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

Identity Architectures for Innovative Technology Enhanced Learning

Jose A. ACCINO[1], Manuel CEBRIAN[2], Victoriano GIRALT[2]
[1]*Central Computing Facility University of Malaga, Blvd. Louis Pasteur 33, Malaga, E-29071, Spain*
Tel: +34 95 2132366, Fax: +34 95 95 2131492, Email: accino@uma.es
[2]*Faculty of Educational Sciences University of Malaga, Blvd. Louis Pasteur 33, Malaga, E-29071, Spain, Email: mcebrian@uma.es*

Abstract: Integrated learning environment are common ground at most universities. However, the architectures in use for this are starting to show some of their limitations to produce really innovative learning models, especially when the user experience of these tools is confronted by users daily use of the network. This paper explores the pertinence of transforming these platforms into a more *permeable* user centred environment with fuzzy limits instead of the common tool centric model. This paper shows some proof of concepts that are being tested in production in the Ágora-Virtual learning environment, in order to extend its interoperability with special focus on OKI-OSID interfaces and identity technologies. These developments can be applied to any other services and applications.

1. Introduction

In the present paper, we will discuss what seesm to be a new way of looking at learning environments. Some experts in the field are starting to find that learning environments in use are not going further than a copycat of the physical classrooms on the net, and that some kind of paradigm shift is really needed for the teaching techniques that use the net as a way of delivering knowledge to the learners that most probably are going to follow a self paced self directed path for picking what is being delivered.

So, we are going to present our way of achieving an identity centric learning environment using modern technologies for identity federation that allow us to leverage a paradigm shift. This paper is relevant both to technical IT people supporting learning environments and the education experts that use them to get a head start of a proposed new paradigm.

As the reader most probably already knows, integrated learning environments are commonplace in universities, and most institutions have, at least, one of them in use as a support tool for presence learning. However, this has not resulted, as initially expected, in many changes in the learning models. It can be said that in most cases their deployment has just resulted in the extension of a physical space – the classroom – into a sort of virtual annex – the platform – where the same teaching and learning practices are used.

There could be many factors contributing to that situation. In our view, there is one that can be easily grasped: the difference in user experience in this environments and the much richer and integrated one experienced in the daily use of the network, even though the way the user perceives this tools, is key to the learning development.

The network expands its reach daily and users start to use it at increasingly earlier ages, which means that most university students do not arrive as *clean sheets*, but they have a previous technological background, that has often been self-taught. Which is not to say that it is less important for their later learning and that allows for comparisons and evaluation of

the environment they will be presented with. This will result in a level of satisfaction that will influence their learning results.

It is more frequent that students and, even, a significant number of the university personnel have their own free e-mail accounts and they use them in preference to the institutional ones, as they also prefer using their usual instant messaging tools, that fulfil most their communication needs. Users want to easily and seamlessly share their resources as they do in their daily experience of the network with their bookmarks, images or documents. Thus, what is *inside* the academic environment – even more so if it is virtual – should not be placed aside of the rest of the world. User do not want to be confronted to multiple authentication processes, diverse passwords, application specific search tools, or communication tools different from the ones they are used to use.

Using a spatial metaphor, the network experience is not that of a flat divided in rooms with doors that have to be crossed to leave a room before entering another one. It is rather an open loft with differentiated areas with free circulation without losing sight of the whole when using one of them.

The use of an online learning platform, under these circumstances, becomes another compulsory task that extends the separation between personal and academic contexts into the network, thus reducing the possibility of a really user centric learning. So, we wonder if labelling some platforms as more or less constructivist (fashionable term in pedagogical papers) than others, as real daily practice is not different from traditional presence learning models.

2. Design alternatives: application centric versus user centric

The process for integrating the user experience outside the learning environment in order to enrich it has both wide pedagogical implications as design and technology election ones. The most commonly used alternative is what we refer to as application centric design. We could define the paradigm for this methodology as “trying to enrich the user experience providing the platform with every feature that could be thought of and then some” (a.k.a. Kitchen sink syndrome). This paradigm has close links to a desire of the given platform becoming dominant in the market as the best *solution* to the integration problem. Thus, learning environments develop into specific universes with their own access rules, authorization, resource management, communications – chat, mail – and so on, which has important shortcomings.

The reader has probably been confronted sometime with the problems associated to the deployment of one of these environments inside an institutional infrastructure, like the ones that derive from the integration of other applications inside said environment or the ones related to student management: the former usually require an application rewrite to some extent (as happens with OSP in Sakai) or the use of some concocted mechanism that result in a n integration that is more perceived than real (e.g. LAMS in Moodle); the later have no other solution that devising some mechanisms that keep the diverse databases in synch or multiply the data entry processes.

The concept of a “dominant platform”, due to the size of the scene, is neither feasible nor desirable, as it is neither viable, not even in the medium term, an unlimited growth of components and modules that replicate already existing functions from other tools.

The second alternative, that we designate as *user centric design* is focused, on the other hand, on making platforms more permeable, on transforming them into a fuzzy delimited environment, like the network experience is, without hampering security. This means wondering which are the most adequate architectures when the desired result is placing the user in the middle of his diverse experiences, in all, it is the same as talking about identity centred architectures and applications that collaborate amongst themselves.

Interoperability is key: “Interoperability is the degree to which a provider and consumer can successfully interface having never met” [1] or, rephrasing in more modest terms, collaboration between applications. The technologies required to achieve this goal are well known: APIs, standards as OKI-OSID, identity management, …

The selection of technologies for the work presented here have been made on the basis of simplicity, that being that their deployment does not require a significant amount of technical expertise in not too extended technologies, this meaning that most of the presented architectures are very easy to deploy on top of widespread infrastructural software like Apache and PHP, require few other things, apart from the obviously existing services in any academic environments like universities (student registration and suchlike). Seamless integration of elearning environments into normal academic environments already in place for presence learning is key to an easy transition into a the new model that will be most probably be required in the coming years for many levels of education, specially in Europe with the implementation of the Bologna process guidelines.

Technologies that are difficult to the deploy, or require specially concocted procedures to get corporate information fed into them, face a serious risk of not being used due to many reasons, scarcity of trained experts and economies of scale not being minor ones.

3. *Agora Virtual*: an evolving architecture

We presented Agora Virtual as a *collaborative* platform in [2] giving the term a dual meaning, an environment for user collaboration and also a platform that works with other applications and services to minimize the need for any wheel reinvention.

This way of working includes, for example, an initial implementation of the authentication OSID, using Google Maps API for one of the modules or an external Jabber server as instant messaging server (jabberd2 at first, now Openfire [3]), but other modules are still in the old traditional format, like the Rubric one (fig. 2), developed as a way of experimenting the use of formative evaluation in big groups along the lines defined by EEES [4].

Once the platform has reached an adequate level of stability, after two years of intensive use in several projects and formative activities, the authors thought about future development for advancing in the above direction, and identified two models that are equivalent to the already described design alternatives; i.e., follow a tool centric model and start and endless race of gadget additions or, on the other hand, evaluate how to centre the current architecture around the user, working in two closely related areas: collaboration amongst applications and identity technologies.

4. Interoperability: OKI-OSID and the Harmoni framework Description

OKI project OSID (*Open Service Interface Definitions)* are a set of specifications that define how the different components of a software environment communicate with each other and with other systems [5]. Ágora Virtual has used its own implementation of the authentication OSID – and its required OSID like Shared – since its first version, thus the next step for increasing its interoperability is extending OSID to the rest of modules and functions.

The new architecture is service oriented, based on the Harmoni framework [6] developed by the Curricular Technologies Group if Middlebury College, for providing an OSID based infrastructure that eases the development and maintenance of educational environment applications. The Harmoni framework has two components: the Harmoni architecture and the services that include the OKI-OSID implementations (see figure 3).

Both components can be used together or by themselves, as services and OSID implementations are designed to work regardless of the applications structure, thus, they can be integrated into other architectures, as is the case with Ágora Virtual.

The Harmoni services offer, apart from PHP implementations of most OSID, added functionalities that can be accessed through very useful medium level service APIs, such as Database Manager (for building and executing safe SQL queries), Sets (for managing Id sets), Tagging (folksonomies API), Image Processor (image scaling and thumbnail management), GUI and Data Manager (for repository management).

All services can be included, configured and used as they are needed in the application, because, like OSID, their implementations are independent of the rest of services and most of them are not even linked to an specific environment (e.g.: they do not use PHP environment variables like $_REQUEST).

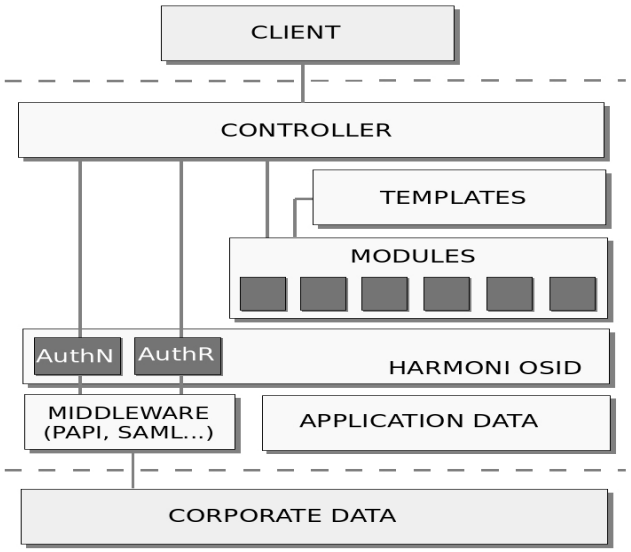

Figure 1: Agora Virtual architecture

This new architecture (see figure 1) has several advantages: application modules may access the Harmoni services API or the OSID as convenient; modules and services can be modified and reimplemented without influencing the other modules and services, as they are based on the interfaces and not on the implementations themselves; and, finally, decoupling the authentication and authorisation OSID from the rest of the application allows for an optimal integration of the platform to external middleware services. Actually, authentication is one of the less fortunate aspects of OSID version 2, and is being totally revised for version 3 – together with Agents -, this means that decoupling and connecting it to a service external to the implementation will ease adapting to the foreseeable changes.

5. Identity: OSID + phpPoA + SimpleSAML

Once we have decided to decouple the functional modules from authentication and authorisation mechanism, we need to select the applicable implementation model.

It should be stressed that the technologies we are presenting here can be used in many environments inside and outside education, as shown by the raising interest in identity federation technologies. Identity fragmentation is one of the most prominent issues that users are facing, they are the same person whichever service or application they are using and a sea of credentials for accessing them does contribute neither to usability nor to security. Thus, any technology that can be used to reduce this number of credentials, is worth the efforts even if they require a paradigm shift from the established status quo. Once

these principles have been laid out, we will present how we have applied them to evolve an existing elearning platform in use for several years into an identity centric learning environment.

Agora Virtual present architecture defines a single point of authentication and authorisation in the application front controller, and this eases the integration with external mechanisms. At first, we considered the possibility of using an OpenID server that would act as an identity provider (IdP) for which the authentication OSID would act as consumer. However, we discarded the solution due to two main reasons: OpenID present low security level [7] and, above all, the availability of a simplified but tested and versatile PHP version of PAPI [8] that offered great possibilities of integration to other identity management tools like SimpleSAML.

SimpleSAML [9] is a light PHP library based on Sun's OpenSSO Extensions (a.k.a. Lightbulb) that permits any service developed in this language to easily integrate into any SAML based identity management infrastructure. The most common way of deployment of a SAML 2.0 SP (Service Provider) is to use an interface written in the same language as the application, for easy communication between it and the SP (see figure 2).

Figure 2: SimpleSAML architecture

Final integration becomes easy thanks to OSID, phpPoA and SimpleSAML:

The authentication OSID acts as an interface between application and SAML 2.0 SP. Version 2 of this OSID defines the following methods;

authenticateUser(Type AuthenticationType)
destroyAuthentication()
destroyAuthenticationForType(Type AuthenticationType)
getAutenthicationTypes()
getUserId(Type AuthenticationType)
isUserAuthenticated(Type AuthenticationType)

However, versatility is one of the main advantages of OKI-OSID architecture, among many others it has: it is not necessary to implement each and every methods defined in one OSID, it depends on the deployment. In the present case, method isUserAuthenticated() just calls the PHP PAPI access point (an instance of the phpPoA class) that verifies the user is authenticated.

The phpPoA design requires a GPoA that communicates with the IdP (SimpleSAML in our case). We have a modified GPoA, called SimpleSAMLGPoA, that acts as a hybrid component that creates a encrypted assertion for the phpPoA and also acts as a SimpleSAML SP for the IdP to sent the attributes to.

The SimpleSAML IdP, on its part, has the possibility of using various plug-ins to verify user identities – LDAP, RADIUS, SQL - which result in great flexibility, The authentication architecture can be seen in figure 3.

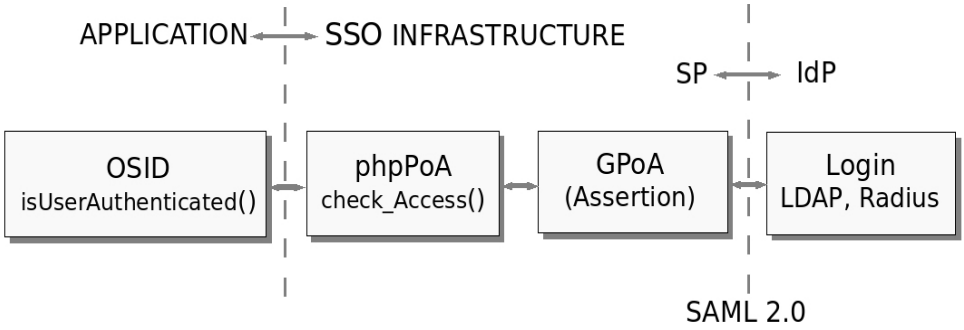

Figure 3: Authentication architecture

SimpleSAMLGPoA acts, then, as a connector between a PAPI and a SAML 2.0 environment. Other applications and services can be plugged into this connector, that will then share the authentication mechanisms.

The communications that happen between these modules for the initial authentication process can be seen, in a simplified form, in figure 4.

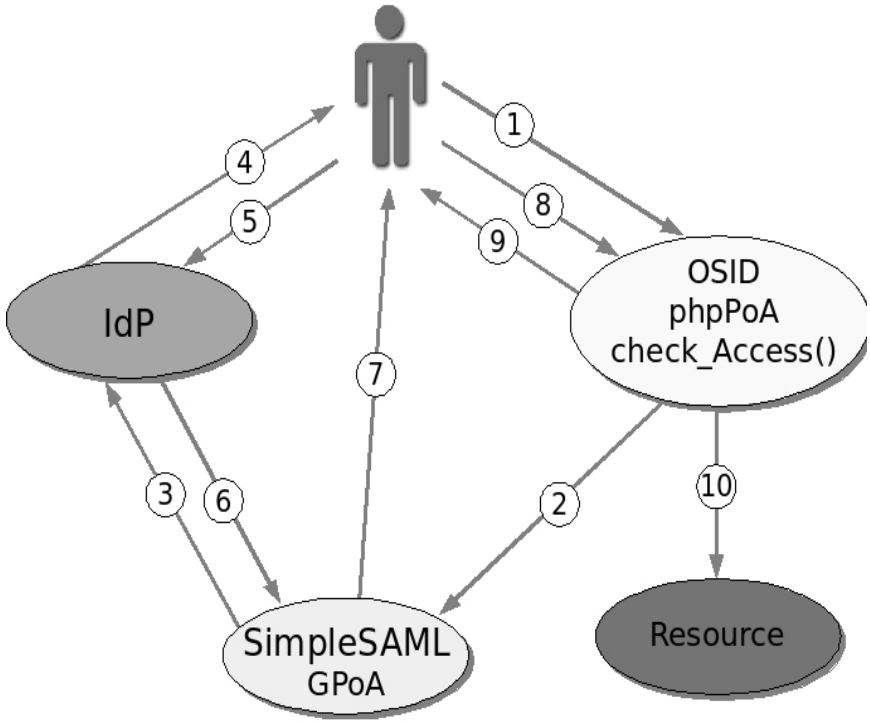

Figure 4: Authetication process

1. The user tries to access a protected area in the application, the request is intercepted by the authentication OSID that call phpPoA to check the user authentication.
2. phpPoA redirects SimpleSAMLGPoA for a PAPI assertion
3. SimpleSAMLGPoA redirects to SimpleSAML IdP for user authentication
4. The IdP presents the user the login form.
5. The user fills in the data
6. The Idp validates the data and sends attributes back to SimpleSAMLGPoA
7. SimpleSAMLGPoA builds a valid assertion and redirects the user back to the requested resource
8. The user accesses the resource again
9. phpPoA sends a cookie to the user
10. phpPoA allows access to the requested resource

Although it may seem complex, the whole process is transparent to the user, who only gets the login form - managed by the IdP and, thus, decoupled from the application – and, once validated, the requested resource. Subsequent requests will be authenticated thanks to the cookie, as usual in PAPI environments, and then by the SimpleSAML session.

The whole authentication infrastructure is hidden to the application behind the OSID interface. This kind of authentication does not require and implementation of the authenticateUser() method because the login process is delegated to the IdP and the authentication itself is delegated to the phpPoA, that will send the pertinent cookie, such as the whole process is external to the application that only knows about the result of the call to isUserAuthenticated(). The set application-phpPoA.GPoA is just one SP for IdP that communicates using SAML 2.0.

6. Results

The model presented thus far, offers various possible point for integration into an already existing infrastructure, offering several alternatives for use:

- In general, it is always possible to develop an specific authentication OSID
- The phpPoA based OSID can be directly connected to a GPoA in an already existing PAPI infrastructure.
- An alternative GPoA can be deployed and authenticated against any other mechanism, even HTTP Basic.
- It is possible to the deploy the full set against a SimpleSAML IdP, just selecting the proper plug-in for validating user credentials.
- Finally, if the service uses SAML 2.0, it is possible to provide the SimpleSAML IdP with a PAPI plug-in that performs the opposite process: integrate a SAML service into a PAPI infrastructure.

The user benefits from the advantages of a single authentication point that is shared with other applications that are compatible with Web SSO architectures and federation mechanisms: PAPI, Shibboleth, SAML 2.0, and others. This is the first step needed for deploying user centred services. Our elearning system serves as a demonstrator that this kind of technologies can be applied for using a new network mediated learning paradigm, but also that the underlying infrastructure can be used to access corporate data instead of digesting it into application specific format and, of course, using a common user identity for authentication and authorisation.

7. Business Benefits

The use of federated identity and standard interfaces that allow access to corporate data have proven as a corner stone for developing a new generation of applications that:

- Collaborate among themselves
- Are centred around the user
- Reduce de burden on the user: single authentication point and set of credentials
- Integrate corporate data into the learning environment
- Take the user experience outside the learning environment into account
- Use best of breed applications for each service
- Reduce the barrier to entry thanks to easier deployment

8. Conclusions

User centric learning, if it is to be really innovative, should be oriented towards a more holistic view of the user experience and, to this respect, daily network use is becoming more and more relevant. Thus, next generation applications, much more so learning

environments, should interoperate inside the new framework where the real platform is the network and that is centred around the user.

Collaborative – *groupware* type - applications are not enough for achieving the above objectives, they must become collaborate among themselves, in the way they show and share their resources, stating with the user identity. Adapting leraning environments development to this new context requires a two-pronged approach: identity management and derivatives (SSO, federations) and application interoperability standards like OKI-OSID, as a basic foundation for developing user centric environments.

Acknowledgements

The authors want to thank Diego R. López (RedIRIS) and Adam Franco (Middlebury College) for their help and contributions to develop the works we describe in the present paper.

References

[1] Coppeto, T.: Introduction To OSID V3 (for developers) http://plectrudis.mit.edu/okicommunity/filemgmt/visit.php?lid=89
[2] Accino, J.A.: ÁGORA VIRTUAL: Una propuesta de entorno colaborativo y de enseñanza sobre interfaces OSID http://www.rediris.es/rediris/boletin/76/enfoque1.pdf
[3] http://www.igniterealtime.org/projects/openfire/index.jsp
[4] Cebrián, M.; Accino, J.A.; Raposo, M.: Formative evaluation tools within ESHE: e-Portfolio and e-Rubric. EUNIS Conference. Grenoble, 2007.
http://www.eunis.org/events/congresses/eunis2007/CD/pdf/papers/p85.pdf
[5] http://www.okiproject.org/
[6] http://harmoni.sourceforge.net/
[7] Tsyrklevitch, E.: Single Sign-On for the Internet: A Security Store. https://www.blackhat.com/presentations/bh-usa-07/Tsyrklevich/Whitepaper/bh-usa-07-tsyrklevich-WP.pdf
[8] González, D.; Palacios, J.: phpPoA: Método simple de autorización mediante PAPI
http://www.rediris.es/rediris/boletin/74-75/ponencia11.pdf
[9] http://rnd.feide.no/category/simplesamlphp/
[10] Wilson, S: "Preparing for disruption: developing insitutional capability for decentralized education technologies", http://www.cetis.ac.uk/members/scott/resources/ed_media.doc
[11] "The future of web 2.0. An interview with WSU's Gary Brown", http://www.campustechnology.com/printarticle.aspx?id=58872