

SIDORA Project: Requirements for an open source Archaeological Information System

ENRIC TARTERA BIETO, ARES VIDAL AIXALÀ

Grup d'Investigació Prehistòrica (GIP), Universitat de Lleida, etartera@historia.udl.es, a Vidal@historia.udl.es

ABSTRACT

Inside the SIDORA project framework we pretend to create and to share, at least, one open source implementation of an Archaeological Information System. This implementation must be originated thanks to a public debate about archaeological informative needs and a proposal of standards. To succeed in that, it will be necessary to build an electronic community around the www.sidora.org site, organized in different thematic areas with forums, blogs, documentation and support, in order to achieve an organized knowledge and to spread the project results.

1. INTRODUCTION

Nowadays it is well known for everybody that the archaeological recording digitalisation is a consolidated phenomenon. For the moment being, computers can be used in any activity we can conceive. They are essential to manage the data originated by the fieldwork, to handle the record graphically and photographically, to reconstruct 3D architectures and artefacts, to make special and quantitative analysis, or any other imaginable task.

This is due to the impressive capacity to calculate of our devices, but over all, it is a consequence of the existence and the accessibility of the software needed. With good software near, life is better.

Usually, the technologies' improvement increases the tasks demanded to our computers and the expectations of what could they do for us. Actually, the progress of computer science has been spectacular, offering possibilities, for the time being, which were only science fiction fifteen years ago and now are pure routine.

The Grup d'Investigació Prehistòrica (GIP) with the collaboration of the Grup de Recerca Interacció Persona-Ordinador (GRIHO), both from the University of Lleida (Catalonia, Spain), have began the SIDORA project, thought to provide the archaeological community with an open source Archaeological Information System. This project began when our prehistoric research group started to feel immersed in the renovation process of his archaeological information system. Seeing the general panorama of new technologies, where the open source philosophy is taking a surprising and renewal role, we wondered if it could be possible to apply this philosophy and working methods to the computer applications for archaeology.

2. ARCHAEOLOGICAL INFORMATION SYSTEMS

An archaeological information system can be defined as a series of human and digital resources interrelated, set up to fulfil the information requirements for an organization with archaeological objectives. This definition, extremely generic, takes into account anything related with the archaeological domain and it also includes the human factor in the system. To have a well-trained team for the use of computer equipment and appropriate roles and protocols to enter the information, is essential to achieve any organization's aim.

Each research group or institution with management responsibilities has already put into gear an implementation of these systems according to his aims, methods, resources and technologies used, so the resultant panorama is, under our point of view, very unequal and difficult to analyze jointly and to extract some conclusions. The concept of Archaeological Information System is usually reduced to the systems concerning the archaeological recording, due to their great diffusion, so we will concentrate our analyse in this more restricted and uniform subgroup to do a modest approach to this reality.

Nowadays, concerning this framework there are some excellent projects in course. They are ambitious, consolidated and, some of them, commercial. We would like to mention some of them, although we are conscious that we take the risk of ignoring other projects as interesting as the examples. First of all, the Syslat, made by Michel Py for a project to study the ancient city of Lattara, in the south of France, (Py *et al.*, 1991; Py, 1997). Secondly, the work developed by the Landscape Archaeology Laboratory in the University of Santiago de Compostela (Spain) (González, 2002). And, finally, the SIDGEIPA project, from the University of Valencia (Spain).

But they are only a very little part of the work done and we wouldn't like to emphasize in these excellent projects which, under our point of view, do not represent the existing reality. To tell the truth, archaeologists keep on working in their own "hand-made" data systems, that is, a Database Management System, in addition to some commercial software to treat graphical documentation, and usually other tools to analyse the information (such as spreadsheets, and statistics or

GIS packages) and, finally, other programs to write and to publish. Sometimes, this system configuration carries out some problems due to the lack of integration of his components, having, in consequence, a detrimental effect on the information quality.

In this context, the greatest efforts have been pointed towards the database design and implementation. The reason is that the archaeological data models are very complex: they combine different kind of information (text, photography, design, etc.), and they have to allow complex relations between multiple entities, which have different levels of abstraction and subjectivity. The final results originated by this kind of design are the information systems based on the relational model, where the data are defined independently from the processes done with them.

As a consequence of this independency, typical in every database, the information's dynamic is usually underdeveloped, because the relational model only takes into account the static structure of data. In fact, programming must treat the dynamic nature of its processes, and the skills needed to perform these programming are far to be usual.

Normally, the other elements from the data system belong to generic uses or they are loaned from other activities. The most common situation is that they are proprietary software, whose commercial nature brings about a difficult to integrate them with the system. This is caused by the extended use of proprietary file formats and the closed nature of commercial software.

Any attempt to make complex tasks whit the data requires programming and, obviously, that fact increases a lot the necessary formation to create and to develop these tools, or, at least, it makes necessary a transdisciplinary collaboration.

3. LOOKING FOR BETTER AIS...

Under our point of view, although the use of computers in archaeology has been intense and extensive during the last twenty years, there are still lots of things to solve.

First of all, we find the paradoxical lack of standards the archaeological concepts and their meanings. And this is contradictory because the most part of the archaeological community is agree about the stratigraphical methods used, that is the Harris method, but, on the other hand, archaeologists still claim their idiosyncrasy using different names to define same things. The problem of standards is extremely complex and difficult to solve, because, perhaps, this idiosyncrasy is not detrimental for archaeology because it offers several ways to approach the past. The key idea would be to achieve respectful standards with the several theoretical positions and the conceptual particularities of everyone. So then, looking for standards is not looking for uniformity.

Otherwise, when we talk about the lack of well-implemented references, we are trying to explain that the community references are the paper forms, more than the computer models. And concerning these computer models, they have been designed and implemented independently and expert hands have not always made them. So, that fact leads us again to the key problem of standards.

On the other hand, the components of our systems don't interact as it would be desired to achieve a correct data representation and an effective interaction between human and computers. Concerning usability problems, many times the system functionalities are left hidden under a too cryptic and complex interface.

As far as accessibility is concerned, it is not always easy to access to a good software, sometimes due to its proprietary nature and sometimes because these implementations are made for an inner consume and it is impossible to access to them.

4. WHY AN OPEN SOURCE AIS?

Firstly, we think it is better just for a question of common sense. Nowadays, there are lots of technologies available to everybody, that will help us to liberate ourselves of legal and economical obstacles and they are complete and secure solutions. The technologies are numerous, they are ready to use in our implementations and there are lots of finished products available to everyone and free.

All these products and technologies are the result of reference projects made with volunteer work by thousands of people from all around the world. These people spend part of their time every day searching security holes and bugs in a code that they haven't written. Afterwards, they document and correct these problems, freely distributing their improvements to the community without receiving any economical reward for their work.

There are also ideological reasons to justify the choice of an open source software, and they are based on its social value. The diffusion of technology by the capitalist way increases the digital fracture between the globalizer and the globalized countries and it puts a barrier in the development of the least favoured communities in the world.

From our point of view, innovation is a desirable factor. And you just know that proprietary technology does not belong always to its legitimate developers, otherwise is usually the property of huge companies, which control the software creation business. For these proprietors innovation must follow an economical, and not technological, strategy.

A good open source implementation for an archaeological information system can help standards by the way of providing an implemented reference, open to any discussion and evolution. Ideally, research teams will more efficiently achieve their objectives if they can easily access to cutting-edge technologies and take advantage of a collaborative community.

5. THE SIDORA PROJECT

Inside the SIDORA project framework we pretend to create and to share, at least, one open source implementation of an Archaeological Information System. This implementation must be originated thanks to a public debate about archaeological informative needs and a proposal of standards. To succeed in that, it will be necessary to build an electronic community around the www.sidora.org site, organized in different thematic areas with forums, blogs, documentation and support, in order to achieve an organized knowledge and to spread the project results.

6. REQUIREMENTS FOR AN OPEN SOURCE AIS

Obviously, the Archaeological Information System's functionality is to manage archaeological information. And, at first, we do not want to renounce to any subject related to this information in our models. That is the reason why we are talking about archaeological recording in widest sense. However, we have to treat separately and specifically certain domains of the archaeological information, in order to make something applied and functional.

As examples of clear domains, we can quote the stratigraphical management and the fieldwork, the documentation of the remains, the recording of the materials or the stratigraphical diagrams. Each one of these subjects must have a specific treatment.

There are other aspects we want to solve by an open and accessible way. We want an information system open source, multiplatform, usable, with information quality and integrated.

6.1 OPEN SOURCE

At first our code will be open. As all you know, that means that everybody will be able to obtain this software free, both the archives and the own source code, which will be used, changed and redistributed by anyone. As the GNU website declare, free software is a matter of liberty, not price (www.gnu.org).

Open source is not just a legal contract between a user and a developer. Eric Raymond has described the difference between the proprietary and the open source development with the metaphor of the cathedral and the bazaar (Raymond 2000). The first one is institutionalised, monumental, directed and certainly opaque, while the second one is governed by an apparent chaos with multiples agents interacting.

6.2 MULTIPLATFORM

The product will be multiplatform in order to break accessibility barriers. In other words, Wintel, UNIX or Mac OS X computers will be able to run our software, because it is good to respect the freedom of the users choosing their working platform.

Nowadays we are working on Java technology, which allows us to carry out the maxim "write once, run everywhere". Java is simple to learn and powerful at the same time, it is object oriented, distributed, extremely robust and secure, multitasks and it is always in evolution. Some issues can be appointed about the legal license of the Java libraries that are not still open source but they are freely distributed.

6.3 USABLE

We pretend to make a product usable. As we said before, it does not worth to implement fantastic functionalities if they are going to be left hidden under a cryptic or incomprehensible interface.

6.4 INFORMATION QUALITY

We also want information quality. That means that both data models and processes will help the information to be precise, opportune, complete, significant, coherent and secure. All these information characteristics are not only related with design and implementation, also human factor is basic. But the computer can help us to increase the information quality, which is introduced with usable procedures, suggesting normalized alternatives, efficient representation methods and very detailed data models.

6.5 INTEGRATED

The integration between the different components of the system is an extremely difficult aim to achieve. It is not about offering a multipurpose tool; it is about presenting a toolbox whose components are interacting efficiently.

7. PROJECT DESIGN LAYOUT

In the SIDORA project we bet for simplicity, as the best guaranty to easily change or improve anything we detect that is not working well. That's the reason why we are working with a three-layer architecture: interface, logic and persistence. That is, we divide the system functionality in three independent layers.

The persistence layer takes care of saving and recovering information. The logic layer is concerned itself with information models and processes to do. And, finally, the interface layer has to show the information and has to interact with the user.

As for the persistence layer we are working on the relational data base management system, MySQL, because we don't know any open source and object oriented data base manager system. As we said before, the information models from the logic layer are implemented with Java, which is object-oriented. So, in the zone of contact between both layers we are working with the persistence engine Hibernate.

Concerning the interface, we are working on an architecture based on modules. Each module is thought to do a specific task, thus is possible to more explicitly delimit the needs of users and to better control the type of representation needed. At the same time, the functions to implement become simplified just to the most necessary.

In this way, the system is easily extensible, adaptable to each one necessities and it preserve the integrity in the logical and the persistence layers.

REFERENCES

GONZÁLEZ PÉREZ, C. (2002) – *Sistemas de Información para Arqueología*, BAR International Series 1015, Archaeopress.

PY, M. (1997) – Syslat 3.1, Manuel de référence, *Lattara 10*.

PY, M. *et al.* (1991) – Système d'enregistrement, de gestion et d'exploitation de la documentation issue des fouilles de Lattes, *Lattara 4*.

RAYMOND, E. S. (2002) – The cathedral and the bazaar, <http://www.catb.org/~esr/writings/cathedral-bazaar/cathedral-bazaar/>.