Extending existing applications functionality through OpenAdap.net

Javier Iglesias, Alessandro E.P. Villa

Information Systems Institute, Université de Lausanne, Switzerland,
{javier.iglesias,alessandro.villa}@unil.ch

Abstract

OpenAdap.net (OAN) is an information system aimed at knowledge dissemination and processing for sustaining communities of users who share the same knowledge representation within their field of interest. Such information system is not only the technology a community uses, but also the way in which the members of the community interact with the technology and the way in which the technology processes knowledge within the community. OpenAdap.net provides multiple business opportunities through the introduction of an innovative distributed working environment and creative user collaborations. In the framework, OAN-aware applications are established software pieces, open or closed sourced, developed to their own independent goals, which are extended to access the shared resources through the use of OAN-specific code.

Keywords

Free software, knowledge sharing, virtualization, distributed operating system

1 Introduction

OpenAdap.net (OAN) is an information system aimed at knowledge dissemination and processing for sustaining communities of users who share the same knowledge representation within their field of interest. Such information system is not only the technology a community uses, but also the way in which the members of the community interact with the technology and the way in which the technology processes knowledge within the community. Information systems are distinct from information technology (IT) in that an information system has an information technology component that interacts with the processes components.

The key component of OAN is an Open Source framework that allows resource sharing including content, applications, databases, computational power, disk space, services, Grids, etc tailored by the community members to tackle specific aspects of their topic of interest. Moreover, OpenAdap.net is aimed at virtually connecting communities that use the same knowledge representation framework (e.g., multivariate time series, pictures, sound files, etc.) in order to open innovative bridges between semantically close but unrelated communities emerging in the Information Society. Within this network of interconnected communities, it becomes possible for a community interested in, say, drug design, to gain a transparent access to analysis methods developed for road traffic forecast, thus breaking the boundaries created, among other reasons, by the fact that these communities don't publish their knowledge in the same media.

2 Relation to Existing Theories and Work

At this time, four architectures are used in this field: Grid, Web-services (WS), Peer-to-Peer networks (P2P) and Cloud computing. The peculiar strongholds of these architectures are briefly described here and synthesized in Table 1.

Grid: "Each user has a large dataset to manipulate with one application distributed on a set of computers." The problem addressed by the Grid is to distribute the processing of large data sets. The required application is copied to distinct computers over a network with each one treating a portion of the data. The results are recomposed from all the partial results at the end of the treatment. End-users have control on both data and applications, but little information on remote execution hosts.

Web Services: "Many users exploit the same services permanently provided through a centralized authority." Web Services provide a secured and controlled access to applications, usually to collaborators in the same company or institution. The goal is to provide distributed access to the same references, databases or applications. The system is articulated around a repository where the service interfaces are published in a self-contained form. The architecture is rather static, as services seldom change and are expected to be permanently available. End-users have no control over the applications and little information on remote execution hosts.

P2P networks: "Many users exchanging pieces of data in an unsupervised way." P2P (peer-to-peer) systems address the data-sharing problem. Copies of the applications installed on end-users' computers keep open connections from one computer to peers', forwarding queries and results back and forth until a match is found somewhere on the network. The architecture is open and competing implementations coexist on a self-organized network. End-users have control over their data and information on the peer hosts. It is interesting to note that end-users tolerate incredibly poor service quality and that this architecture raises social as well as technical issues.

Cloud computing: "An organization balancing computational load and data storage on-the-fly on (owned or rented) distributed assets." Dynamic scaling is the primary issue addressed by Cloud computing. By temporarily colonizing serviceable infrastructures, providers adapt the amount of resources allocated to services to follow their clients changing needs. As the main benefit from sharing computing power is the reduction in IT costs, this architecture mainly solves providers' technical issues in a transparent way for the end user.

	Data treatment distribution	Hardware resource allocation	Hidden execution hosts	Application sharing	Published application interface	Data sharing	Highly dynamic system	Transparent user / resource connection
Grid	×	×	×					
WS			×	×	×	×		
P2P						×	×	×
Cloud		×	×			×	×	×
OAN	×	×	×	×	×	×	×	×

Table 1: Comparison of OpenAdap.net with the other approaches over Internet

OpenAdap.net (OAN) falls somewhere between these architectures exploiting several of their interesting aspects, but with the intention to address a two-way problem: To provide to a community of users i) means to interchange their resources in an open, dynamic and secure way; and ii) the access to the solutions contributed by users belonging to other communities. End-users have control over their data, but do not need to manage the resources, nor do they have complete information on remote execution hosts. Collaboration within the OAN network allows the dynamic integration of these resources, possibly yielding new or previously unexpected composite resources. This can be summarized as follows: "Many users interchanging resources (data, applications, knowledge…) provided by dynamically interconnected domain-oriented brokers."

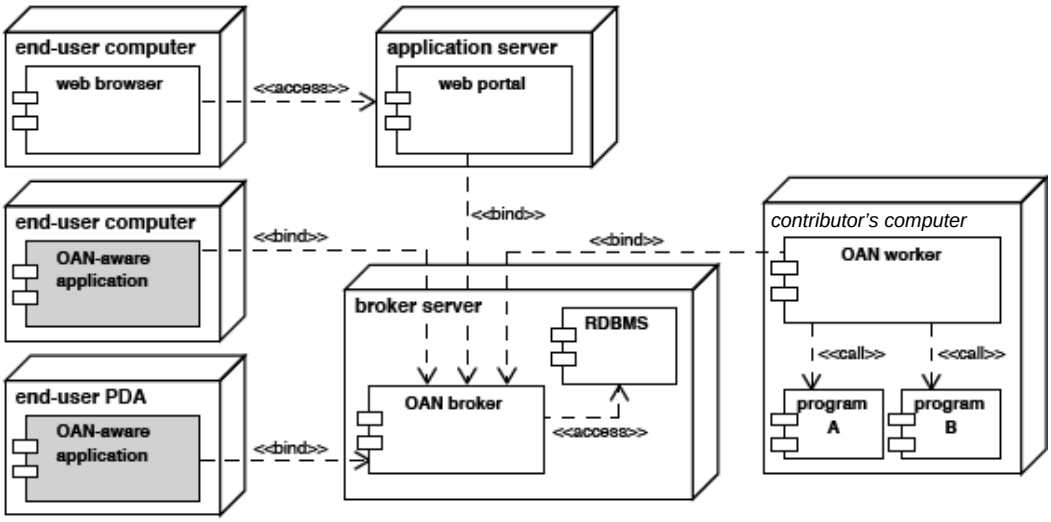

Figure 1: Deployment diagram for the OpenAdap.net components. Boxes represent computers, rectangles represent processes, and arrows represent inter-process communications over Internet.

3 Research Approach

Brokers, workers and OAN-aware applications are the three types of components in the framework. They are loosely coupled, distributed components that asynchronously communicate through a message-oriented middleware (MOM) [Mühl et al., 2004].

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.

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.

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 end-user through the necessary steps.

In line with the European Union NESSI initiative[1], almost "commoditised" Open Source software pieces and Open Standards implementations are integrated and extended into the middleware stack at the core of OpenAdap.net around a virtual distributed operating system. The objective is to reduce vendor lock in and obtain an open and secure platform at low development and maintenance cost.

End-users of OpenAdap.net can be consumers and/or contributors of the resources. By sharing their resources with a community, contributors might be looking for more visibility for their expertise, as they virtually connect to a broad audience, unlimited by geographical, timing and semantic borders. But contributors could as well be looking for a source of income or leveraging an IT infrastructure to reduce its costs.

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

[1] http://www.nessi-europe.com/

Software license 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 [13], and contains no profit motivation whatsoever.

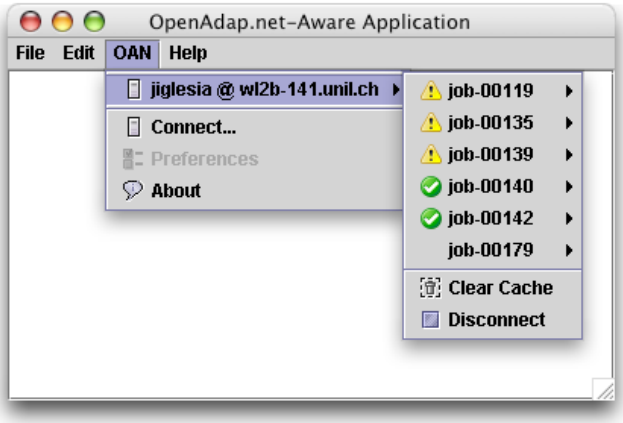

Figure 2**:** Details of the OAN-aware menu providing access to jobs stored on a distant broker.

Different business models are envisioned to govern the sharing of resources, including free, barter and pay-per-use models. In this respect, OpenAdap.net provides multiple business opportunities through the introduction of a new distributed working environment and creative user collaborations. The complementary business plan is rooted on the potential benefits that appear with the growth of the adoption of the OAN platform. It takes into account many of the traditional business models that are associated to the Web [Rappa, 2004], 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 a dual business plan is not new and several success stories are associated to members of the Free Software Foundation who developed money-making businesses, e.g. the Neda Communications Inc. in the field of 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. In particular, 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.

4 Findings

As seen above (Figure 1), end-user access is made through two types of software: OAN applications and OAN-aware applications.

OAN applications are specifically developed in the frame of the OpenAdap.net project to sustain the communications and social aspects of the communities. They take several forms like server-side tiers, mobile applications, command-line tools and dedicated community web portals.

OAN-aware applications, on the other hand, are established software pieces, open or closed sourced, developed to their own independent goals, which are given access to the OpenAdap.net framework through the use of OAN-specific code. In a way, OAN-aware applications extend their functionality by the addition of an access to the distributed resources shared through a closed or corporate community (e.g. with a monthly fee) or through an unbound network of OpenAdap.net communities. Other usages involve the creation and the submission of queries

(jobs) to a shared community resource, from within an OAN-aware application, either explicitly with an end-user interface, or implicitly through a programmatic process, transparent to the end-user.

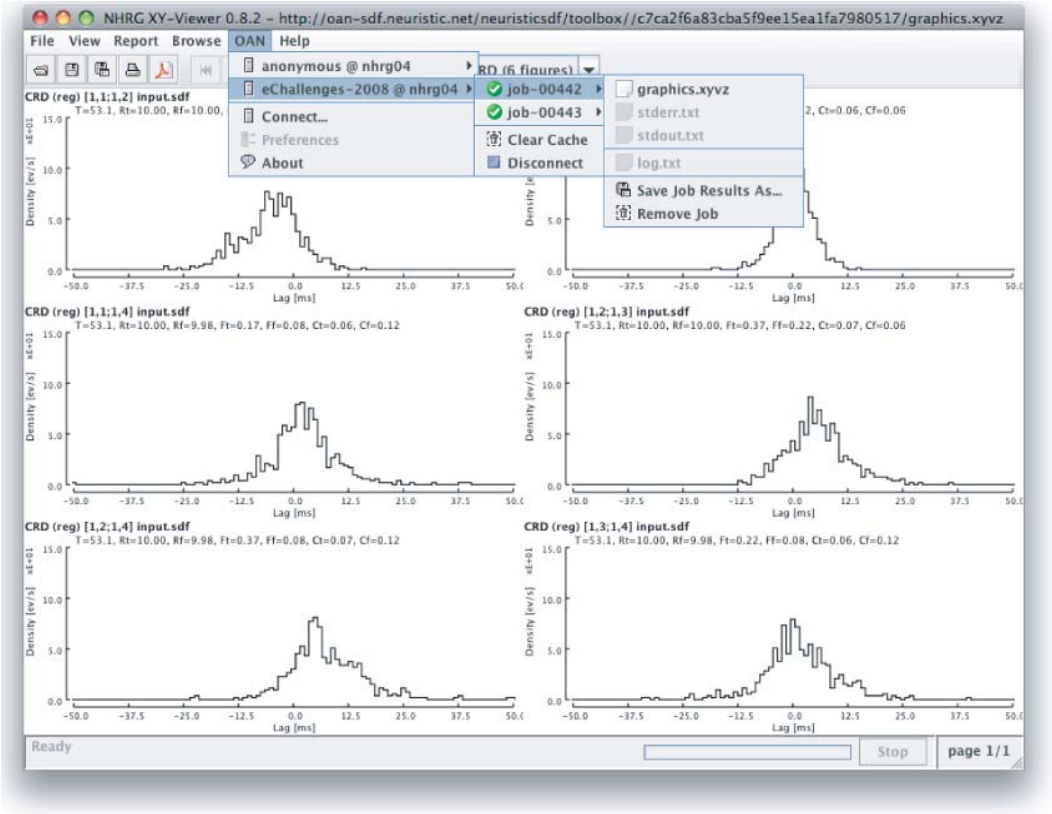

Figure 3 – Example of the OAN-aware menu embedded in a third-party application. The community resources extend the standalone application features.

This functionality extension is instantly updated with the latest contributions and comes with both a substantial benefit for the OAN-aware applications users and e-Business opportunities for the software developers, through innovative models yet to be uncovered by the emergence of a world-wide information space working environment.

From a technical point of view, the challenge of distributed interoperability is tackled by the OpenAdap.net framework with the use of open, standardized technologies (including the extensive use of XML, in particular) backed by Open Source software packages. In the current implementation, a comprehensive Java library, distributed under the terms of the GNU Lesser General Public License (LGPL) license, is available for inclusion into existing host applications.

For the OAN-aware application end-user, the extension takes the form of a single internationalized graphical user interface menu item labelled "OAN" (Figure 2) which contains all the functionality required to connect and authenticate on the OpenAdap.net framework, possibly on multiple brokers and/or with multiple identities. Once identified, the end-user can browse on-the-fly generated menu content to get access to up-to-date information related to jobs and results. In Figure 3, a sample OAN-aware application is presented: a general purpose scientific graphical data viewer extended through the addition of an Open Source library to provide access to the status of an identified OpenAdap.net user's jobs list.

The interaction with the OAN-aware application is performed through a single Java Interface, which defines one method to ask if the host application is able to process a certain type of

content (defined by its MIME content-type), plus one method requesting the host application to process a given content, whatever this processing actually is. All the bells and whistles required to manage the secured network connections and information handling is provided by the API implementation.

This design was optimized to minimize the amount of modification required on the host application. The use of the Java programming language is mainly justified by the interoperability requirements. Of course, other programming languages integration are envisioned (particularly C/C++ and Python), in addition to the current availability of standard interfaces between major programming languages and Java libraries.

It is to mention that other means of extending existing applications with OpenAdap.net capability have been considered and applied. These include (but are not limited to) direct programmatic access to the OpenAdap.net framework through a documented API; and mobile access.

For example, field data gathering is a sample use-case where a mobile OAN-aware application could be implemented to sustain data collection on a portable device. Some of the main advantages provided by such a mobile OAN-aware application would be the use of functionalities featured by mobile devices, including (i) Real time connectivity; (ii) Picture-tagging collected field data; (iii) Location-awareness provided by the telephone network and/or a GPS device. Data could be produced through sensors – temperature, pressure, etc – or high added-value information could be collected by experts (e.g. birdwatchers analyzing species distributions in the wilderness in real time, etc.). Analyses and results would be available for visualization on the same mobile device and/or remotely on a desktop computer using a desktop OAN-aware application. Data and results would be instantly accessible to the user's authorised group of co-workers, as well.

5 Conclusion

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, particularly 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 [Fahrnberger, 2006]. 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.

The addition of OAN-aware applications to the OpenAdap.net framework is sustained by a win-win principle: On the one hand, existing and legacy applications can be easily extended to access up-to-date and innovative contributions emerged from the Information Society, connecting to a vivid distributed working environment; On the other hand, OpenAdap.net communities experience can be enriched by the host applications user bases and existing functionality integration. At the same time, by keeping the technical footprint as small as possible, this approach to extending existing applications creates business opportunities offered by the OpenAdap.net consortium.

References

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.

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.

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.

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.

Hattori, F., Ohguro, T., Yokoo, M., Matsubara, S., and Yoshida, S. (1999). Socialware: Multiagent systems for supporting network communities. Comm. ACM, Vol 42, No 3, pp. 55–61.

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

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.

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

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

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.

Villa A.E.P., Iglesias J. (2008) OpenAdap.net: Dynamical and Open Source Approach to Knowledge Sharing. In: P. Cunningham and M. Cunningham (Eds): Collaboration and the Knowledge Economy: Issues, Applications, Case Studies. Amsterdam, IOS Press.

Villa A.E.P., Iglesias J. (2007) OpenAdap.net: Evolvable Information Processing Environment. Lecture Notes in Artificial Intelligence, Vol 4578, pp. 227-236.

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

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