fault recovery, management and job scheduling relies on resource monitoring When the failure arise it also monitor the state of resources [12] [31] [34] [36]

4.3.4 Auditing and Reporting

Reporting and Auditing Monitor user operations, generate reports, etc
It provides the cloud service providers a manner to make their performance and security data willingly available to prospective customers The audit plan specify a standard way to describe the security and performance statistics The standardized information make it easy for the customer to compare the service providers and investigate the data [48] [49]

B. Structural View

The architectural structural view describes the working of multiple components altogether It describes the process or flow of multiple components

Interoperability Manager

Service request will be sent by the user to interoperability manager Interoperability manager is the main worker component as described in Figure 9, it manages all the requests from the user Once request is submitted to interoperability manager Interoperability manager shall requests the rules from rule engines according to the service request by the user Rule engine imports rules from interoperability rule DB and portability Rule DB in accordance with interoperability or portability Additionally rule engine shall check compliance with the environment conformance rules Since client need services from more than one cloud service provider for this purpose user will request interoperability manager for peer cloud services Interoperability manager will use the communication libraries to interact with peer cloud

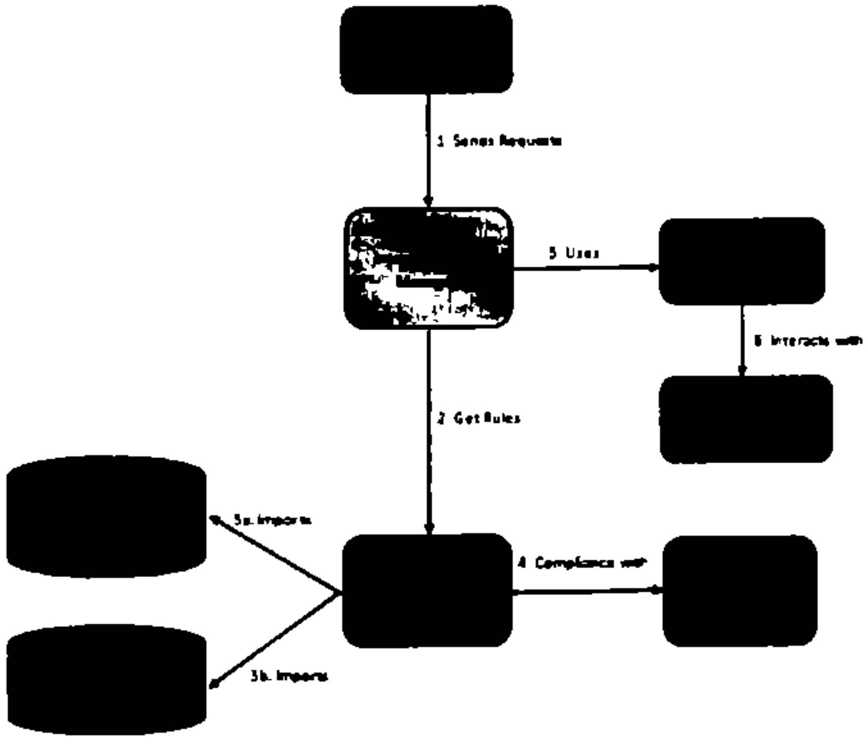

Figure 9 Interoperability Manager

Metering and Billing

Metering shall get service utilization data from service monitoring component and store the service usage report to metering log Billing requests the service usage information from metering module and stores the bill in billing record as depicted in Figure 10

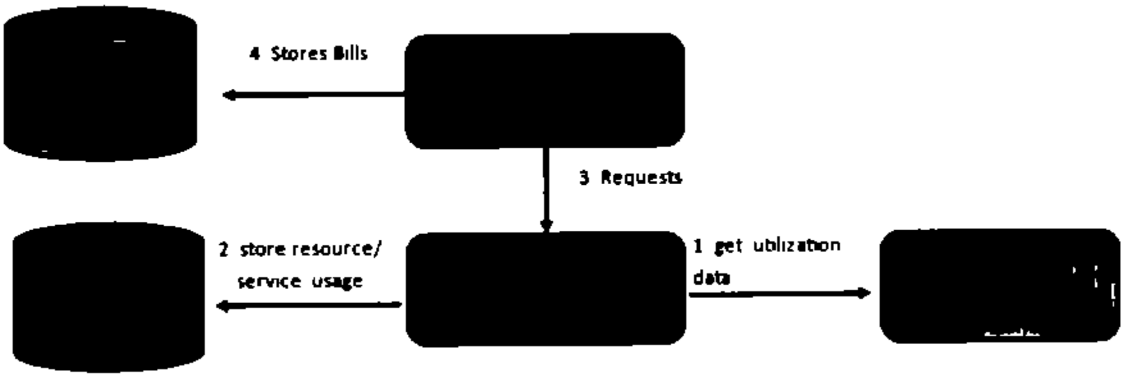

Figure 10 Metering and Billing

Monitoring and Reporting

In Figure 11 below, monitoring and reporting process is described Peer service ontology submits report to peer resource monitoring module Incident reporting module will submit the incident report to auditing and reporting module in that way peer can access audit report by using peer resource monitoring module

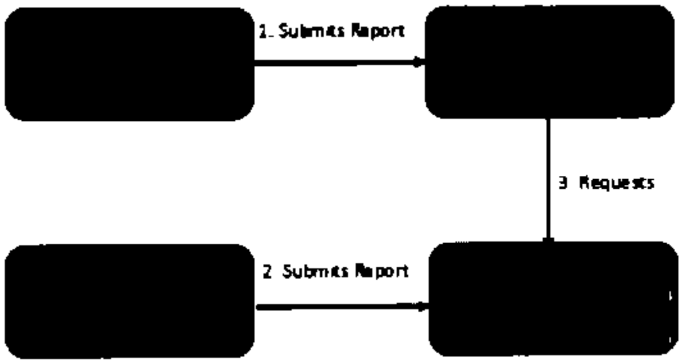

Figure 11 Monitoring and Reporting

Security Management

Identity management manages access privileges of the user Authentication is being done using single sign on as illustrated in Figure 12 Single sign on pulls the policies from policy server

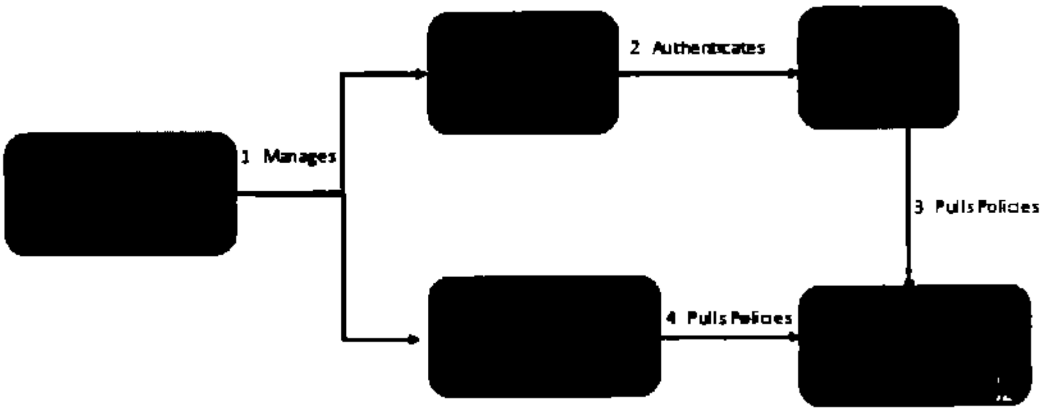

Figure 12 Security Management

SLA Management

As shown in Figure 13, Peer SLA and customer SLA module submits the agreement to negotiation process component on the basis of which it generates the common SLA

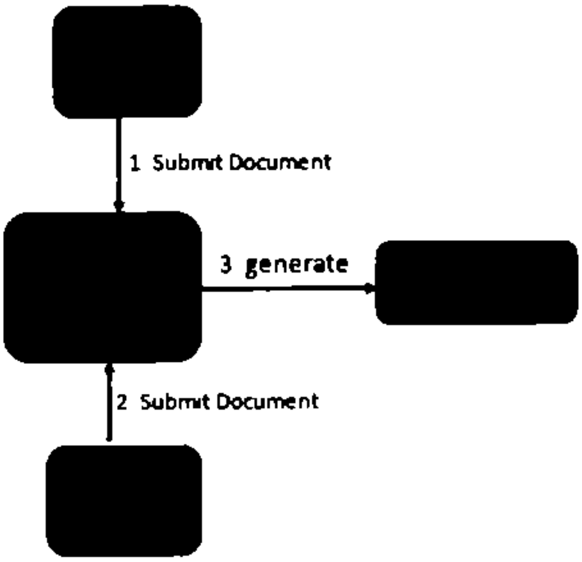

Figure 13 SLA Management

In this chapter we have proposed a reference architecture based upon the research In next chapter, we will conclude the theoretical evaluation of proposed architecture

CHAPTER 5

Theoretical Evaluation

5 Theoretical Evaluation

Based on the research design, this study comprises results and analysis by performing expert review which is mentioned earlier in methodology chapter The expert review was conducted through the evaluation questionnaire where some questions related to proposed architecture were asked from researchers and industry professionals The results were also compared and analyzed with the results of other experts The results from expert reviews helped to analyze the efficiency and importance of proposed architecture The reason of low number of respondent was it was conducted in month of November which is majorly the mid exams season of students and for professionals its end of year closings Therefore the questionnaire which was distributed over the email was not successful and there were no responses from most of the industry professionals However the evaluation survey provided the important findings

5.1 Expert reviewer 1

The first reviewer is cloud solution architect He has 12 years of experience in cloud solution providing industry He has various cloud certifications and currently working in BigFive cloud companies

If we see *Table 3*, most of the components which are proposed in this thesis are missing from existing framework Only few components are adequately present in few frameworks Few are partially identified which doesn't cater the actual need of interoperability For example, Rule Engine, Interoperability Rule DB, Environmental Conformance Rule, Cross auditing and reporting and incident reporting is missing from each framework which are mentioned in Table A These components are stringent requirements are interoperability The proper explanation of these components can be found in chapter Number 4

On the other hand, Peer service ontology and peer Resource Monitoring is partially defined in RASIC, SOCCA, CO and mOSAIC Partial means, those frameworks explained mentioned components in general and in proposed reference architecture, those components are specifically addressing for peer service provider with respect to interoperability The components which are adequate in mentioned framework addresses completely for peer service providers

Table 3 Core Components of Interoperability Reference Architecture

Reference Architecture	Contrail	FCM	RASIC	ICAF	SOCCA	CO	mOSAIC	CSAL	OCCI
Rule Engine									
Portability Rule DB							✓		
Interoperability Rule DB									
Environmental Conformance Rule									
Interoperability Manager			✓						
Billing						✓			✓
Metering							✓		
Peer Service Ontology									
Peer Resource Monitoring									
Cross Auditing and Reporting									
Service Monitoring	✓		✓	✓			✓		
Incident Reporting									
Communication Libraries API	✓		✓	✓		✓	✓	✓	

If we see above *Table 4*, Peer Pricing and Rating, Peer Elasticity Rule, Customer Pricing and rating, Inventory Publication, Resource Change and pricing and missing supporting components from above mentioned frameworks The remaining proposed components such as peer inventory list, identity management, authentication, Peer SLA and Customer SLA and adequately addresses for mOSAIC, CO, ICAF and Contrail Rest of the framework doesn t addresses those components General Components which are addressed in our current study that includes elasticity rule, SLA management, and resource monitoring, auditing and reporting Although these components are general but important which are adequately addresses for interoperability in Contrail, RASIC, mOSAIC and OCCI

Table 4 Support Components of Interoperability Reference Architecture

Reference Architecture	Contrail	FCM	RASIC	ICAF	SOCCA	CO	mOSAIC	CSAL	OCCI
Peer Inventory List							✓		
Peer Pricing and Rating									
Peer Elasticity Rule									
Customer Pricing & Rating									
Inventory Publication									
Resource Change									
Identity Management	✓			✓		✓			
Authentication	✓					✓			✓
Authorization	✓					✓			✓
Single Sign On	✓								✓
Pricing									
Peer SLA							✓		
Customer SLA							✓		

If we see above *Table 5*, resource monitoring and auditing and reporting are required components which should be adequately addressed like other components In our proposed reference architecture, those components are addressed adequately which fulfils the requirements of interoperability

Table 5 General Components of Interoperability Reference Architecture

Reference Architecture	Contrail	FCM	RASIC	ICAF	SOCCA	CO	mOSAIC	CSAL	O((
Elasticity Rule				✓			✓		
SLA Management	✓		✓	✓			✓		✓
Resource Monitoring				✓				✓	
Auditing & Reporting		✓							

As per the feedback of 1st reviewer, we have received feedback as yes on all the questions Reviewer has studied our reference architecture and confirmed that current information given in our architecture is detailed for interoperability architecture The categorization which was done after studying the literature review is also accepted by the reviewer When reviewing the selected components from literature review, reviewer also accepted that these components are suitable and appropriate for proposed architecture The reason is that some of the components are present in one framework but it's missing in other framework regardless of its importance As we have selected components, categorized in best way according to best practice and given detail in reference architecture

5.2 Expert reviewer 2

Second reviewer is Assistant Professor of Computer Science He is working in the area of Model Driven Software Engineering, in particular focusing on Model Transformations and Automated Model based Software Testing which has 5 years of teaching experience

In *Table 6*, as per the reviewer's perspective we have analyzed that proposed reference architecture's components are found missing in well-known framework which is lack of fulfilment of requirement For example, rule engine, portability rule DB, interoperability rule DB, environmental conformance rule including peer resource monitoring which are identified as a core components in proposed architecture are missing in Contrail and FCM Some of the mentioned components are also missing in ICAF, CO and SOCCA As compared to 1st reviewer some of the components are partially discussed in industry framework Those components are present in frameworks

Table 6 Core Components of Interoperability Reference Architecture

Reference Architecture	Contrail	FCM	RASIC	ICAF	SOCCA	CO	mOSAIC	CSAL	OCCI
Rule Engine									
Portability Rule DB							✓		
Interoperability Rule DB									
Environmental Conformance Rule									
Interoperability Manager			✓						
Billing						✓			✓
Metering							✓		
Peer Service Ontology			✓		✓	✓	✓		
Peer Resource Monitoring				✓					
Cross Auditing and Reporting						✓			
Service Monitoring	✓		✓	✓	✓		✓		
Incident Reporting									
Communication Libraries/ API	✓		✓	✓		✓	✓	✓	✓

As per expert reviewer, he has selected that authorization, authentication and SSO is not present in OCCI while the expert reviewer 1 has selected it as yes in OCCI While if we notice that peer SLA and customer SLA has been selected as present in both of the reviews Same is the result for some other components It means that these components are important to be added in reference architecture as depicted in *Table 7*

Table 7 Support Components of Interoperability Reference Architecture

Reference Architecture	Contrail	FCM	RASIC	ICAF	SOCCA	CO	mOSAIC	CSAL	OCCI
Peer Inventory List							✓		
Peer Pricing and Rating									
Peer Elasticity Rule									
Customer Pricing & Rating									
Inventory Publication									
Resource Change									
Identity Management	✓		✓	✓		✓			
Authentication	✓		✓			✓			
Authorization	✓		✓			✓			
Single Sign On	✓								
Pricing									
Peer SLA							✓		
Customer SLA							✓		

As per this reviewer, the general components has been analyzed by both of the reviewer are more or less same Most of the components are selected as present in ICAF which is the famous framework in industry So it means that these general components are also important for our reference architecture as shown in *Table 8*

Table 8 General Components of Interoperability Reference Architecture

Reference Architecture	Contrail	FCM	RASIC	ICAF	SOCCA	CO	mOSAIC	CSAL	OCCI
Elasticity Rule				✓			✓		
SLA Management	✓		✓	✓	✓		✓		✓
Resource Monitoring				✓		✓			
Auditing & Reporting									

As per the feedback of 2nd reviewer, he has confirmed that the given detail of cloud interoperability reference architecture is sufficient but categorization and flow of modules are not appropriate Although as per our study, different researchers has given the different view in categorizations and this is not much important because components can be swapped as per understanding The expert has confirmed that proposed components are suitable and appropriate for cloud interoperability reference architecture

5.3 Concluding Remarks

This section presented the overall conclusion of the results based upon the survey results from cloud experts of proposed cloud interoperability reference architecture Also the research overview was given where the aim for the research was given The research aim was to propose an architecture that enables to eliminate the vendor lock in issue As per the feedback from cloud industry, the given detail in architecture is sufficient which is agreed by both reviewer On the other hand, the proposed components which were selected from thesis literature review and selected in our proposed architecture are suitable, appropriate and important So this was also agreed by both The categorization and flow of module has less importance as compare to detail and selection of components Components can be swapped based upon understanding of researchers We have applied the best practices and after analysis we have categorized them as core, general and supporting components

CHAPTER 6

Conclusion

6. CONCLUSION

The concluding portion debates on the summary of research described in this thesis the contributions of the thesis and probable directions for future research

6.1 Contribution of Research

In this thesis we have presented a Cloud Interoperability Reference Architecture that could allow the multiple cloud service providers to interoperate with each other The Standardized Reference Architecture comprises of three types of components namely the core components, supportive components and general components This work focuses on to provide maximum interoperability solutions to eradicate the vendor lock-in problem

We have presented a systematic review of approaches that address issues relating to vendor lock-in, portability and interoperability for cloud computing Our review has been conducted to identify, analyses and classify existing solutions In contrast with related work, our work
(i) extends the scope beyond one specific issue, addressing vendor lock-in, portability and interoperability, (ii) identifies the solutions according to the devised artefact, (iii) provides an overview of the research area, (iv) highlights areas in which solutions have not been proposed. in addition, we identified the gaps of research regardless the interest in one specific solution, (v) identifies the relations amongst solutions, and (vi) summarizes the most relevant work done so far addressing vendor lock-in, portability, and interoperability in cloud computing

6.2 Limitation and Future Work

Limitation

It was possible to synthesize the roadmap by comparing the results obtained from the survey by those obtained from other researches and the findings from the literature review The reason of low number of respondent was it was conducted in month of November which is majorly the mid exams season of students and for professionals its end of year closings Therefore the questionnaire which was distributed over the email was not successful and there were no responses from most of the industry professionals

Future Work

The possible direction for the future work of this research leads to find the differences between proposed Cloud Interoperability Reference Architecture (CIRA) and the existing interoperability libraries In addition, suggest some modifications to those interoperability libraries to make them aligned to Cloud Interoperability Reference Architecture (CIRA) This will guide the cloud service providers to interoperate with each other seamlessly

References

References

[1] Armbrust, M , Fox, A , Griffith, R , Joseph, A D , Katz, R , Konwinski, A , & Zaharia M (2010) A view of cloud computing *Communications of the ACM*, *53*(4), 50-58

[2] Voorsluys, W , Broberg, J , & Buyya, R (2011) Introduction to cloud computing *Cloud computing Principles and paradigms*, 1-44

[3] Kurze, T , Klems, M , Bermbach, D , Lenk, A , Tai, S , & Kunze, M (2011) Cloud federation *CLOUD COMPUTING, 2011*, 32-38

[4] Teckelmann, R , Reich, C , & Sulistio, A (2011, November) Mapping of cloud standards to the taxonomy of interoperability in IaaS In *Cloud Computing Technology and Science (CloudCom). 2011 IEEE Third International Conference on* (pp 522-526) IEEE

[5] Amazon Web Services, http //aws amazon com, Dec 2010

[6] Rackspace, http //www rackspacecloud com/index php, Dec 2010

[7] Flexiscale, http //www flexiant com/products/flexiscale/, Dec 2010

[8] Prodan, R , & Ostermann, S (2009, October) A survey and taxonomy of infrastructure as a service and web hosting cloud providers In *2009 10th IEEE/ACM International Conference on Grid Computing* (pp 17-25) IEEE

[9] Rochwerger, B , Breitgand, D , Levy, E , Galis, A , Nagin, K , Llorente, I M , & Ben-Yehuda, M (2009) The reservoir model and architecture for open federated cloud computing *IBM Journal of Research and Development*, *53*(4), 4-1

[10] Gao, J , Xiao, Y , Liu, J , Liang, W , & Chen, C P (2012) A survey of communication/networking in Smart Grids *Future Generation Computer Systems*, *28*(2) 391-404

[11] Mell, P , & Grance, T (2010) The NIST definition of cloud computing *Communications of the ACM*, *53*(6), 50

[12] Loutas, N , Peristeras, V , Bouras, T , Kamateri, E , Zeginis, D , & Tarabanis, K (2010 November) Towards a reference architecture for semantically interoperable clouds In *Cloud Computing Technology and Science (CloudCom), 2010 IEEE Second International Conference on* (pp 143-150) IEEE

[13] Amazon EC2 and S3, http //aws amazon com

[14] Google App Engine
http //code google com/intl/en/appengine/docs/whatisgoogleappengine html

[15] Mell, P , & Grance, T (2011) The NIST definition of cloud computing (draft) *NIST special publication, 800*(145), 7

[16] Fox, A , Griffith, R , Joseph, A , Katz, R , Konwinski, A , Lee, G , & Stoica, I (2009) Above the clouds A Berkeley view of cloud computing *Dept Electrical Eng and Comput Sciences, University of California, Berkeley, Rep UCB/EECS, 28*(13), 2009

[17] ComPUtING, C (2011) Cloud computing privacy concerns on our doorstep *Communications of the ACM, 54*(1), 36-38

[18] Rimal, B P , Choi, E , & Lumb, I (2009) A taxonomy and survey of cloud computing systems *INC, IMS and IDC*, 44-51

[19] Sheth, A , & Ranabahu, A (2010) Semantic modeling for cloud computing, part 2 *IEEE Internet Computing, 14*(4), 81-84

[20] Elmroth, E , Hernandez, F , & Tordsson, J (2010) Three fundamental dimensions of scientific workflow interoperability Model of computation, language, and execution environment *Future Generation Computer Systems, 26*(2), 245-256

[21] Burguillo, J C (2010) Using game theory and competition-based learning to stimulate student motivation and performance *Computers & Education, 55*(2), 566-575

[22] Maali, F , Cyganiak, R , & Peristeras, V (2010, August) Enabling interoperability of government data catalogues In *International Conference on Electronic Government* (pp 339-350) Springer Berlin Heidelberg

[23] Urquhart, J (2009) Exploring cloud interoperability, part 2 *The Wisdom of Clouds DOI http //news cnet Com/8301-19413_3-10231290-240 html*

[24] Ranabahu, A , & Sheth, A (2010, November) Semantics centric solutions for application and data portability in cloud computing In *Cloud Computing Technology and Science (CloudCom), 2010 IEEE Second International Conference on* (pp 234-241) IEEE

[25] Foster, I , Zhao, Y , Raicu, I , & Lu, S (2008, November) Cloud computing and grid computing 360-degree compared In *2008 Grid Computing Environments Workshop* (pp 1-10)

[26] Zhou, M , Zhang, R , Xie, W , Qian, W , & Zhou, A (2010, November) Security and privacy in cloud computing A survey In *Semantics Knowledge and Grid (SKG), 2010 Sixth International Conference on* (pp 105-112) IEEE

[27] H Takabi, J Joshi, and G Ahn, "Security and Privacy Challenges in Cloud Computing Environments," *Security & Privacy*, IEEE, Vol 8, No 6, Nov/Dec 2010, pp 24-31

[28] B Michael and G Dinolt, "Establishing Trust in Cloud Computing, *IAnewsletter*, Vol 13, No 2, pp 4-8, 2010

[29] Petcu, D, Di Martino, B, Venticinque, S, Rak, M, Máhr, T, Lopez, G E, & Stankovski, V (2013) Experiences in building a mOSAIC of clouds *Journal of Cloud Computing Advances, Systems and Applications*, *2*(1), 1

[30] Marosi, A, Kecskemeti, G, Kertesz, A, & Kacsuk, P (2011) FCM an architecture for integrating IaaS cloud systems

[31] Carlini, E, Coppola, M, Dazzi, P, Ricci, L, & Righetti, G (2011 August) Cloud federations in contrail In *European Conference on Parallel Processing* (pp 159-168) Springer Berlin Heidelberg

[32] Lloret, J, Garcia, M, Tomas, J, & Rodrigues, J J (2014) Architecture and protocol for intercloud communication *Information Sciences*, *258*, 434-451

[33] Vitek, A J, & Morris, M N (2012) Service oriented cloud computing architectures In *UMM CSci Senior Seminar Conference. Available online https //wiki umn edu/[07 11 2013]*

[34] Yangui, S, Marshall, I J, Laisne, J P, & Tata, S (2014) CompatibleOne The open source cloud broker *Journal of Grid Computing*, *12*(1), 93-109

[35] Hill, Z, & Humphrey, M (2010, November) CSAL A cloud storage abstraction layer to enable portable cloud applications In *Cloud Computing Technology and Science (CloudCom), 2010 IEEE Second International Conference on* (pp 504-511) IEEE

[36] Metsch, T, & Edmonds, A (2010, November) Open cloud computing interface-infrastructure In *Standards Track, no GFD-R in The Open Grid Forum Document Series Open Cloud Computing Interface (OCCI) Working Group, Muncie (IN)*

[37] McKendrick, J (2010) Does Platform as a Service have interoperability issues? *Cloud Commons* In DOI http //www zdnet com/blog/service-oriented/doesplatform- as-a-service-have-interoperability-issues/4890, 2010

[38] Economist, "Battle of the Clouds," in DOI http //www economist com/opinion/displaystory cfm?story_id=1464439 3, 2009

[39] D Catteddu and G Hogben, "Cloud Computing-Benefits, risks and recommendations for information security," in DOI http //www enisa europa eu/

[40] Demchenko, Y , Makkes, M X , Strijkers, R , & de Laat, C (2012, December) Intercloud architecture for interoperability and integration In *Cloud Computing Technology and Science (CloudCom), 2012 IEEE 4th International Conference on* (pp 666-674) IEEE

[41] Di Martino, B , Petcu, D , Cossu, R , Goncalves, P , Mahr, T , & Loichate, M (2010, August) Building a mosaic of clouds In *European Conference on Parallel Processing* (pp 571-578) Springer Berlin Heidelberg

[42] Ahmed, M , Chowdhury, A. S M , Ahmed, M , Rafee, M H (2012). An advanced survey on cloud computing and state-of- the-art research issues ' International Journal of Computer Science Issues (IJCSI) 9 (1)

[43] Kitchenham, B (2004) Procedures for performing systematic reviews *Keele, UK Keele University, 33*(2004), 1-26

[44] Kuhlthau, C C (1991) Inside the search process Information seeking from the user's perspective *Journal of the American society for information science, 42*(5), 361

[45] Feng, D G , Zhang, M , Zhang, Y , & Xu, Z (2011) Study on cloud computing security *Journal of software, 22*(1), 71-83

[46] Randles, M , Lamb, D , & Taleb-Bendiab, A (2010, April) A comparative study into distributed load balancing algorithms for cloud computing In *Advanced Information Networking and Applications Workshops (WAINA), 2010 IEEE 24th International Conference on* (pp 551-556) IEEE

[47] Vaquero, L M , Rodero-Merino, L , & Buyya, R (2011) Dynamically scaling applications in the cloud *ACM SIGCOMM Computer Communication Review, 41*(1), 45-52

[48] Ren, K , Wang, C , & Wang, Q (2012) Security challenges for the public cloud *IEEE Internet Computing, 16*(1), 69

[49] Khorshed, M T , Ali, A S , & Wasimi, S A (2012) A survey on gaps, threat remediation challenges and some thoughts for proactive attack detection in cloud computing *Future Generation computer systems, 28*(6), 833-851

[50] Grobauer, B , & Schreck, T (2010, October) Towards incident handling in the cloud challenges and approaches In *Proceedings of the 2010 ACM workshop on Cloud computing security workshop* (pp 77-86) ACM

[51] Xiao, Z., & Xiao, Y (2013) Security and privacy in cloud computing *IEEE Communications Surveys & Tutorials*, *15*(2), 843-859

[52] Chen, D., & Zhao, H (2012, March) Data security and privacy protection issues in cloud computing In *Computer Science and Electronics Engineering (ICCSEE), 2012 International Conference on* (Vol 1, pp 647-651) IEEE

[53] Benson, T, Akella, A, Shaikh, A, & Sahu, S (2011, October) CloudNaaS a cloud networking platform for enterprise applications In *Proceedings of the 2nd ACM Symposium on Cloud Computing* (p 8) ACM

[54] Galante, G, & de Bona, L C E (2012, November) A survey on cloud computing elasticity In *Utility and Cloud Computing (UCC), 2012 IEEE Fifth International Conference on* (pp 263-270) IEEE

[55] Rodero-Merino, L, Vaquero, L M, Gil, V, Galán, F, Fontan J, Montero, R S & Llorente, I M (2010) From infrastructure delivery to service management in clouds *Future Generation Computer Systems*, *26*(8), 1226-1240

[56] Tang, K, Zhang, J M, & Jiang, Z B (2010, November) Framework for SaaS management platform In *e-Business Engineering (ICEBE) 2010 IEEE 7th International Conference on* (pp 345-350) IEEE

Appendix

Questionnaire made for evaluation sent to two cloud experts and their answers are attached in this section

Questionnaire 1

After reviewing the cloud interoperability reference architecture, is the given information detailed?	Yes	No
Is the proposed categorization in cloud interoperability reference architecture and flow of modules is appropriate?	Yes	No
Please see the attached proposed components and existing components in table 3 4 and 5 Are these proposed components appropriate and suitable for cloud interoperability reference architecture as compared to existing components?	Yes	No

Table 3 Core Components of Interoperability Reference Architecture

Reference Architecture	Contrail	FCM	RASIC	ICAF	SOCCA	CO	mOSAIC	CSAI	OCC
Rule Engine									
Portability Rule DB							✓		
Interoperability Rule DB									
Environmental Conformance Rule									
Interoperability Manager			✓						
Billing						✓			✓
Metering							✓		
Peer Service Ontology									
Peer Resource Monitoring									
Cross Auditing and Reporting									
Service Monitoring	✓		✓	✓			✓		
Incident Reporting									
Communication Libraries/ API	✓		✓	✓		✓	✓	✓	✓

Table 4 Support Components of Interoperability Reference Architecture

Reference Architecture	Contrail	FCM	RASIC	ICAF	SOCCA	CO	mOSAIC	CSAI	OCCI
Peer Inventory List							✓		
Peer Pricing and Rating									
Peer Elasticity Rule									
Customer Pricing & Rating									
Inventory Publication									
Resource Change									
Identity Management	✓			✓		✓			
Authentication	✓					✓			✓
Authorization	✓					✓			✓
Single Sign On	✓								✓
Pricing									
Peer SLA							✓		
Customer SLA							✓		

Table 5 General Components of Interoperability Reference Architecture

Reference Architecture	Contrail	FCM	RASIC	ICAF	SOCCA	CO	mOSAIC	CSAI	OCCI
Elasticity Rule				✓			✓		[illegible]
SLA Management	✓		✓	✓			✓		✓
Resource Monitoring				✓				✓	
Auditing & Reporting		✓							

Questionnaire 2

After reviewing the cloud interoperability reference architecture, is the given information detailed?	Yes	No
Is the proposed categorization in cloud interoperability reference architecture and flow of modules is appropriate?	Yes	No
Please see the attached proposed components and existing components in table 6, 7 and 8 Are these proposed components appropriate and suitable for cloud interoperability reference architecture as compared to existing components?	Yes	No

Table 6 Core Components of Interoperability Reference Architecture

Reference Architecture	Contrail	FCM	RASIC	ICAF	SOCCA	CO	mOSAIC	CSAL	OCC
Rule Engine									
Portability Rule DB							✓		
Interoperability Rule DB									
Environmental Conformance Rule									
Interoperability Manager			✓						
Billing						✓			✓
Metering							✓		
Peer Service Ontology			✓		✓	✓	✓		
Peer Resource Monitoring				✓					
Cross Auditing and Reporting						✓			
Service Monitoring	✓		✓	✓	✓		✓		
Incident Reporting									
Communication Libraries/API	✓		✓	✓		✓	✓	✓	

Table 7 Support Components of Interoperability Reference Architecture

Reference Architecture	Contrail	FCM	RASIC	ICAF	SOCCA	CO	mOSAIC	CSAL	OCC
Peer Inventory List							✓		
Peer Pricing and Rating									
Peer Elasticity Rule									
Customer Pricing & Rating									
Inventory Publication									
Resource Change									
Identity Management	✓		✓	✓		✓			
Authentication	✓		✓			✓			
Authorization	✓		✓			✓			
Single Sign On	✓								
Pricing									
Peer SLA							✓		
Customer SLA							✓		

Table 8 General Components of Interoperability Reference Architecture

Reference Architecture	Contrail	FCM	RASIC	ICAF	SOCCA	CO	mOSAIC	CSAL	CC
Elasticity Rule				✓			✓		
SLA Management	✓		✓	✓	✓		✓		✓
Resource Monitoring				✓		✓			
Auditing & Reporting									