On Supporting Service Selection for Collaborative Multi-Cloud Ecosystems in Community Networks

Amin M. Khan*, Felix Freitag*, Smrati Gupta†, Victor Muntès-Mulero†, Jacek Dominiak†, Peter Matthews†
* Department of Computer Architecture, Universitat Politècnica de Catalunya. Barcelona, Spain
{mkhan, felix}@ac.upc.edu
† CA Labs, CA Technologies. Barcelona, Spain
{smrati.gupta, victor.muntes, jacek.dominiak, peter.matthews}@ca.com

***Abstract*—Internet and communication technologies have lowered the costs for communities to collaborate, leading to new services and collectively built infrastructures like community networks. Community networks get formed when individuals and local organisations from a geographic area team up to create and run a community-owned IP network to satisfy the community's demand for ICT, such as facilitating Internet access and providing services of local interest. To address the limitation and enhance utility of community networks, we deploy collaborative clouds in community networks that allow interesting applications to be developed for serving local needs of communities. Such collaborative clouds employ resources contributed by the members of the community network for provisioning infrastructure and software services, and adapt to the specific social, economic and technical characteristics of the community networks. We need to support mechanisms that provide assistance in cloud service selection while taking into account different aspects pertaining to associated risks in community clouds, quality concerns of the users and cost limitations specifically in multi-clouds ecosystems. This paper proposes a risk-cost-quality based decision support system to assist the community cloud users to select the most appropriate cloud services meeting their needs. The proposed framework not only increases the ease of adoption of community clouds by providing assistance to users in cloud service selection, but also provides insights into the improvement of community clouds based on user behaviour.**

***Index Terms*—multi-cloud; decision support; collaborative systems; cloud computing; community cloud; community networks**

I. Introduction

Recent developments in information and communication technologies have reduced the barriers for communication, coordination and collaboration for communities, enabling infrastructures like community wireless mesh networks [1] that are based on a cooperative model. Using off-the-shelf network equipment and open unlicensed wireless spectrum, volunteers teamed up to invest, create and run wireless networks in their local communities as an open telecommunication infrastructure based on self-service and self-management by the users. These community networks have proved quite successful, for example Guifi.net[1] provides wireless and optical fibre based broadband access to more than 20,000 users. Current community networks use mainly wireless technology to interconnect nodes, though with the commoditization of optical fibre, some have also started providing broadband services combining both technologies.

Community networks are a successful case of resource sharing among a collective, where resources shared are not only the networking hardware but also the time, effort and knowledge contributed by its members that are required for maintaining the network. Resource sharing in community networks from the equipment perspective refers in practice to the sharing of the nodes' bandwidth. This sharing enables the traffic from other nodes to be routed over the nodes of different node owners, allowing community networks to successfully operate as IP networks. Despite achieving sharing of bandwidth, community networks have not been able to extend this sharing to other computing resources like storage, which is now common practice in today's Internet through cloud computing. There are not many applications and services used by members of community networks that take advantage of resources available within community networks. To overcome this limitation, we envision a specific kind of a community cloud [2] in which sharing of computing resources is from within community networks, using the application models of cloud computing in general [3], [4]. When members of community network can share and trade resources based on a collaborative cloud computing model, they can provide their excess capacity to others as the demand fluctuates and in return can take advantage of services and applications that were not possible earlier due to the limited resources.

The successful functioning of the community cloud relies on the active participation of the community members that in turn is highly dependent on the level of satisfaction experienced by the users of the community cloud services. The user experience can be maximised if the offered cloud service accurately matches the user requirements. A decision support system (DSS) is therefore required that can recommend the users to select the accurate cloud services from the services offered by the network satisfying their requirements. Such DSS is particularly crucial in multi-cloud ecosystem where cloud service selection becomes exceedingly difficult for users. Another crucial challenge in cloud service selection is translating the user requirements from a highly pragmatic perspective into technical properties that should be possessed by the cloud service. The DSS should take into account all these challenges and provide the user a recommendation system to assist the

[1] http://guifi.net

cloud service selection.

The concept of DSS essentially is implementation of a multi-criteria decision making problem (MCDM) introduced in [5], which has been developed as DSS for different scenarios since then. In particular, for cloud service selection, DSS implementation has been surveyed in detail in [6]. In multi-cloud environments however, the DSS should take into account multiple factors that are specific to the characteristics of such ecosystems. For instance, DSS adopting three dimensional approach of satisfying risk, cost and quality based aspects in such multi-cloud environments are particularly effective. In terms of cloud service selection, a novel DSS proposed in [7] is based on such approach. This DSS implements a risk-driven methodology to translate the user requirements into technical characteristics of the cloud services, and recommends the most accurate service selection that minimises the risk and maximises the quality at an appropriate cost.

In this paper, we present a DSS framework that allows the users of the community cloud to translate their needs into technical characteristics of the cloud services and recommends them a set of cloud services meeting their requirements. In particular, the novelty of this DSS over the one proposed in [7] lies in accounting for multi-cloud environments unique to community clouds.

The contributions of this paper are the following:

1) Presenting the use cases of storage applications in collaborative multi-cloud environment.
2) Extending the idea of DSS to collaborative multi-cloud environments, which differ from public and enterprise cloud systems because consumers can also act as providers.

The rest of the paper is organised as follows. Section II presents the related work. Section III discusses the multi-clouds in the community networks, and presents the use cases for service selection in collaborative multi-cloud systems. Section IV introduces the key characteristics and implementation of DSS, and how it applies to community network clouds. Section V explains the case study with the deployment of storage services in community cloud. Section VI concludes and discusses future research directions.

II. Related Work

The idea of collaboratively built community clouds follows on from earlier distributed volunteer computing platforms, like SETI@Home [8], PlanetLab [9] and Seattle [10], and, in general, extends the peer-to-peer paradigm [11]. There are only a few research proposals for community cloud computing [2]. Cloud@Home [12] project aims to harvest resources from the community for meeting the peaks in demand, working with public, private and hybrid clouds to form cloud federations. Social cloud computing [13] takes advantage of the trust relationships between members of the online social networks to motivate the sharing of infrastructure resources, and explores bidirectional preference-based resource allocation. The Clouds@home [14] project focuses on providing guaranteed performance and ensuring quality of service even when using volatile volunteered resources connected by Internet. Other collaborative community-oriented projects include FreedomBox[2] and MeshNet[3] which focus on ensuring privacy, while FI-WARE CoudEdge[4] and ownCloud[5] let cloud applications consume resources locally.

In multi-cloud environments, it is essential to provide tools that guide multi-cloud application architects to choose the services providing the necessary quality and ensuring acceptable level of risk and concurrently satisfying the cost constraints. Previous work has focused on describing quality aspects and metrics to measure the suitability of a cloud service from a multi-dimensional perspective. Taha et al. [15] have looked into comparing cloud service providers from security point of view. PaaSage project [16] has explored model-driven deployment of application in multi-cloud systems, and aims to use social media as a feedback mechanism to evaluate the matching of requirements and offerings of cloud service providers in terms of quality and customer satisfaction. Service Measurement Index (SMI) [17] is a framework designed to allow for quick and reliable comparison of IT business services. SMI is a standardisation effort from the Cloud Services Measurement Index Consortium (CSMIC) consisting of academic and industry organisations. SMI uses a series of characteristics and measures to create a common means to compare different services from different suppliers on basis of characteristics like usability, performance, agility, assurance, usability, financial, security and privacy. Each of these characteristics has a number of measures that can be used to evaluate the risk in using a service. However, they do not explicitly analyse these aspects in a multi-cloud context. The DSS proposed in [7] is inspired by the principles of SMI. It takes into account the multi-cloud ecosystems in three dimensional approach. Adaptation of such DSS for community cloud is critical to a holistic development of DSS technology.

III. Collaborative Multi-clouds in Community Networks

A community network is managed and owned by the community, where nodes are managed independently by their owners. Community networks seem to be rather successful and there are several large community networks in Europe, having from 500 to 20,000 nodes, such as FunkFeuer[6], AWMN[7], Ninux[8], Guifi.net and Freifunk[9] among many others. A community network like Guifi.net is organised into zones where a zone can be a village, a small city, a region, or districts of a larger city. Mostly, the detailed technical support for the members is only available within the community of their zone, so we identify a zone to have the highest social strength within the

[2]http://freedomboxfoundation.org
[3]http://projectmeshnet.org
[4]http://catalogue.fi-ware.eu/enablers/cloud-edge
[5]http://owncloud.org
[6]http://www.funkfeuer.at
[7]http://www.awmn.gr
[8]http://ninux.org
[9]http://www.freifunk.net

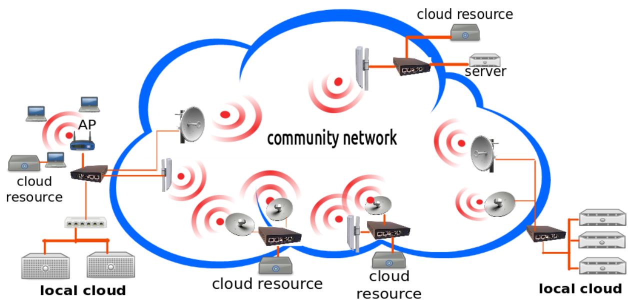


Figure 1. Nodes in a community network with cloud resources

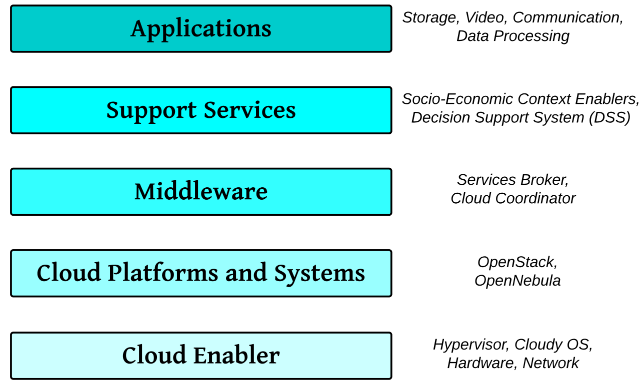


Figure 2. Architecture of the community cloud management system

community network. The computer machines or nodes in a community network vary widely in their capacity, function and capability, as illustrated in Figure 1. Some hardware is used as super nodes (SNs) that have multiple wireless links and connect with other super nodes to form the backbone of the community network, and are usually intended to be stable with permanent connectivity. Others act just as ordinary nodes (ON) and are only connected to the access point of a super node.

From the node types shown in Figure 1, it can be seen that principally the hardware for computation and storage is already available in community networks, consisting of some servers attached to the networking nodes. No cloud services, however, are yet deployed in community networks to use this hardware as a cloud, leaving the community network services significantly behind the current standard of the Internet. Our vision is that some community wireless routers will have cloud resources attached, building the infrastructure for a community cloud formed by several cloud resources attached to the community nodes. We note that client nodes could principally also contribute cloud resources.

The cohesive nature of zones gives rise to the scenario of the local community cloud, interpreting the characteristics of the social networks existing within zones and the topology of the community network. In this scenario, some SNs with their better connectivity and high availability are responsible for the management of a set of attached ONs that are contributing cloud resources.

Local community cloud can provide services for the users within its zone. Multiple SNs from different zones in a community network, however, can participate together in a federated community cloud to support greater functionality and higher capacity. The ONs in a given zone are directly managed by the SN in that zone but they can also consume resources from other zones, given that there is a coordination mechanism among zones in place.

Enabling collaborative cloud services in the community networks will require deploying a cloud management platform tailored to community networks on the nodes attached to the network. There are a few cloud management systems available to manage public and private clouds, notably OpenStack[10] and OpenNebula[11] among others. Such cloud management systems can be customised for community networks by extending the existing functionality to address the particular conditions of community networks. For example, incentive mechanisms inspired by the social nature of community networks [18] can be built into resource regulation component to encourage users to contribute resources [19], [20]. The architecture for the cloud management system that we propose for community networks consists of multiple layers, as shown in Figure 2 [3]. Here a DSS as part of the support services can prove pivotal in ensuring quality of experience for the members of the community networks, facilitating adoption of community cloud services.

A. *Use Cases for Cloud Services Selection*

We identify in this section the use cases that reflect how the collaborative cloud services can be used in a multi-cloud environment, and how DSS can assist different actors – developers, providers, and end-users – in service selection. We focus mainly on storage service here [21], where factors like vendor lock-in and migration costs can be considered when comparing different providers in the community cloud. The use cases are summarized in Table I, where we are differentiating between public or enterprise clouds and the community clouds.

1) Storage for Video Streaming: Consider the developers of video streaming and video-on-demand services that are looking for a storage service at the back-end to store all the multimedia content. Their requirements include:

- Abundant storage
- Guarantees about data persistence
- Replication
- High bandwidth to the back-end storage

The features that are not critical in this context are:

- Reliable backups
- Data privacy and security

[10] http://www.openstack.org
[11] http://www.opennebula.org

The preferable providers in this case will be the ones that can support more resourceful machines with good interconnectivity.

2) Distributed Storage for Content Distribution: Consider distributors of video streaming and video-on-demand services that are looking for a caching service that can improve the performance and reduce the latency when delivering multimedia content to the end users, hence work as content distribution network (CDN). Their requirements include:

- Good storage
- Locality, as nodes should be closer to the end users
- Large number of nodes geographically well distributed
- Good bandwidth for local connections

The features that are not critical in this context are:

- Abundant storage
- High bandwidth for the overall network

The preferable providers in this case will be the ones that can support more machines even if they are less resourceful, but distributed well in the community.

3) Secure Personal Data Storage: Consider users who want a secure and private cloud storage solution. This includes backup as well as data synchronisation services. The backups at the provider's end are important but may not be very critical, as the end users also have the copies of the data. Their requirements include:

- Reasonable amount of storage
- Guaranteed privacy
- Guaranteed security
- Reliable backups
- End-to-end encryption

The features that are not critical in this context are:

- Abundant storage
- High bandwidth
- Network latency
- Processing power

The storage service from community clouds with strong end-to-end encryption can be a good candidate for this use case.

4) Reliable Backup Service: Consider the end users who want guaranteed backup for their data with the assurance that the data is never lost. The privacy and security is important but so is the fact that the data is available forever. Their requirements include:

- Abundant storage
- Guaranteed persistence
- High level of replication
- Reliable backups

The features that are not critical in this context are:

- High bandwidth
- Network latency
- Processing power

The community clouds are well suited for this, provided they can offer a lot of machines that are also geographically well distributed.

5) Serving Public Data Sets: Scientific research community works with huge public data sets that need to be stored and distributed to different geographical locations. Some examples include data feeds from Twitter, sensor data from high energy physics experiments, and output data from simulation experiments run on supercomputers. The requirements include:

- Abundant storage
- High bandwidth
- Nodes are geographically well distributed

The features that are not critical in this context are:

- Privacy
- Security

The community clouds coupled with data centres and clusters at the universities and research labs can be used for this use case.

6) Storage for Photo Sharing: Consider developers of a public photo sharing service that are looking for a storage service at the back-end to store all the content. Their requirements include:

- Abundant storage
- Guarantees about persistence
- Replication
- Reliable backups

The features that are not critical in this context are:

- Processing power
- High bandwidth
- Network latency

The community clouds can be suitable if they can provide sufficient storage in aggregate.

7) Process-Intensive Batch Jobs: Consider the developers of cloud services that need to run process-intensive batch jobs, and need access to the extra computing power. Alternatively, consider a computer game development enterprise in the community that needs to render 3D imagery. The previous use cases dealt with storage resources, but this one focuses on computation facilities. Their requirements include:

- Processing power

The features that are not critical in this context are:

- Abundant storage
- Network latency

The preferable providers in this case will be the ones with more resourceful machines.

IV. Multi-Clouds Decision Support System

DSS is based on the solution to an MCDM problem, and there are many ways to solve this like analytical hierarchy process, condition based optimization, etc. [5]. The multi-cloud decision support system introduced in [7] is in the context of selection of cloud services from different independent cloud service providers based on risk analysis methodology. In this paper, we propose to adapt this DSS as per the requirements of the community clouds elucidated in the previous section.

Table I
USE CASES FOR SERVICE SELECTION IN COLLABORATIVE MULTI-CLOUDS

	Use Cases	Public Clouds	Community Clouds	Primary Requirements
1	Storage for Video Streaming	✓		Storage, Bandwidth
2	Distributed Storage for CDNs		✓	Locality, Latency
3	Secure Personal Data Storage		✓	Privacy, Security
4	Reliable Backup Service		✓	Persistence, Storage
5	Serving Public Datasets	✓	✓	Storage
6	Storage for Photo Sharing		✓	Storage
7	Process-Intensive Batch Jobs	✓		Processing power

A. *Key Characteristics*

Our proposed DSS is based on a number of characteristics that allows it to be agile enough to accommodate the crucial requirements derived from the problem of service selection in community clouds. We now describe these key characteristics. Firstly, the DSS framework allows the specification of the pragmatic needs of the users from different perspectives by allowing them to specify the assets that they intend to protect. These assets could be business oriented or technical oriented assets. Hence, the community cloud users from varied background can specify their main concerns with the use of the cloud services. Secondly, in the next step, the DSS using a pre-defined background mapping produces the list of risks associated with the assets and provides relevant treatments for these risks. The users are allowed to choose the risks and treatments. The set of treatments eventually form the necessary technical and non-technical features that the desired cloud services should have. Hence the proposed DSS provides a risk-analysis based methodology to translate user needs into desired characteristics of cloud services. Thirdly, each cloud service is scored from 1 to 10, inspired from SMI scores, in order to estimate the desired level of capabilities it possesses to mitigate a particular risk. These scores are assigned by careful analysis and are heuristically improved with repeated use of DSS. Fourthly, gathering the data from the service providers forms an integral part of the proposed framework and not a separate module that allows advanced implementation in complex environments. Fifthly, the multi-cloud environment is particularly accommodated in this framework by allowing the prevention of specific risks associated with the multi-cloud systems in community clouds. For example, the problem of obtaining the services from the same provider in community cloud, translated otherwise as vendor lock-in, is tackled by discarding such recommendations. The ease of migration from one provider to another is also considered while providing a final score and ranking to each of the recommendations provided by the DSS.

B. *Implementation of DSS*

We have developed a prototype DSS for supporting service selection in collaborative manner for multi-cloud applications. The prototype uses distributed graph database ArangoDB[12] as a persistent layer data store with graph exploration capabilities and AngularJS[13] based front-end which provides wizard based approach to identify best match of selection criteria. The prototype supports saving and sharing sessions capability in order to allow participation from multiple users in the definition of the requirements. In Figure 3, we have shown the generic MODAClouds DSS for cloud service selection that provides recommendations of the cloud providers as proposed in [7]. The final screen provides recommendation in multi-cloud environment for the possible cloud service providers matching the requirements of the user along with a score indicating how close a recommendation is to the desired properties provided by the user. The score is calculated based on what percentage of risks that were specified by the users have been mitigated.

[12]http://www.arangodb.com

C. *Service Selection with DSS*

For evaluating the use cases for community cloud services, we need to identify both the low-level and high-level metrics relevant for the community cloud, and take into account the data provided by the service providers in the community cloud testbed. Low-level data can be collected from the monitoring systems in testbed, and for high-level data we can survey the providers and users of the services. Some useful metrics particular to the collaborative nature of community clouds include providers' reputation among the users, users' as well as providers' geographical location, and mutual trust and standing among the providers and users based on the social status and cumulative history of past transactions. Considering the different applications already deployed in community cloud [21], we continue collecting relevant data from service providers.

V. CASE STUDY: COMMUNITY NETWORK CLOUDS

We explain in this section our current work in setting up a prototype cloud infrastructure in Guifi.net and AWMN community networks, as well as research labs at KTH Royal Institute of Technology, Swedish Institute of Computer Science (SICS), Abdus Salam International Centre for Theoretical Physics (ICTP) and Polytechnic University of Catalonia (UPC).

A. *Cloud infrastructure*

For having a realistic community network setting for the collaborative cloud services, we have used Community-Lab[14]

[13]http://angularjs.org
[14]http://community-lab.net

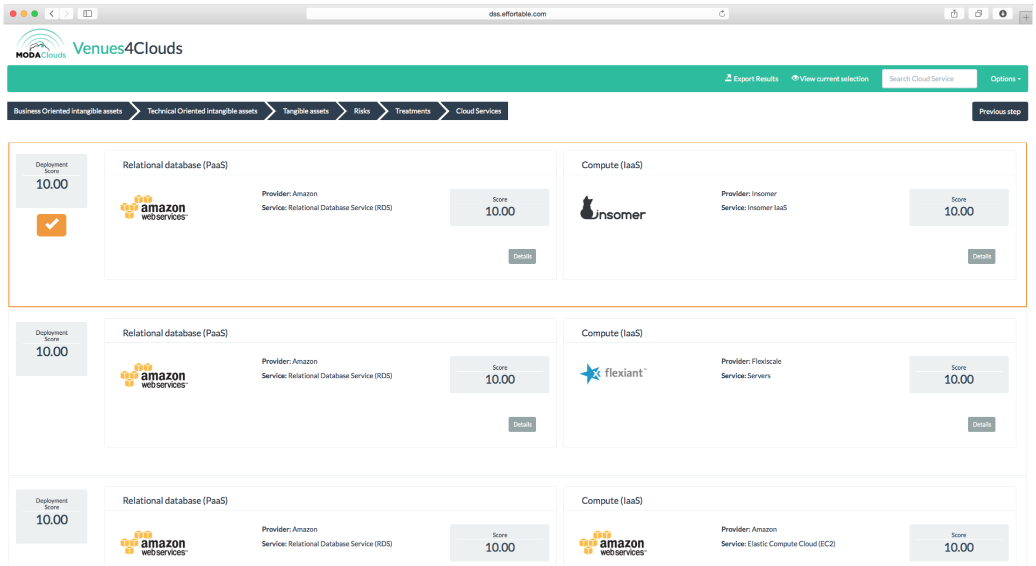


Figure 3. Prototype DSS for supporting service selection in collaborative multi-clouds

testbed for setting up our community cloud infrastructure. Community-Lab is a distributed infrastructure developed by the CONFINE project [1], where researchers can deploy experimental services on several nodes deployed within federated community networks. Within these nodes we deploy Cloudy[15] [22], a Debian based distribution, which comes pre-installed with some of the collaborative distributed applications, like Tahoe-LAFS[16], ownCloud[17], XtreemFS[18], Syncthing[19], etc. Figure 4 shows the Cloudy interface for viewing the list of Tahoe-LAFS service providers in the community cloud.

The primary configuration for our application deployments consists of nodes from the two community networks, Guifi.net in Spain and AWMN in Greece, which are connected on the IP layer, enabling network federation. This implies that some part of the distributed applications are in fact spread over nodes in Guifi.net, while the other components are hosted on the nodes belonging to AMWN. The nodes of our experiments are the real nodes from both the community networks, and they are connected to other actively used nodes within the community network through wireless IEEE 802.11 a/b/n connections. The hardware of most of these Community-Lab nodes consists of Jetway devices that are equipped with an Intel Atom N2600 CPU, 4 GB of RAM and 120 GB SSD. There are also some nodes in research labs consisting of machines with 4x Intel Core i7-3770 3.40 GHz CPU with 16 GB RAM and 1 TB hard disk. This provides heterogeneity in terms of storage space and processing power.

B. Community Cloud Services

We present here a brief overview of the different collaborative cloud services that we have deployed in our community cloud testbed based on Community-Lab.

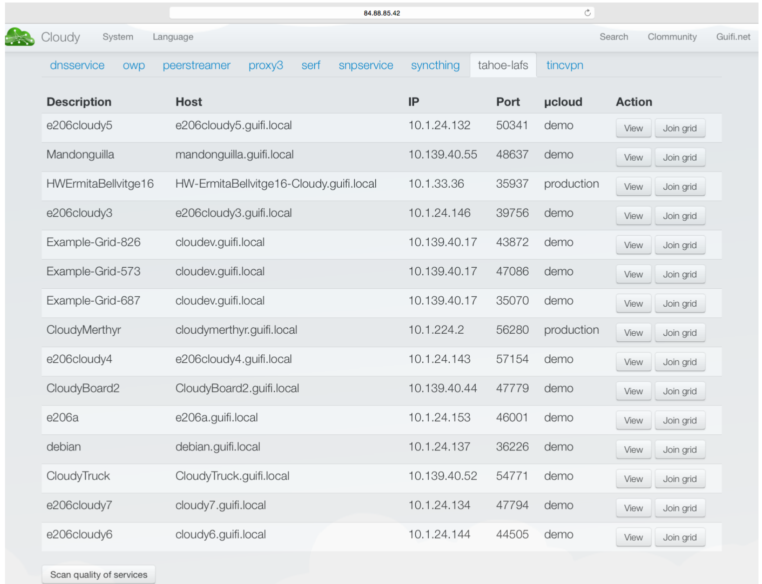
1) Infrastructure-as-a-Service (IaaS): We have set up different machines with OpenStack, Proxmox[20], Docker.io[21], OpenVZ[22], and OpenWRT[23]/LXC[24] installations, which provide either a virtual machine or container based environment. All the nodes share the same Guifi.net IP-address space and are network reachable. This means that they can support applications and services deployed on the federated infrastructure from multiple cloud setups.

2) Platform-as-a-Service (PaaS): We set up a storage service (based on Tahoe-LAFS and ownCloud) and a database service (based on CATS project's Caracal database[25]) at platform level to support the development of cloud applications for the users. The Cloudy distribution also integrates support services, like Avahi[26] and Tinc[27] which provide service discovery and management.

3) Software-as-a-Service (SaaS): We are also looking into providing useful collaborative services for the end users because these application are critical for the uptake of community cloud model among the existing users of community networks. For instance, we are setting up collaborative distributed storage service using a combination of ownCloud,

[15]http://cloudy.community
[16]http://tahoe-lafs.org
[17]http://owncloud.org
[18]http://xtreemfs.org
[19]http://syncthing.net

[20]http://proxmox.com/
[21]http://docker.com
[22]http://openvz.org
[23]http://openwrt.org
[24]http://linuxcontainers.org
[25]http://cats.sics.se
[26]http://avahi.org
[27]http://tinc-vpn.org



Description	Host	IP	Port	µcloud	Action
e206cloudy5	e206cloudy5.guifi.local	10.1.24.132	50341	demo	View Join grid
Mandonguilla	mandonguilla.guifi.local	10.139.40.55	48637	demo	View Join grid
HWErmitaBellvitge16	HW-ErmitaBellvitge16-Cloudy.guifi.local	10.1.33.36	35937	production	View Join grid
e206cloudy3	e206cloudy3.guifi.local	10.1.24.146	39756	demo	View Join grid
Example-Grid-826	cloudev.guifi.local	10.139.40.17	43872	demo	View Join grid
Example-Grid-573	cloudev.guifi.local	10.139.40.17	47086	demo	View Join grid
Example-Grid-687	cloudev.guifi.local	10.139.40.17	35070	demo	View Join grid
CloudyMerthyr	cloudymerthyr.guifi.local	10.1.224.2	56280	production	View Join grid
e206cloudy4	e206cloudy4.guifi.local	10.1.24.143	57154	demo	View Join grid
CloudyBoard2	CloudyBoard2.guifi.local	10.139.40.44	47779	demo	View Join grid
e206a	e206a.guifi.local	10.1.24.153	46001	demo	View Join grid
debian	debian.guifi.local	10.1.24.137	36226	demo	View Join grid
CloudyTruck	CloudyTruck.guifi.local	10.139.40.52	54771	demo	View Join grid
e206cloudy7	cloudy7.guifi.local	10.1.24.134	47794	demo	View Join grid
e206cloudy6	cloudy6.guifi.local	10.1.24.144	44505	demo	View Join grid

Figure 4. Cloudy interface for listing providers of Tahoe-LAFS storage service

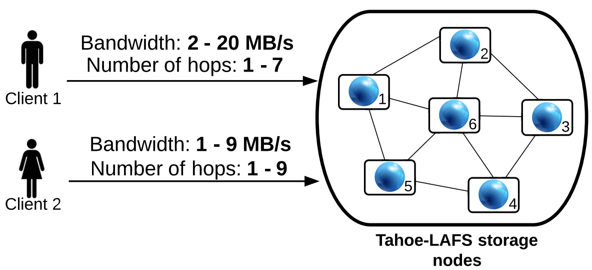


Figure 5. Two Tahoe-LAFS clients at different locations

XtreemFS and Tahoe-LAFS, and video streaming service using PeerStreamer[28] and PeerTV[29].

C. Need for Decision Support System

We have studied earlier that storage applications like Tahoe-LAFS can have quite different performance depending upon configuration, location and capability of storage nodes, as evident from Figure 5 [3]. We evaluated the storage performance of Tahoe-LAFS in the community network in order to assess the impact of network latency, connectivity and bandwidth. We focused on the read and write performance and ignored other Tahoe-LAFS features such as data recovery, repair, maintainability, etc. We collected measurements from the two Tahoe-LAFS clients at different locations in the community network for reading and writing fixed-size files. We noticed that performance is affected by the heterogeneous and dynamic network conditions of the community network.

In other experiments with BitTorrent application, we observed similar differences in quality of experience and application performance [21]. We evaluated the performance of BitTorrent for sharing small files between Guifi.net and AWMN. We set up 10 nodes from Guifi.net in Barcelona and 10 nodes from AWMN in Athens. We installed Opentracker software[30] as BitTorrent tracker on a node located in Guifi.net. The BitTorrent Transmission client[31] was installed on the other nodes, and the seeder node, which serves the file, was located in AWMN. The initial seeder provided the complete file of 30 MB and the other nodes from both Guifi.net and AWMN downloaded this file. We observed that the download performance depended on the location of the nodes and the mechanisms of the BitTorrent protocol itself. For nodes located in Guifi.net, the average download rate achieved was 5.6 Mbps resulting in download latency of 42 seconds for 30 MB file. For the nodes located in AWMN, download rate achieved was 9.2 Mbps resulting in download latency of 26 seconds for 30 MB file. This difference in performance for the similar services from different providers shows that DSS can assist in selecting appropriate services in collaborative multi-cloud systems.

We experimented with Avahi and Tinc service discovery system [23] that automatically publishes available services from a cloud provider to all the other users in the multi-cloud, since an important aspect of the multi-cloud environment is the announcement of cloud services and resources to the users. From the social perspective of the network, good communication between service providers and consumers will result in better engagement and usage dynamics. This highlighted the need to differentiate services based on their performance to make selection convenient for the users.

VI. Conclusion and Outlook

Community clouds are motivated by the additional value they would bring to community networks. Applications and

[28]http://peerstreamer.org
[29]http://www.sics.se/projects/peertv
[30]http://erdgeist.org/arts/software/opentracker
[31]http://www.transmissionbt.com

services deployed upon community clouds would boost the usage and spread of the community network model as ICT infrastructure for society. As such, it is timely to research on clouds for community networks, since mainstream cloud computing technologies are mature now and are widely used in today's Internet.

The paper analyses the key socio-technical characteristics of community networks in order to derive two community cloud scenarios, the local community cloud and the federated community cloud. These scenarios are targeted by a community cloud architecture, with the need for a service selection support system identified as a key component to ensure quality of service for the users and foster adoption of community clouds. We have deployed useful applications on community cloud infrastructures in the Guifi.net community network to assess the applications' performance, and are working on a prototype DSS that helps in selecting the appropriate services given the criteria specified by the users of these community cloud services. Such a DSS that can facilitate selection between service providers with very different resource profiles depending upon given criteria can be very useful for the users of community clouds.

Carrying onwards from the experience and results with this prototype, larger scale deployments are required with extended implementation of the different components of the community cloud architecture. This should be complemented by additional services and applications deployed in the cloud infrastructure, which will provide enhanced value and utility to the members of community networks for their contribution towards the community cloud.

We observe that the risk-analysis based DSS framework holds a high potential to meet the requirements of the community cloud users. By assisting the users to make cloud service selection, the DSS not only increases the potential to attract more active users, but also provides efficient usage of the cloud services. The agility of DSS framework makes it possible to tailor it to the community cloud, and we envision the adaptation of the DSS to more specific needs of the community cloud users in future.

Acknowledgement

This work is supported by European Community Framework Programme 7 FIRE Initiative projects Community Networks Testbed for the Future Internet (CONFINE), FP7-288535, CLOMMUNITY, FP7-317879, and MODAClouds, FP7-318484. Support is also provided by the Universitat Politècnica de Catalunya BarcelonaTECH and the Spanish Government under contract TIN2013-47245-C2-1-R.

References

[1] B. Braem *et al.*, "A case for research with and on community networks," *ACM SIGCOMM Computer Communication Review*, vol. 43, no. 3, pp. 68–73, Jul. 2013.

[2] A. Marinos and G. Briscoe, "Community Cloud Computing," in *1st International Conference on Cloud Computing (CloudCom 2009)*, ser. Lecture Notes in Computer Science. Beijing, China: Springer Berlin Heidelberg, Dec. 2009, vol. 5931, pp. 472–484.

[3] A. M. Khan, M. Selimi, and F. Freitag, "Towards Distributed Architecture for Collaborative Cloud Services in Community Networks," in *6th International Conference on Intelligent Networking and Collaborative Systems (INCoS 2014)*. Salerno, Italy: IEEE, Sep. 2014.

[4] A. M. Khan, L. Sharifi, L. Veiga, and L. Navarro, "Clouds of Small Things: Provisioning Infrastructure-as-a-Service from within Community Networks," in *2nd International Workshop on Community Networks and Bottom-up-Broadband (CNBuB 2013), within IEEE WiMob*. Lyon, France: IEEE, Oct. 2013, pp. 16–21.

[5] M. Zeleny, *Multiple Criteria Decision Making*. McGraw-Hill, 1982.

[6] B. Martens and F. Teuteberg, "Decision-making in cloud computing environments: A cost and risk based approach," *Information Systems Frontiers*, vol. 14, no. 4, pp. 871–893, Jul. 2012.

[7] S. Gupta *et al.*, "Risk-driven Framework for Decision Support in Cloud Service Selection," in *15th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID 2015)*. Shenzhen, China: IEEE, May 2015.

[8] D. P. Anderson *et al.*, "SETI@home: an experiment in public-resource computing," *Communications of the ACM*, vol. 45, no. 11, pp. 56–61, Nov. 2002.

[9] B. Chun *et al.*, "PlanetLab: An Overlay Testbed for Broad-Coverage Services," *ACM SIGCOMM Computer Communication Review*, vol. 33, no. 3, pp. 3–12, Jul. 2003.

[10] J. Cappos, I. Beschastnikh, A. Krishnamurthy, and T. Anderson, "Seattle: a platform for educational cloud computing," in *40th ACM Technical Symposium on Computer Science Education (SIGCSE 2009)*. Chattanooga, USA: ACM, Mar. 2009, pp. 111–115.

[11] X. Shen, H. Yu, J. Buford, and M. Akon, *Handbook of Peer-to-Peer Networking*. Springer Heidelberg, 2010, vol. 1.

[12] S. Distefano and A. Puliafito, "Cloud@Home: Toward a Volunteer Cloud," *IT Professional*, vol. 14, no. 1, pp. 27–31, Jan. 2012.

[13] S. Caton *et al.*, "A Social Compute Cloud: Allocating and Sharing Infrastructure Resources via Social Networks," *IEEE Transactions on Services Computing*, vol. 7, no. 3, pp. 359–372, Jul. 2014.

[14] S. Yi, E. Jeannot, D. Kondo, and D. P. Anderson, "Towards Real-Time, Volunteer Distributed Computing," in *11th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid 2011)*. Newport Beach, CA, USA: IEEE, May 2011, pp. 154–163.

[15] A. Taha, R. Trapero, J. Luna, and N. Suri, "AHP-Based Quantitative Approach for Assessing and Comparing Cloud Security," in *13th IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom 2014)*. Beijing, China: IEEE, Sep. 2014.

[16] N. Ferry *et al.*, "Towards Model-Driven Provisioning, Deployment, Monitoring, and Adaptation of Multi-cloud Systems," in *6th IEEE International Conference on Cloud Computing (IEEE Cloud 2013)*. Santa Clara, CA, USA: IEEE, Jun. 2013, pp. 887–894.

[17] "Cloud Services Measurement Index Consortium (CSMIC)," 2014. [Online]. Available: http://csmic.org

[18] A. M. Khan and F. Freitag, "Sparks in the Fog: Social and Economic Mechanisms as Enablers for Community Network Clouds," *ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal*, vol. 3, no. 8, 2014.

[19] A. M. Khan, U. C. Buyuksahin, and F. Freitag, "Incentive-based Resource Assignment and Regulation for Collaborative Cloud Services in Community Networks," *Journal of Computer and System Sciences*, Dec. 2014.

[20] ——, "Prototyping Incentive-based Resource Assignment for Clouds in Community Networks," in *28th IEEE International Conference on Advanced Information Networking and Applications (AINA 2014)*. Victoria, Canada: IEEE, May 2014, pp. 719 – 726.

[21] M. Selimi and F. Freitag, "Towards Application Deployment in Community Network Clouds," in *14th International Conference on Computational Science and Its Applications (ICCSA 2014)*, ser. Lecture Notes in Computer Science. Guimaraes, Portugal: Springer International Publishing, Jul. 2014, vol. 8584, pp. 614–627.

[22] J. Jiménez *et al.*, "Supporting cloud deployment in the Guifi.net community network," in *5th Global Information Infrastructure and Networking Symposium (GIIS 2013)*. Trento, Italy: IEEE, Oct. 2013, pp. 1–3.

[23] M. Selimi *et al.*, "TROBADOR: Service Discovery for Distributed Community Network Micro-Clouds," in *29th International Conference on Advanced Information Networking and Applications (AINA 2015)*. Gwangju, Korea: IEEE, Mar. 2015.