EVALUATION OF POSSIBLE CLOUD COMPUTING STANDARDS

Magdalena Kostoska
Faculty of Computer Science and Engineering
Skopje, Macedonia

Marjan Gusev
Faculty of Computer Science and Engineering
Skopje, Macedonia

ABSTRACT

The general objective of this paper is explore and evaluate proposed standards for cloud computing. In this paper we analyse relevant proposed cloud specific standards, for each of them we evaluate their extensiveness, so as the possibility of acceptance.

I. INTRODUCTION

The need for standardization in the field of computer science has always been recognized. And yet, one of the most popular new discipline in this filed – the cloud computing is lacking standardization. Many attempts have been done and there is no visible result yet. A lot of groups work at standardization in different directions and areas at this moment and part of them offer draft documents as a result of their work. Each of these groups focuses on some layer in the cloud computing stack (such as IaaS, PaaS or SaaS) or tries to standardize some of the functions.

In section 2 we give overview of some of the cloud computing standards organizations.

In section 3 we give overview of some of the proposed draft cloud computing standards.

In section 4 we evaluate and compare the described standards.

II. CLOUD COMPUTING STANDARDS ORGANIZATIONS

The need for standardization in the field of computer science has always been recognized. And yet, one of the most popular new discipline in this filed – the cloud computing is lacking standardization. Many attempts have been done and there is no visible result yet. A lot of groups work at standardization in different directions and areas at this moment and part of them offer draft documents as a result of their work. Each of these groups focuses on some layer in the cloud computing stack (such as IaaS, PaaS or SaaS) or tries to standardize some of the functions.

In this paper we will analyse and evaluate some of the proposed draft cloud computing standards.

III. PROPOSED STANDARDS

In this section we will describe six proposed standards originated from different work groups.

A. Cloud Infrastructure Management Interface (CIMI)

The Cloud Infrastructure Management Interface (CIMI) is developed by the Distributed Management Task Force (DMTF), more precisely by the Cloud Management Working Group (CMWG). This standard defines logical model for management of resources at the Infrastructure as a Service layer of cloud computing. It defines model and uses Hyper Text Transfer Protocol (HTTP) Representational State Transfer (REST)-style protocol to manage the interaction between the providers and consumers. The messages that are formatted using either Java Script Object Notation (JSON) or the eXtensible Markup Language (XML). Its goal is to enable portability between cloud implementations that support this specification and to allow interoperability between a consumer and multiple providers. [1]

CIMI models the basic resources as machines, storage, and networks and its scope is one separately administered cloud. Each of the resources is represented by a set of key/value pairs that define the configuration, the operation or the relationship between resources. It uses various types for the pairs as boolean, dateTime, duration, integer etc. The resources are organized into catalogues:

- Cloud Entry Point - the entry point for finding resources and capabilities
- Machine Resources - compute infrastructure resources
- Volume Resources - storage infrastructure resources
- Network Resources - networking infrastructure resources
- System Resources - relationships between machines, volumes an networks
- Monitoring Resources - tracking the progress of operations, metering and monitoring the status of other resources

The model is self-describing, allows querying of own metadata and defines serialization of its entities in XML and JSON. [1]

Some of the scenarios in which this protocol can be used are: creating new machine, adding new volume to a machine, creating new machine from an existing volume and defining and using templates. [2]

B. Open Cloud Computing Interface (OCCI)

The Open Cloud Computing Interface (OCCI) is developed by Open Grid Forum (OGF). This protocol represents an API for all kinds of management tasks. The main goal of this standard is same as CIMI – to enable interoperability. OCCI represents a front-end service to a provider's internal management framework. Initially this API was intended "*for the development of interoperable tools for common tasks including deployment, autonomic scaling and monitoring*". The idea of this standard is to represent a unified management API for all providers. Figure 1 shows the placement of OCCI in a provider's infrastructure. [3]

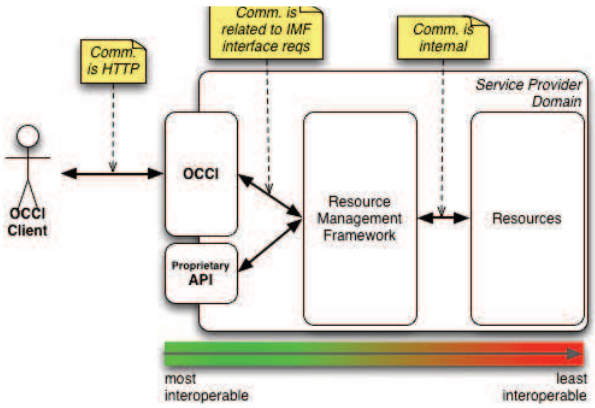

Figure 1: OCCI's place in a provider's architecture. [3]

The OCCI Core Model defines how the resources are represented and how they can be manipulated through an OCCI rendering implementation and can be represented by UML class diagram. A resource can be a virtual machine, a job in a job submission system, a user, etc.

This protocol also describes OCCI Infrastructure, which shows how an OCCI implementation can model and implement an IaaS API for the creation and management of resources. Additionally the core model can be interacted through RESTful OCCI API.

This protocol already offers implementations for OpenStack, OpenNebula, Eucalyptus and etc.

C. *Cloud Data Management Interface (CDMI)*

The Cloud Data Management Interface (CDMI) protocol is developed by Storage Networking Industry Association (SNIA) and now is an ISO standard (ISO/IEC 17826:2012). This protocol "*defines the functional interface that applications will use to create, retrieve, update and delete data elements from the Cloud*". [4] It provides management of Data storage as a Service (DaaS). CDMI may also be used for managing containers, domains, security access, and monitoring/billing information, etc. This standard also uses RESTful protocol (where possible) to interface the storage capabilities. It also uses metadata which enables managing large amount of data with different requirements. The goal of this standard is to enable interoperability and portability between cloud storage service providers.

CDMI standard defines a model for the interfaces that may be mapped to the various offerings. It provides data and control path for cloud storage. Figure 2 shows CDMI basic data flow.

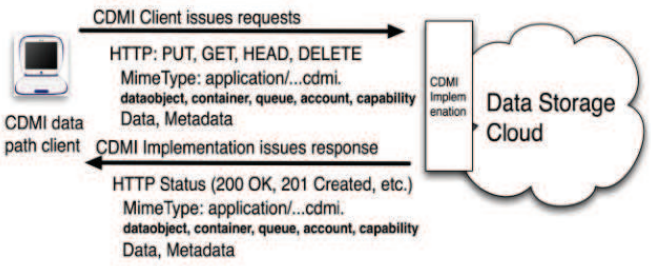

Figure 2: CDMI basic data flow. [5]

Applications (TOSCA)

The Topology and Orchestration Specification for Cloud Applications (TOSCA) standard is developed by OASIS in order to enable the interoperable description of application and infrastructure cloud services, the relationships between parts of the service, and the operational behaviour of these services [6].

This standard uses open standards to describe a service and how to manage it independently from the supplier who creates the service, as well as from any particular cloud provider (or its technology) that hosts the service. A definition of service is provided by a Service Template document. The TOSCA language introduces a grammar for describing service templates by means of Topology Templates and plans. TOSCA utilizes XML Schema 1.0 and WSDL 1.1. It also contains non-normative references to BPEL 2.0 (Web Services Business Process Execution Language Version 2.0, OASIS Standard, BPMN 2.0 (OMG Business Process Model and Notation Version 2.0), OVF (Open Virtualization Format Specification Version 1.1.0) and XPATH 1.0 (XML Path Language Version 1.0, W3C Recommendation).

The TOSCA specification uses TOSCA xml and xs namespace prefixes, but allows extensibility from other namespaces for attributes and elements, which do not oppose the semantics of the TOSCA namespace. The specification defines a metamodel for defining the structure of an IT service and its management. The Topology Template defines the structure of a service. Plans define the process models that are used to create, terminate and manage a service. The major elements defining a service are depicted in Figure 3.

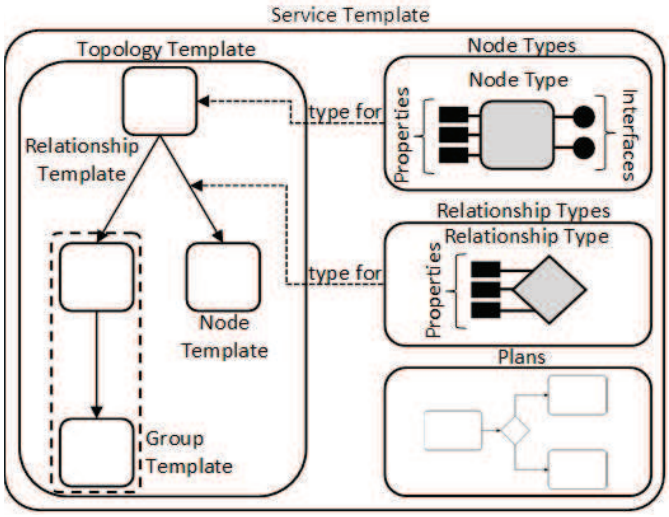

Figure 3: TOSCA Service Template. [6]

The specification supports the following major use cases:

- Services as Marketable Entities,
- Portability of Service Templates,
- Service Composition and
- Relation to Virtual Images.

E. *IEEE P2301 (Cloud Profiles)*

The goal of this standard is to develop Guide for Cloud Portability and Interoperability Profiles (CPIP). The guide is intended to "*advise cloud computing ecosystem participants*

based choices in areas such as application interfaces, portability interfaces, management interfaces, interoperability interfaces, file formats, and operation conventions". The guide groups these choices into multiple logical profiles, which are organized to address different cloud personalities. [7]

This guide enumerates Cloud Computing systems elements options, grouped in a logical fashion called "profiles," for such definitions of interfaces, formats, and conventions, from a variety of sources.

F. IEEE P2302 (Intercloud)

The Standard for Intercloud Interoperability and Federation (SIIF) defines "*topology, functions, and governance for cloud-to-cloud interoperability and federation*". Topological elements include clouds, roots, exchanges (which mediate governance between clouds), and gateways (which mediate data exchange between clouds). Functional elements include name spaces, presence, messaging, resource ontologies (including standardized units of measurement), and trust infrastructure. Governance elements include registration, geo-independence, trust anchor, and potentially compliance and audit. The standard does not address intra-cloud (within cloud) operation, as this is cloud implementation-specific, nor does it address proprietary hybrid-cloud implementations. [8]

IV. Comparison and Evaluation

For each of the described proposed standards we evaluate layer of CC stack and status. The comparison is described in table 1.

Table 1: Standards comparison

Standard	Layer	Status	Usage
CIMI	IaaS	Available documentation	Support by many companies, no available implementation yet
OCCI	IaaS	Available documentation	Availbale implementation in OpenStack, OpenNebula, Eucaliptus etc…
CDMI	DaaS	Available documentation	ISO standard
TOSCA	SaaS	Available documentation	No available implementation yet
CPIP	Misc.	In progress	No available implementation yet
SIIF	IaaS	In progress	No available implementation yet

The usage of cloud computer is increasing especially in the area of offering infrastructure and software in cloud as a Service (IaaS and SaaS). There are several approaches and research initiatives claiming progress in creation of interoperability in the area of IaaS.

However there is no evidence of progress for setting the interoperability in the area of SaaS. The logical steps lead to creation of general interoperability frameworks for each layer, starting from lowest, and then to expand them. At this moment it seems unlikely for any of proposed interoperability model to be adopted as a standard, since none of them is generally accepted.

Acknowledgment. This work was partially financed by the Faculty of Computer Science and Engineering at the "Ss.Cyril and Methodius" University.

References

[1] DMTF, "Cloud Infrastructure Management Interface (CIMI) Model and RESTful HTTP-based Protocol", September 2012, http://dmtf.org/sites/default/files/standards/documents/DSP0263_1.0.1.pdf.

[2] DMTF, "Cloud Infrastructure Management Interface (CIMI) Primer", August 2012, http://dmtf.org/sites/default/files/standards/documents/DSP2027_1.0.0.pdf.

[3] OGF, "OCCI Open Cloud Computing Interface". http://occi-wg.org/.

[4] SNIA, "Cloud Data Management Interface (CDMI)", June 2012, http://snia.org/sites/default/files/CDMI%20v1.0.2.pdf.

[5] SNIA, "Implementing, Serving, and Using Cloud Storage", October 2010, https://www.snia.org/sites/default/files/2010-10-WP-ImplementingServingandUsingTheCloud.pdf.

[6] OASIS, "Topology and orchestration specification for cloud applications version 1.0.", March 2013, http://docs.oasis-open.org/tosca/TOSCA/v1.0/cs01/TOSCA-v1.0-cs01.html.

[7] IEEE P2301 Working Group (Cloud Profiles), http://grouper.ieee.org/groups/2301/.

[8] IEEE P2302 Working Group (Intercloud), http://grouper.ieee.org/groups/2302/