Collaboration and the Knowledge Economy: Issues, Applications, Case Studies
Paul Cunningham and Miriam Cunningham (Eds)
IOS Press, 2008 Amsterdam
ISBN 978–1–58603–924-0

OpenAdap.net: Dynamical and Open Source Approach to Knowledge Sharing

Alessandro E.P. VILLA[1,2], Javier IGLESIAS[1,2]
[1]*Grenoble Institute of Neuroscience GIN, Centre de Recherche Inserm U 836-UJF-CEA-CHU, Neuroheuristic Research Group, Université Joseph Fourier, France*
Email: Alessandro.Villa@ujf-grenoble.fr, Javier.Iglesias@ujf-grenoble.fr
[2]*Information Systems Institute, Université de Lausanne, Switzerland*

Abstract: Repeated unnecessary expenses are generated at worldwide scale with tremendous impact, particularly on less favoured countries. We envision the emergence of Communities of users sharing knowledge, information processing, communication, storage and content for their specific domain of interest within an innovative generic platform-independent middleware framework, called OpenAdap.net (OAN), which is not bound to a specific type of information. The key to this vision is an Open Source development based on Open Standards that provides Community members with pervasive access points like web portals, desktop and mobile applications. Internet Communities will have the possibility to transparently compose meta-resources based on the resources available in their own Community, as well as in Communities with semantically related interest, by means of a network-centric operating system driven by the activity of intelligent adaptable brokers. The win-win business model of OpenAdap.net is based on the adoption of OAN and the development of the brokers thanks to an Open Source and patent-free policy. This model is aimed at the widespread dissemination of knowledge without economical barriers, e.g. at no cost to the Communities that will rely on this system, and contains no profit motivation whatsoever. A case study illustrates how collaborative working is actually implemented by OAN in a Community of scientific researchers. The potential benefits appearing with the growth of the adoption of OAN open the way to unlimited Third Parties and added-value services with an embedded advantage for worldwide competitive businesses and less favoured Web users.

Keywords: Free Software, knowledge sharing, virtualization, distributed operating system

1. Introduction

The Cyberspace has become the main site of information exchange and its extension in terms of bandwidth is opening the way to all media of information. This extension has produced a shift towards increasing user interaction with content. However, most of this information flow still remains associated to a declarative approach, i.e. associated to the knowledge of "what", as illustrated by the popular success of Wikipedia and search engines in general. Internet has still a huge unexploited potential in applying the procedural approach, complementary to the declarative one, based on the knowledge of "how".

Along the direction set by the Web 2.0 we believe that the people who actively access Internet and use the Web and its newly expanded multimedia capabilities should not passively absorb the available information. Rather, Internet users should become active contributors, helping customize media and technology for their own purposes, as well as those of their Communities. This view implies that Communities should be clearly defined by the format of the shared information, still maintaining a dynamic, adaptive and extensible capability to evolve.

This paper introduces OpenAdap.net, which is a middleware that provides platform-independent access and sharing of knowledge, not bound to a specific type of information, by enabling the execution of tasks and workflows across multiple domains of expertise. Internet Communities have the possibility to transparently compose and share meta-resources based on the resources available in their own Community, as well as in Communities with semantically related interest, by means of a network-centric operating system driven by the activity of intelligent adaptable brokers.

2. Objectives

The emergence of Communities of users sharing knowledge, information processing, communication, storage and content for their specific domain of interest should rely on an Open Source development based on Open Standards that provides Community members with pervasive access points like web portals, desktop and mobile applications. OpenAdap.net detailed objectives are:

- to provide a trusted and dependable service infrastructure that supports sharing and dynamic composition of resources across multiple domains and multiple platforms;
- to provide a new virtualization middleware for a better use of the already existing resources avoiding the re-invention of solutions tested and validated by Community members by means of an intelligent workflow analysis of the circulating information and the aggregation of emerging resources;
- to open the way to cross-fertilization and serendipity by providing the means to break current boundaries of resource sharing between semantically close, but possibly isolated, communities, thus enabling a true transdisciplinary approach to information processing.

3. Methodology

3.1 Who is Who?

Individual users of OpenAdap.net can be classified as either *contributors* of shared resources or *service consumers* (end-users). People who develop and distribute new knowledge methods of information processing can share their contribution with people who have collected the information to process or have access to content stored elsewhere. In addition to individual use, OpenAdap.net is open to exploitation by networked organizations by providing an open infrastructure offering new business opportunities.

By sharing a program and the computer running it contributors keep complete control over their authorship as well as the source and binary codes for the software. By this way, the service consumer does not own the service and contributors' intellectual property is preserved. Users' privacy is as important as contributors' traceability: any piece of information sent by an end-user for processing to a contributors' resource is anonymised, yet the transactions are identified and the activity is traced like in any computer system. Meta-information concerning the contributed resources including program authorship and version number are made available to the end-user, as a mark of diligence to the contributor, but also to emphasize the responsibilities in the information processing chain and enhance quality assessment and reproducibility. From the viewpoint of contributors, OpenAdap.net makes possible the dissemination of resources, and their exposure to application and evaluation by a broader user community with limited packaging, distribution and maintenance costs. Then, the evaluation of the quality of service becomes performed, at least partly, by the service consumers within their community. OpenAdap.net supports this collaborative process through the integration of existing technologies encompassed by Web 2.0 tools (e.g. RSS, tags, social bookmarking, blogs, AJAX) [1].

3.2 Virtualization

The components of OpenAdap.net are running on a network of computers, each of them defined by their specific CPU architecture, operating system (OS), amount of memory and available programs. Their resources are partially shared in a dynamic way. OpenAdap.net is designed by analogy with a virtual distributed OS in which all the information is presented in a structured virtual file system. Using this high-level paradigm, resources are assigned to files in the file system tree. Security can be enforced through ownership and access permissions defined on a per-file basis.

OpenAdap.net goes beyond a traditional OS as the configuration is highly volatile. The file system structure and contents result from the continuous runtime aggregation of several sources of information: the user database, the inter-process message queues status, the worker status, etc. For example, information concerning the system (including load, user's jobs status, etc) is made available to all the components through virtual files containing this information formatted in a self-describing format (an XML grammar) to be interpreted and rendered to the end user in the appropriate way. Brokers dynamically generate the content of these virtual files on-demand with up-to-date information collected at runtime, while the other components (other brokers, workers and OAN-aware applications) request for the contents through a documented URL, provided authorisation requirements met by the authenticated connection.

Meta-information is associated to each file including authorisation details, size, content-type, etc, and becomes accessible to all components of the middleware via the OpenAdap.net API. This implementation enables, for example, the content-specific graphical rendering of the information by end-user tools tailor suited to a community (see a case study in Section 5). The composition of resources are mapped to a daisy chain of resources creating a pipe-like meta-resource operating on virtual file system files.

In the virtual operating system paradigm, management tools are designed like their desktop OS counterparts offering an intuitive interface for people with limited knowledge in computer science who wish to undertake the task of assembling and developing a new Community. Concepts like *users* (adding, removing, contacting, etc), *resources* (disk space, load, etc) and *process statuses* are integrated by the middleware and presented in an intuitive way akin to a desktop computer. This approach to the management of broker and worker operations is aimed to facilitate the adoption of the platform and the maintenance of the system by the broadest Internet community.

4. Technology Description

4.1 System Design

The resources shared by the community can take many forms, including but not limited to: data (like benchmark data), methods (like programs), and hardware (like dedicated devices or data storage space). They are semantically organized and appear as a single entity able to orchestrate unlimited, heterogeneous and dynamic resources distributed across multiple platforms. Brokers, workers and OAN-aware applications are the three types of innovative interconnected components of OpenAdap.net.

Brokers are designed to optimize usage of distributed resources with the specific goal to address scalability and interoperability issues. In practice, a broker is a process permanently running on a server, in charge of managing a community of users and dispatching tasks and results on their behalf in a secured, trusted and dependable way [2]. Internally, domain-oriented brokers are responsible for decomposing and routing end-user tasks to appropriate workers for execution. This situation is dynamically evolving with the addition and removal of users, resources and workers. In addition, two brokers, say broker A and broker B

serving communities A and B, respectively, can establish direct communication links, either after an autonomous discovery mechanism and semantic evaluation of their respective know-how and/or an explicit agreement/validation between the broker managers.

A broker can syndicate to its community the resources of interest provided by other brokers on the basis of the ontological description of each resource. The network of brokers remains transparent to the end-users, such that information processing by a user belonging to the community A that requests a resource shared by a contributor of the community B will be anonymised by broker A and transmitted to broker B for execution. The interconnected brokers are designed to behave as user and worker components from the point of view of the peer. This relationship is transitive and broker B can also anonymise the information submitted by the community A user via the broker A and send it to a third broker (say broker D) for execution by one of its workers. This raises issues related to distributed trust and resource contextualisation that must be addressed in our further development. The required negotiation between brokers (and workers) may be compared to agent interactions. A key element in our approach consists in making brokers adaptive in the dynamically interconnected OpenAdap.net network. The requests for resources are dispatched among the components of the system following a set of rules able to modify dynamically the routing according to bioinspired machine learning algorithms.

Workers are processes shared by community members in charge of giving secured distant access to contributed resources. Workers are responsible for connecting to one or more brokers; for notifying the list of resources they share; for receiving requests from brokers; for supervising their realization; and for returning the results. The reference implementation is able to execute programs, execute database queries, retrieve documents and interact with existing Grid systems.

OAN-aware applications are pieces of software providing end-users access to the community resources through identified connections to a broker. These OAN-aware applications take many forms: standalone programs, mobile applications, light-weight web interfaces, command-line tools, etc. The resource definitions are interpreted by the OAN-aware applications, which generate an appropriate interface for guiding the user through the necessary steps required for submitting information to the broker.

4.2 Implementation

In line with the NESSI initiative[i], we intend to integrate almost "commoditised" Open Source software pieces and Open Standards implementations and extend them into the middleware stack at the core of OpenAdap.net, with the objective to reduce vendor lock in and obtain an open and secure platform at low development and maintenance cost.

The Java 2 Platform was chosen for the project, based on portability and platform neutrality requirements, as Java Virtual Machines are available for free from several sources and for almost every computer platform. Brokers, workers and OAN-aware applications are loosely coupled, distributed components that asynchronously communicate through a message-oriented middleware (MOM) as defined by the Java Message Service (JMS) API (see Figure 1), an open standard maintained under the Java Community Process JSR-914 [4]. Furthermore, the Java Platform Micro Edition (J2ME) is an application environment running on a broad range of portable devices, such as mobile phones, PDAs, TV-set boxes and printers. The compatibility with the standard Java edition enables the implementation within the project of mobile applications, apart from the mobile access to the reference portal through optimized mark-up languages.

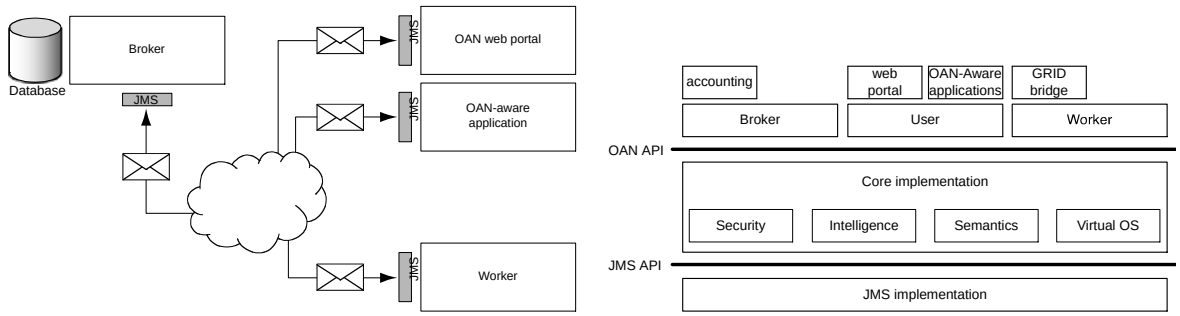

Figure 1: ***Left.*** *Message Oriented Middleware overview. Components of the distributed OpenAdap.net system exchange information in a way similar to the use of email among humans. Component A explicitly addresses a message to Component B. The JMS implementation is in charge of delivering the message through Internet while the logics of the message is determined by the OpenAdap.net middleware for maximal interoperability;* ***Right.*** *OpenAdap.net architecture showing the relation between the different components of the middleware*

The core implementation includes the security, intelligence, semantics and virtual OS features (Figure 1, right panel). We are currently working to provide API with Java, C and Python reference implementations. Based on the API and Open Source examples, Third Parties are allowed to develop and/or extend generic or propietary tools (e.g., see the Case Study). Furthermore, a generic OAN-aware web portal featuring support for mobile devices and examples of OAN-aware standalone and applications is scheduled in the roadmap.

5. Case Study

In December 2006 a case study was implemented in collaboration with a community of neuroscientists[ii] interested in the analysis of electrophysiologically recorded brain activity. This small-scale implementation, with partial secure, ontological and intelligence features, was aimed as testbed within a single community in absence of inter-broker communication.

The OAN-aware web pages were embedded in the community portal managed by a generic content management system (the Plone CMS[iii]). Through the use of web styling techniques, the OAN content could be visually integrated and transparently presented to the users like any other page stored in the CMS. Members of the community have contributed several scientific resources (Fig. 2a). Some are standard methods and some are innovative methods contributed and presented to the community scrutiny. Service consumers can

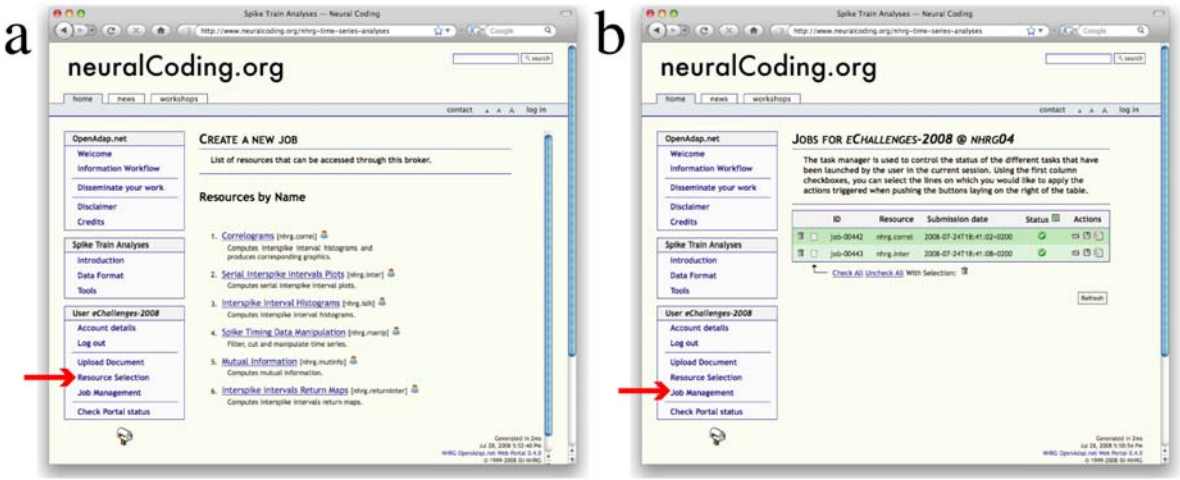

Figure 2: ***a.*** *Resources Currently Available Through the NeuralCoding.org Web Portal;*
b. *The Job Management Page Allows the Inspection of the Submitted Tasks*

upload their own piece of information to process and submit it in a transparent way to the broker for distributed processing, which appear as queued jobs (Figure 2b).

In addition to the resources provided to service consumers, *tools* can extend the portal features in an unlimited way. The commonest tools may appear as action icons next to the appropriate documents, including user data and analysis results (see the "Actions" tag in the job management application, Fig. 2b). In the standard OAN web-portal two such actions are built-in: the "download" tool instructs the web browser to download the corresponding content when activated by the user and the "browser default action" leverages the ability of the browser to associate content found on the web with specific applications installed on the user computer, e.g. opening spreadsheet documents with OpenOffice[iv].

A community web portal is usually the preferred site to gather community-oriented tools representing content-specific add-ons contributed by Third Parties that can be validated, installed and activated in the OAN-portal application under the supervision of the portal manager (Fig. 3a). Tools are short scripts executed on the portal application server. According to the type of output produced and the way the output is presented, three types of tools have been defined. *Standalone tools* produce content in any format (e.g., enabling the launch of external applications using techniques such as Java Web Start) but *embedded tools* and *modal tools* produce exclusively XHTML output. Embedded tools are displayed inline like any other page of the portal, while modal tools are presented in a dialog box more adequate for short, loosely formatted outputs.

Figure 3b shows an example of a Standalone tool that launches a Third Party application for scientific graphics, named NHRG XY-Viewer. This application illustrates also the power of OpenAdap.net for collaborative working environments [7-9], letting a "OAN-aware application" to communicate directly with a OAN broker. Through the appropriate menu and provided the required authorizations, users can list the available jobs located remotely in a collaborative space, browse their contents and open valid documents directly from the broker, without going through the web portal, as the menu is dynamically updated to reflect the changes in job status. Any Third Party software editor willing to extend its desktop applications with such OAN features can get the information from the Authors.

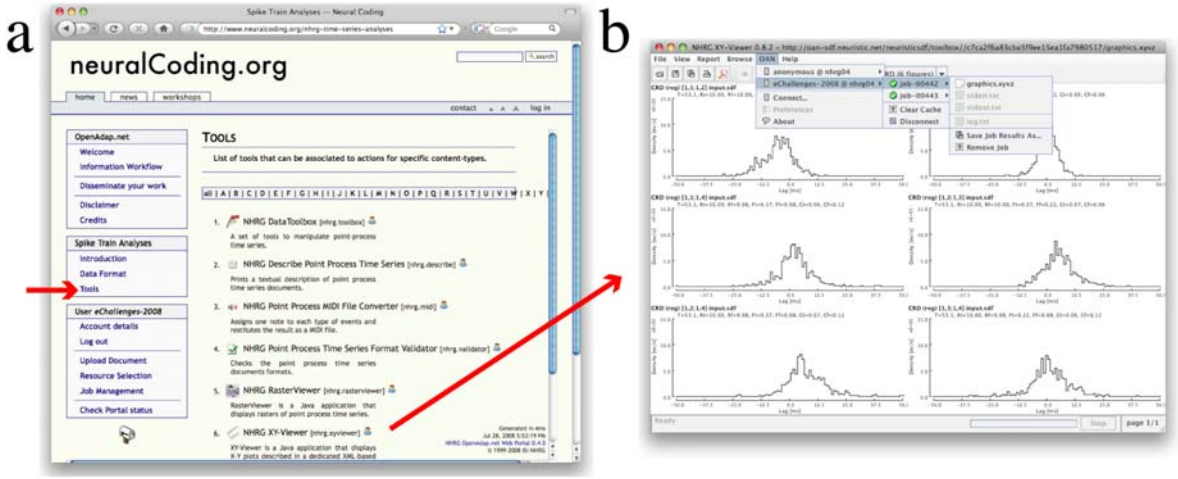

Figure 3: Third parties can enhance the user experience by providing tools featuring innovative access to the content (visualization, analysis, …). ***a.*** *List of tools currently available from the neuralcoding.org community portal;* ***b.*** *Example of an OAN-aware tool plotting scientific results and accessing the broker directly from the application menu (compare menu contents with Figure 2b jobs list).*

6. Business Benefits

The business model of OpenAdap.net includes an explicit no profit business model, based on the adoption of the OpenAdap.net middleware and the development of the brokers

thanks to a Free Software and patent-free policy. This model is aimed at the widespread dissemination of knowledge without economical barriers, e.g. at no cost to the communities that will rely on this system, and contains no profit motivation whatsoever. A complementary business plan is rooted on the potential benefits that appear with the growth of the adoption of the middleware. It takes into account many of the traditional business models that are associated to the Web [10], in particular the Community, the Brokerage and the Affiliate models. Thus, our overall business model opens the way to unlimited Third Parties and added-value services and carries an embedded advantage for competitive businesses. Such dual business plan is not new and several success stories are associated to members of the Free Software Foundation[v] who developed money-making businesses, e.g. the Neda Communications Inc.[vi] in the field of the communication protocols.

It is important to emphasize that the architecture of OpenAdap.net is not bound to a specific Community: it constitutes a global virtualization tool offering new opportunities to SMEs to improve their efficiency and strengthen their competitiveness independently from their domain. Major benefits may be expected by the adoption of the platform through the development of a mobile extension characterized by its pervasive and ubiquitous features. Data will be generated in large sets through sensors – temperature, light, pressure, etc – or high added-value information could be collected by human experts (e.g. birdwatchers analyzing in real time species distributions in the wilderness, automotive traffic monitoring on motorway by moving patrols to determine the coming of a traffic jam, etc.). For example, OAN-aware applications running on car-mounted devices could act as community members. By using the mobile telephone network each member would provide information, such as individual car's cinematic data and GPS coordinates, that could be processed as knowledged data about traffic conditions and fed-back to improve the safety of driving.

For cost-effectiveness, the data will be transferred over wireless telephone or computer networks depending on their availability. Using specific mobile features, users will be asynchronously notified as soon as tasks will be completed thanks to a widget permanently running on the mobile screen foreground in a way much akin of the push technology [11,12] (e.g., “Mobile me” implementation on Apple iPhone). Results will be available for visualization locally on the same mobile device and/or remotely on a desktop computer using a desktop OAN-aware application authenticated on the broker where the results will be stored and accessible for the user’s authorised group of co-workers.

7. Conclusions

The OpenAdap.net infrastructure makes possible the dissemination of resources and their exposure to application and evaluation across domains in ways that might not be anticipated. For example, processing tools from physics could permeate studies oriented to the dynamics of social interactions, linguistic analyses, crops forecast, traffic congestions, and life sciences. The OpenAdap.net Project itself is based on the collaboration between computer scientists, telecommunication engineers and neuroscientists having very specialized backgrounds. We feel that such a transdisciplinary approach is necessary for the achievement of real advances in the Information Society Technologies. The ability to tackle problems relies on both the past experience and new skills adopted by an individual. This is the feature of the trial-and-error paradigm and characterizes the procedural (“how”) approach vs. the declarative (“what”) approach. In our view the “know what”, i.e. the explicit knowledge, is basically determined by supervised rules and relies on the accessibility to a good teacher. Conversely, the “know how” is mainly determined by a self-experience (trial-and-error) and is keen to represent an appetitive way of information processing, particulalrly well suited for a fully-fledged implementation within a Community. Citizens of less favoured countries will gain access to all shared OpenAdap.net resources with a basic Internet connection, thus benefiting from the knowledge transfer and

shared infrastructures, and contributing back to the community with their own approaches and resources [13]. The outcome of such cross-cultural interactions is unpredictable and is likely to bring a full spectrum of challenges and opportunities to OpenAdap.net end-users and contributors throughout the world, especially for SMEs able to develop innovative business plans.

The development of pervasive tools represent an interesting aspect of the OpenAdap.net middleware for the less favoured countries because they produce an increasing amount of information which processing is delayed by limited access to expensive added-value processing applications. Field data gathering is a sample use-case where mobility can be of special interest for end-users. An embedded mobile application exploiting the OpenAdap.net middleware will be implemented to allow data collection on a portable device and transparent submission to a broker over an authenticated connection. Some of the main advantages provided by these mobile OAN-aware applications will be the use of advanced functionalities featured by mobile devices compared to traditional notepads including (i) connectivity, i.e. real time data gathering and submission, access to online databases, etc; (ii) possibility of taking pictures/videos/sounds and associate them with the collected field data; (iii) location-awareness provided by the mobile telephone network or an optionally integrated GPS device.

References

[1] Hattori, F., Ohguro, T., Yokoo, M., Matsubara, S., and Yoshida, S. (1999). Socialware: Multiagent systems for supporting network communities. Comm. ACM, 42(3):55–61.

[2] Nardeli, E., Posadziejewski, S., and Talamo, M. (2003). Certification and security in e-services, from e-government to e-business. Kluwer Academic Publishers.

[3] Ghernaouti-Hélie S. (2006) Sécurité Informatique et Réseaux, Dunod/Paris, pp. 368.

[4] Laufer, K. (2005). A hike through post-ejb j2ee web application architecture. Computing in Science and Engineering, 7(5):808.

[5] Zhang, X., Chen, S., and Sandhu, R. (2005). Enhancing data authenticity and integrity in P2P systems. IEEE Internet computing, 9(6):429.

[6] Villa, A. E. P., Tetko, I. V., and Iglesias, J. (2001). Computer assisted neurophysiological analysis of cell assemblies. Neurocomputing, 38-40:1025–1030.

[7] Cooper M. (2000) The challenge of practical work in an eUniversity-real, virtual and remote experiments, in Proc. of the Information Society Technologies Conference, 34-40.

[8] Callaghan, M.J., Harkin, J., McColgan, E., McGinnity, T.M. and Maguire, L.P. (2007). Client-server architecture for collaborative remote experimentation, Journal of Network and Computer Applications 30:1295-1308.

[9] Skopik, F., Truong, H.L., and Dustdar S. (2008). Current and Future Technologies for Collaborative Working Environments, ESA ITT Number AO/3-12280/07/NL/CB, May.

[10] Rappa, M.A. (2004). The utility business model and the future of computing services, IBM Systems Journal 43: 32-42.

[11] Franklin, M.J. and Zdonik, S.B. (1998). 'Data In Your Face': Push Technology in Perspective, Proc. ACM SIGMOD Int'l Conf. Management of Data, ACM Press, 516-519.

[12] Mühl, G., Ulbrich, A., Herrmann, K. and Weis, T. (2004). Disseminating Information to Mobile Clients Using Publish-Subscribe, IEEE Internet Computing, 8(3):46-53, May/June.

[13] Fahrnberger, H. (2006). Open source architecture for Africa – A bottom-up approach to innovation building in the context of construction in rural Africa. In IST-Africa 2006 Conference Proceedings. IIMC International Information Management Corporation.

[i] http://www.nessi-europe.com/Nessi/

[ii] http://www.NeuralCoding.org/

[iii] http://plone.org/

[iv] http://www.openoffice.org/

[v] http://www.fsf.org/

[vi] http://www.neda.com/