Bringing Semantic Resources together in the Cloud: from Theory to Application

Salvatore F. Pileggi
Department of Computer Science
The University of Auckland
Auckland, New Zealand
f.pileggi@auckland.ac.nz

Jaime Calvo-Gallego
Department of Computing and Automatics
University of Salamanca
Salamanca, Spain
jaime.calvo@usal.es

Robert Amor
Department of Computer Science
The University of Auckland
Auckland, New Zealand
trebor@cs.auckland.ac.nz

***Abstract*—This paper deals with the added value provided by Semantic Technologies in cloud environments. In these contexts, semantics are not understood as a massive technology but as a resource in order to improve cloud platforms' capabilities in terms of interoperability, knowledge building/representation and management. The proposed approach aims at the extension of the common middleware functional layer in complex architectures through semantics. This added capability should enable (active and passive) heterogeneous resources to work together as in a unique ecosystem, as well as supporting innovative interaction models involving these resources. The ideal application could be the Smart City.**

Keywords: Cloud Computing; Semantic Technologies; Smart City; Semantic Profiling.

I. Introduction

The ICT society is experiencing a number of critical trends [1] that are radically changing the understanding of most systems and services. Apart from the advances involving each and any aspect of the technological environments and from the progressive growth of infrastructures (e.g. fast internet connection) and device capabilities, computational resources are progressively converging into cloud infrastructures as virtual resources [2].

On the other hand, the social trend to information appears to be unstoppable, especially if current applications and services are analyzed according to an evolving understanding of themselves or as futuristic concepts [3].

The explicit and evident need for semantics in a technological context not yet ready for a massive application of semantic technologies [4] is completing a complex research scenario mostly featured by the convergence of different trends and processes.

Suddenly, current cloud infrastructures (and related technology) look as if they are not able to cover a domain that is progressively increasing for both its complexity and its purpose. The great availability of stand-alone services that independently work even if they are logically related is the clear evidence today's platform models are not fully addressing new assets from the real world. The cloud approach assures scalable, competitive and sustainable solutions and it is potentially able to enable ecosystems among heterogeneous resources through pervasive virtual environments in which resources are managed at virtual levels assuring highly-interoperable capabilities. Semantics could play a key role in this context, enabling semantic ecosystems among cross-domain platforms: the core infrastructure of the platforms is not domain-specific and the potential application range increases with the expressivity of the semantics. Theoretical aspects have to be considered in the context of real business and social scenarios where highly-flexible solutions, able to meet the needs and requirements of complex virtual organizations, are requested. In fact, simple theoretical processes (such as the migration to the cloud) could be strongly limited by factors completely unrelated to technological issues (e.g. law restrictions).

This paper proposes a platform model resulting from the convergence of cloud technologies, semantics and social trends aimed to enable semantic ecosystems among heterogeneous resources. Even if different research challenges from different research areas coexist in the model, the paper mostly focuses on the added value provided by Semantic Technologies in cloud environments. In these contexts, Semantics are not understood as a massive technology but as a resource in order to improve cloud platforms' capabilities in terms of interoperability, knowledge building/representation and management. The proposed approach aims at the extension of the common middleware functional layer in complex architectures through semantics. This added capability should enable (active and passive) heterogeneous resources to work together as in a unique ecosystem, as well as supporting innovative interaction models involving these resources. The ideal application could be the Smart City [5].

A. *Methodology: Heuristic Approach for Knowledge*

The design of a platform able to bring together heterogeneous resources as in modern trends proposes the convergence of critical issues from different research areas (e.g. Cloud platforms [2], Social Computing [3] and Semantic Technologies [6]). Each of these areas contributes to the overall platform by introducing advanced features and also, unfortunately, key trade-offs.

The real convergence point for these features is knowledge modeling. In fact, the semantic approach mostly implies advanced profiling for all the actors involved in the process (user profile, social profile, service profile). Under this perspective, the overall platform capabilities match the knowledge the platform is able to represent and manage.

Methodological aspects of research have a strong impact on the platform design and on the applicability, extensibility and maintenance. A flexible understanding of knowledge and the consequent extensibility of ontological structures has to be assured in order to support heterogeneous domains.

The ideal approach would be the full modeling of profiles according to any perspective (user, social, service), as well as the full modeling of any domain aspect involved in the process. That is an interesting and challenging topic, as the great number of studies and research initiatives appear to confirm. But, if the goal of the research is effectively a working platform more than an academic exercise, a global approach could result in inefficient models and in an objective difficulty of application in different contexts and domains. Furthermore, any life cycle step posterior to design (such as maintenance or extensibility) could be strongly limited or conditioned.

From the expertise collected so far emerges a strong step from theory and application. An alternative approach aimed at concrete applications is informally called a *heuristic approach*. A full understanding of profiles and domains is not requested a priori. Required models are the result of the progressive integration with *Clusters of Knowledge* as a consequence of further resource enablement. In practice, the platform knows exactly what it needs to know in order to allow current services and applications to run.

The existent set of assets (use cases, scenarios, requirements, concerns from involved stakeholders) corresponding to the set of resources currently enabled on the platform contributes in order to provide the platform with a "piece" (or cluster) of knowledge, as well as an input for domain modeling (*Reference Model*). The inability to map an informal representation of the knowledge into an ontological representation [7] determines the need of simplifications or, in very complex cases, the impossibility to include the considered knowledge cluster into the platform. Local knowledge availability basically means resources enabled to work on the platform and any service/application using them working on the platform as a stand-alone service/application. The process of including clusters of knowledge is repeated for any new resource or service/application to be integrated in the platform that acts as logic trigger for the knowledge building.

B. Related Work

Resource ecosystems [8] are not an absolute novelty since they are a quite popular research field in the context of different domains (e.g. [9]), for different purposes (e.g. [10]) and in order to reach different scopes (e.g. [5]).

Most existing platforms manage virtual resources but, in the most cases, they focus on specific resources (e.g. services or sensors) and they are normally oriented to specific domains.

Furthermore, they seem to completely or partially ignore social trends and so resources look as they are sharing a physical infrastructure more than that they are part of an ecosystem. In fact, the level of cooperation among resources is quite poor or non-existent even when the logic relations among user services are strong. An overall approach could significantly improve most of the features. Finally, semantics inside models are quite limited and they normally focus on specific aspects of profiling (service or users). Even if the level of interoperability is progressively increasing, semantics should have a more consistent role inside platforms.

II. Platform Model

Complex platforms representing virtual ecosystems are not easy to specify as a unique reference architecture, mostly because different stakeholders could have a completely different view of the platform. Furthermore, as for any reference architecture, concrete specific-purpose design and implementation could propose peculiarities and strong differentiation since they are the response to different problems, requirements, needs and concerns. That is mostly the reason for speaking about a *Platform Model* more than about a platform. The paper does not propose a specific architecture but an approach to make that architecture compliant with a set of features prioritizing the question *What* more than *How* (and *Why*). According to the previous considerations, multiple *perspectives* (or *views*) of the platform can be provided. In the following section, three different perspectives will be adopted in order to focus on different features of the platform model (Figure 1).

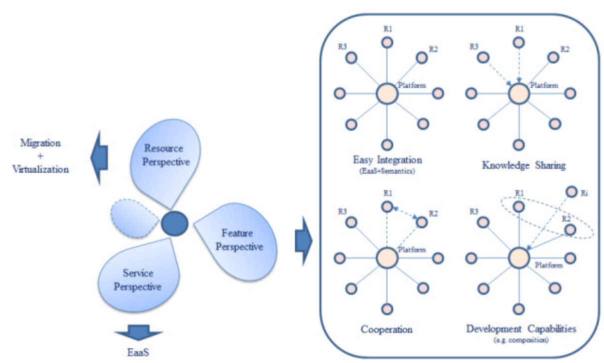

Figure 1. Different perspectives of the model and schematic view of platform features.

Semantic resources bridged together in distributed cloud platforms provide the interconnection of heterogeneous services in a context of interoperability and flexibility, under a virtual organization schema. Only the key concepts featuring the platform in the context of next-generation cloud architectures are considered. An exhaustive definition of the model, as well as its relations with the different business contexts, could be extremely interesting but out of the paper scope.

The most immediate and intuitive perspective is the *Resource Perspective* that provides a classification of resources from the infrastructure point of view and has an impact on both performance and the business model. This perspective is limited to computational resources. At least three different kinds of coexistent resources can be distinguished: *Internal Resources* (resulting from the common migration process and hosted by internal infrastructures), *External Resources* (hosted by external infrastructures, they are pervasively available in the platform as virtual resources) and *Composed Resources* (designed and implemented directly over the virtualized layer provided by the platform).

A parent concept for the resource perspective is the *Service Perspective*. In this context, resources are mostly synonymous with services since the platform is explicitly service-oriented and it works according to an *EaaS* [11] schema. In fact, cloud computing has progressively become a generic concept that commonly describes an easy, flexible, and scalable delivery of resources and services over the Internet. *EaaS* clearly overcomes the classic look at cloud technologies, solutions and applications since each and any resource is modeled as a service. The availability of cloud services for developers is constantly increasing. Apart from common services (such as storage in the cloud), the main feeling is that cloud capabilities are constantly increasing (e.g. *Cloud of Clouds* [12]). Furthermore, the cloud looks to be the line chosen from the giants of the web to deliver their products. A clear example is *BigQuery* [13], a scalable high-performance on-demand service provided by Google for interactive analysis of massive databases (up to billions of rows according to the provider's information).

The third perspective is probably the view that mostly matches the declarative approach since the platform is specified as a list of challenging high-level features. According to the *Feature Perspective*, a great number of features should be defined at different layers in the platform. Considering the complexity of resulting architecture, this approach could not be exhaustive in practice, as it could propose several ambiguities due to a non-functional understanding of the specifications. But if, as in this case, the view is not referred to a concrete platform and, on the contrary, it is related to an abstract model, than the feature perspective can specify just the key challenging feature of the platform without any contextualization or link to functional layers or elements. At least four high-level features have been identified: resources have first to be integrated in the platform, as well as have to be “understood” and managed inside the ecosystem (*Easy Integration*); each resource provides a knowledge that has to be represented inside the platform (*Knowledge Building and Sharing*); resources have to be able to cooperate among themselves in the ecosystem in order to achieve common goals; finally, the semantic layer extending the common functional capabilities of the middleware is a powerful resource for designers and developers that can look at any existent resource as a semantic artifact inside the ecosystem (*Cooperation* and *Extended Development Capabilities*).

In the next future, the model will be integrated with further views mostly aimed at the specification of the functional features.

A. *Knowledge Building and Sharing: Semantic Profiling*

Profiling is the process of examining and modeling the information available in an existing data source. In order to archive dynamic features listed in the previous section, at least three different models (*Profiles*) for data sources are requested: *User Profile* (is aimed at modeling the knowledge of the user as individual), *Social Profile* (which should fully represent the relations among individuals, among resources, as well as among individuals and resources) and *Resource Profile* (is the correspondent of the user profile applied to resources).

The modeling of the resource itself (*Resource Profile*) has to be clearly separated from the representation of the correspondent virtual resources (*Service Profile*).

The coexistence of different profiles results in increased design capabilities in function of concrete requirements. In fact, focusing on different aspects of profiles mostly determines different features for the platform and so a potentially easier adaptation to real contexts. All profiles have a strong impact on platform features. Omitting (or not focusing enough on) the Social Profile allows just limited features. Omitting (or not focusing enough on) the User Profile strongly limits the social focus and, in practice, proposes models similar to existing platforms (e.g. Apple, Google, etc.) Omitting (or not focusing enough on) the Resource/Service Profile makes it hard to relate user needs and services (lack of dynamism).

B. *An application scenario: the Smart City*

An exhaustive overview of the potential application fields for the platform model is out of the paper scope. One of the ideal target scenarios could be the *Smart City*. That is the typical futuristic scenario that, if in the context of a sustainable business model, could be next to being a fact. Regardless of the smart city purpose (e.g. governance, ageing, living, optimization, economy, mobility, environment), apart from common computational resources, a great variety of heterogeneous resources are available in any city.

Humans are the most relevant resources in the city and they have not always had the central role they should have. People live in the city and, from a certain point of view, they are the city. Services for people and strongly involving people are expected in the city of the future.

Social Resources are needed since citizens are a resource as individual but also as collectives. Their needs (e.g. impaired or elder people), their interests (e.g. sport, cinema), the relationships among them establish virtual communities into the societies.

Transport, *city services* (e.g. taxis), *sensor data*, *vehicles*, *Smart Space* (houses as well as any other kind of shared space such as hospitals, libraries), *Smart Things* (e.g. clock, tv, refrigerator) are just a few examples of the great number of resources potentially available.

A deep analysis of the relations among resources, as well as the potential purpose of their cooperation, could be very interesting but out of the paper scope. In order to provide an example of potential benefits for the whole ecosystem, an example of an extended business scenario is proposed (Figure 2). A novel approach such as the *Internet of Things* [14] basically assumes proactive objects connected to the Internet and, so, they are able to be an active part in different processes involving services and applications. For example (Figure 2, IoT Scenario), an intelligent refrigerator could know when some food is finished (or is going to finish) and help business persons automatically asking a supermarket for this food, eventually notifying the owner or asking the owner for approval. Also this innovative scenario can be improved (Figure 2, Smart City Scenario) if the smart refrigerator is connected to the smart city and working in the context of the ecosystem. In this last case, the ecosystem allows interaction with a set of supermarkets in potential competition. They can provide a personalized offer since they know the consumer profile. So the consumer will have more than one offer to choose. The potential benefits in respect to the previous scenario should be quite evident: the consumer, apart from saving time, has a range of offers that should theoretically provide a better price (or relation quality/price) for him/her; supermarkets can provide offers through a competitive and personalized business environment on the basis of facts (he/she is a consumer, he/she is going to buy the good) and not just through massive channels (e.g. tv, radio, internet) or massive social networks (the user just indicated he potentially likes something, he is not necessarily a consumer or he is going to buy…). Furthermore, a new business actor (the platform manager) appears.

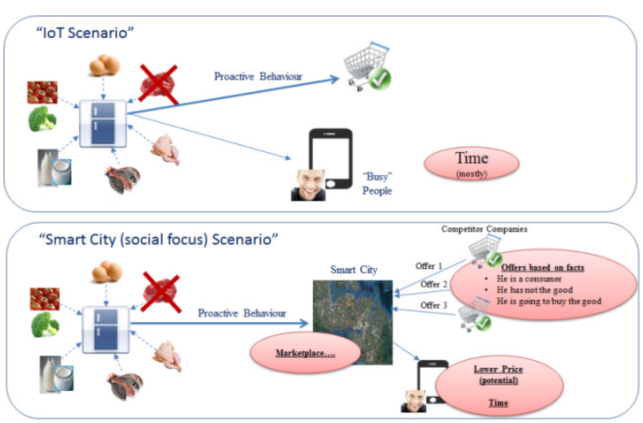

Figure 2. Example of extended business scenario.

C. *Preliminary performance evaluation*

A preliminary performance evaluation is proposed in order to provide reference values related to the potential performance decreasing mostly introduced by the EaaS approach.

In practice, previous stand-alone services are now accessed through a unique platform as part of an ecosystem of resources. The platform, from an architectural perspective and so in terms of performance, acts as a proxy server (end-user side) and as a service coordinator (platform side) allowing resources to work together. Services could be deployed on the platform infrastructure, or be deployed on external clouds, as well as being mixed resources. Other relevant factors for global performance are the ontological approach (that implies semantic computation on the information) and the delivery infrastructure. They are interesting factors for performance evaluation but they are not currently considered.

The performance context has a strong equivalence to service-oriented computational grids [15]. In fact, computational grids enable complex ecosystems among users and resources through services and applications. A generic performance evaluation from the end-user perspective is hard to propose mainly because real performances are characteristic of concrete architectures. Under the not always realistic assumption that service and application performance are proportional to basic operations, the analysis of performance can be generalized.

Two main scenarios are considered: *Local Area Network (LAN)* and *Wide Area Network (WAN)*. In the first one, involved actors are connected to the same LAN. This simple scenario is a good approximation of local clouds/grids, in some cases smart spaces or small-scale cities (e.g. a university campus). In the second one, client-side and server-side are connected to different networks. There is not a well-defined reference topology and any assumption about the status of the network is missed, as well as any class of QoS control on the network. This is a generic scenario that assures a pervasive vision at virtual organizations.

The considered WAN scenario assumes the client and the server are connected by the *RedIRIS* WAN support [16]. *RedIRIS* (Figure 6) is a Spanish academic and research network that provides advanced communication services to the scientific community and national universities. Details about topology, IP addresses and routing tables are omitted due to security reasons.

The main interest parameter for the preliminary experimental evaluation is mostly the response time (Tr), normally related to the ping time (Tp). This is a simple estimated evaluation of the network performance at the time of the measurement. It can be related to the response time as in (1) or in (2), according to the experiment goal. In practice, average values of pings on the period are considered (1.1 and 2.1) in order to assure a consistent reference about the status of the network for experiments that assumes a relatively large observation time.

$$D_d(i) = T_R(i) - T_P(i) \quad (1)$$

$$\overline{D}_{d,I}(i) = T_R(i) - \overline{T}_{P,I} \quad (1.1)$$

$$D(i) = \frac{T_R(i)}{T_P(i)} \quad (2)$$

$$\overline{D}_I(i) = \frac{T_R(i)}{\overline{T}_{P,I}} \quad (2.1)$$

TR(i)	Response Time (i-th experiment)
TP(i)	Ping Time (i-th experiment)
$\overline{T}_{P,t}$	Average Ping Time on the period t
Dd(i)	Response Time, without round-trip time (RTT), i-th exp.
$\overline{D}_{d,I}(i)$	Average value of *Dd(i)* on the period t
D(i):	Relative response index over the ping time (i-th experiment)
$\overline{D}_I(i)$	Relative response index over the average ping time (i-th exp.)

Within both scenarios, LAN and WAN, a client randomly requests services deployed on the platform's server. Meanwhile the monitoring tool checks the status of the network.

Each experiment includes 100 service requests (Figure 5, a.1 and a.2, for LAN and WAN scenario respectively). Analyzing an example experiment, it is evident that for both scenarios there are irregularities of performance due to the load of the network.

An extended analysis is showed in Figure 5 (b.1 and b.2, for the two scenarios respectively), in which 60 independent experiments are represented both with the related standard deviation: LAN performance is extremely regular (low deviation), on the contrary, as expected the WAN scenario proposes some important irregularities (high deviations).

Finally, we present the average values for the response times in relation with the average value of the ping time on the observation period, ordering values according to an increasing value of the corresponding ping times (Figure 5, c1 and c2). They represent the relationship between performance (response time) and network status (ping time). Regular results should show an increasing behavior of the points in the graph.

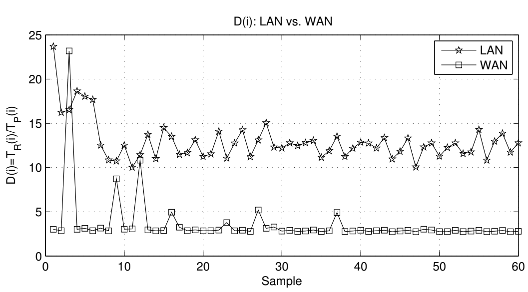

Figure 3. LAN vs WAN.

The services requested to the server from the client are basic services (such as get variables values, set variables values, access to remote file systems).

The comparison of the two scenarios is realized according to the parameter defined by (2.1). It is shown in Figure 3. When the network complexity increases, the ping times and the response times have a partially convergent behavior.

An equivalent compact representation useful for monitoring tools is based on the following metric:

$$\gamma(i) = \frac{1}{D(i)} \quad (3)$$

$$\overline{\gamma}_I(i) = \frac{1}{\overline{D}_I(i)} \quad (3.1)$$

$$0 \le \gamma(i), \overline{\gamma}_I(i) \le 1 \quad (4)$$

$$\begin{cases} \gamma(i) = 0, T_P(i) \to 0 \text{ or } T_R(i) \to \infty \\ \gamma(i) = 1, T_P(i) = T_R(i) \end{cases} \to 0 < \gamma(i), \overline{\gamma}_I(i) < 1 \quad (4.1)$$

$$\overline{\gamma}_I(i) = \cos(\alpha) \;\to\; \alpha = \arccos(\overline{\gamma}_I(i)) \quad (5)$$

$$0 < \alpha < \pi/2$$

According to (6) and (7), lower values of α reflect better performances.

$$\alpha \to 0, \cos(\alpha) \to 1, \overline{\gamma}_I(i) \to 1, T_P(i) = T_R(i) \quad (6)$$

$$\alpha \to \pi/2, \cos(\alpha) \to 0, \overline{\gamma}_I(i) \to 0, T_R(i) \to \infty \quad (7)$$

Results represented according to the α metric are shown in Figure 4. When the connection network complexity increases, the variability of ping times also increases, as well as the variability of the response times. This is clearly reflected in the α values that are included in a short range [84.2788º, 87.5797º] for LAN and in a relatively high range [68.7143º, 87.5279º] for WAN. This last result also shows evidence of the regular physical behavior that characterizes the networks used for the tests. In the future, the experimental analysis will be extended also considering networks with different features.

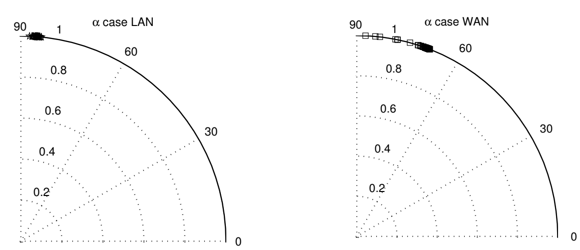

Figure 4. Data representation according to the α (in degree).

III. CONCLUSIONS

Several concrete environments involving complex virtual organizations could require open models in order to allow the integration and management of high level resources bridged together in a unique ecosystem. This understanding of platforms, services and resources provides a new perspective of the exploitation models for cloud environments.

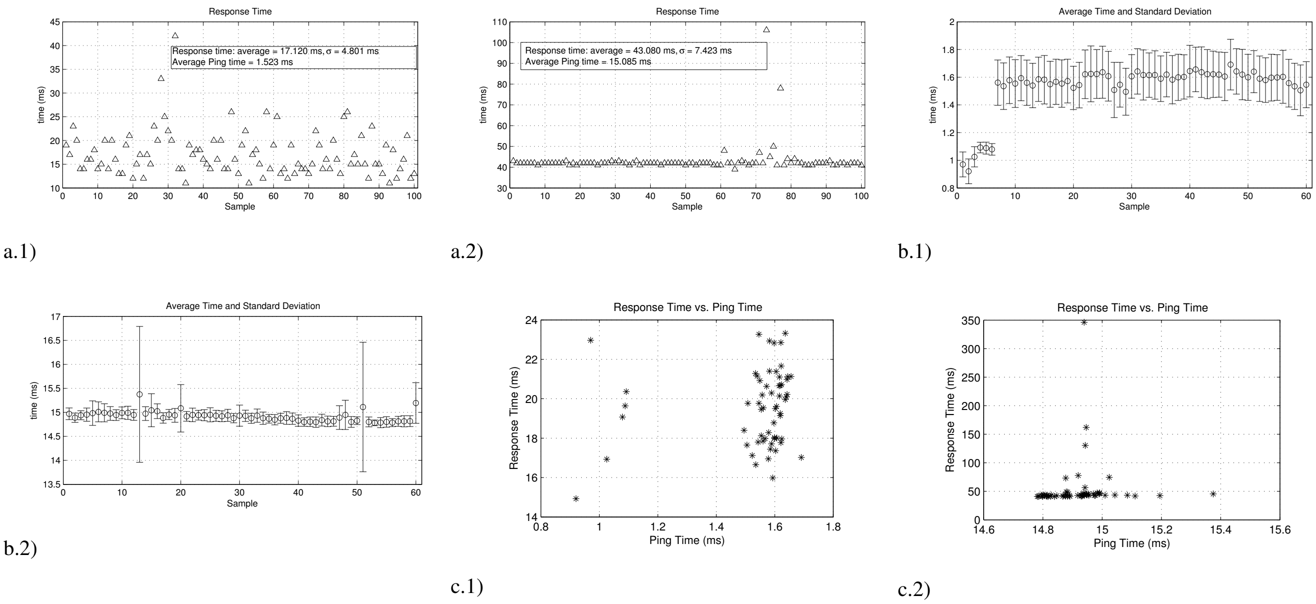

Figure 5. Experimental results.

The flexible support for the effective convergence among dynamic resources in the cloud is provided through a completely open model for both services (*EaaS*) and knowledge (semantic representation). Even providing a preliminary evaluation of the environment and an example of application, this paper mostly deals with the approach itself.

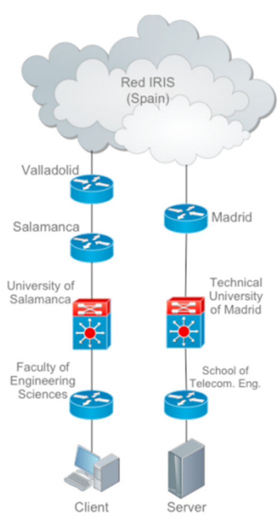

Figure 6. Reference network.

REFERENCES

[1] Adepetu, A.; Arnautovic, E.; Svetinovic, D.; de Weck, O.L.; “Complex Urban Systems ICT Infrastructure Modeling: A Sustainable City Case Study”, IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2013.

[2] Weiss A.; “Computing in the clouds”, netWorker, vol. 11, no. 4, pp. 16–25, 2007.

[3] Wang, Fei-Yue; Carley,K.M; Zeng, D.; Wenji Mao; "Social computing: From social informatics to social intelligence", Intelligent Systems, IEEE 22.2 (2007): 79-83.

[4] Breslin, J.; Decker, S.; “The Future of Social Networks on the Internet: The Need for Semantics”, Internet Computing, IEEE , vol.11, no.6, pp.86,90, Nov.-Dec. 2007.

[5] Gil-Castineira, F.; Costa-Montenegro, E.; Gonzalez-Castano, F.J.; Lopez-Bravo, C.; Ojala, T.; Bose, R.; “Experiences inside the Ubiquitous Oulu Smart City”, Computer , vol.44, no.6, pp.48,55, June 2011.

[6] Hendler, Jim; "Web 3.0 Emerging", Computer 42.1 (2009): 111-113.

[7] Noy, Natalya Fridman, and Carole D. Hafner; "The state of the art in ontology design: A survey and comparative review", AI magazine 18.3 (1997): 53.

[8] Liu Pingfeng; Nie Guihua; “Research on Service Ecosystems: State of the Art”, Management and Service Science, 2009. MASS '09.

[9] Ludwig, C.; Kuster, M.W.; “Digital ecosystems of eHumanities resources and services”, 2nd IEEE International Conference on Digital Ecosystems and Technologies, 2008. DEST 2008.

[10] Zhang Shi Ming; Huang He Xiao; “Research on digital education resources sharing ecosystem”, 2011 International Conference on Electrical and Control Engineering (ICECE), Sept. 2011.

[11] Banerjee, P.; Friedrich, R., Bash, C.; Goldsack, P.; Huberman, B. A.; Manley, J.; Patel, C.; Ranganathan, P.; Veitch, A.; “Everything as a service: Powering the new information economy”, Computer, 44(3), 36-43.

[12] Bessani, A.; Correia, M.; Quaresma, B.; André, F.; Sousa, P; ”DepSky: dependable and secure storage in a cloud-of-clouds”, In Proceedings of the sixth conference on Computer systems (pp. 31-46). ACM.

[13] “Google BigQuery”, https://developers.google.com/bigquery/.

[14] Yinghui Huang; Guanyu Li; “Descriptive models for Internet of Things”, 2010 International Conference on Intelligent Control and Information Processing (ICICIP), 13-15 Aug. 2010.

[15] Fox, F.; Gannon, D.; "Computational grids", Computing in Science & Engineering , vol.3, no.4, pp.74-77, Jul/Aug 2001.

[16] “RedIRIS”, http://www.rediris.es/.