iAKS: A Proposal for a Web 2.0 Archaeological Knowledge Management System

Ethan Watrall, PhD
Matrix, the Center for Humane Arts, Letter, & Social Sciences
Online
Department of Telecommunication, Information Studies, and Media
Michigan State University
East Lansing, MI 48824 USA
watrall@msu.edu

Jeff Siarto
Department of Telecommunication, Information Studies, and Media
Michigan State University
East Lansing, MI 48824 USA
siartoje@msu.edu

Abstract
In this paper we propose a Web 2.0 Archaeological Knowledge Management System entitled iAKS. Developed using the forthcoming Adobe AIR (Adobe Integrated Runtime), iAKS is intended to be a flexible data entry, data archiving, and data visualization system appropriate for use in a wide variety of archaeological settings. Ultimately iAKS is designed to leverage the strengths of Adobe AIR and other web service technologies in order to bypass many of the problems associated traditional field archaeological data management and data interoperability.

Keywords
Archaeological Informatics, Web 2.0, Adobe AIR, Data Management, Data Visualization, Data Archiving, Interactive Media.

ACM Classification Keywords
H5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous.

Introduction
Methodologically speaking, field archaeology, the domain of professional archaeology that is actively
engaged in the excavation of sites, has been content with traditional practices of inquiry whose lineage can be traced back to the beginning of the century. However, as new generations enter the discipline, there has been a trend to accept a greater range of computational methodologies. As a result, techniques such as ground penetrating radar, magnetometry, and GIS (and the computational analysis of associated data) have become more commonplace. This willingness to accept new computational techniques into the discipline, however, has not generally been extended to systems that allow field archaeologists to digitally collect, archive, and access archaeological data. This is no great surprise as the type, amount, and sheer scale of data collected during any given archaeological project is staggering. In addition, the type of data and the way in which that data is collected can vary widely from one archaeological project to another.

While archaeologists generally agree upon what might be called methodological meta-standards, the discipline as a whole lacks any sense of standards for the kinds of data that is recorded and exactly how that data is recorded. This is not necessarily always the result of theoretical differences, but often the result of where archaeologists are working (geographically speaking) and what they are working on (both temporally and culturally speaking). The kinds of artifacts recovered from a Paleolithic site in France, for example, differ enormously from the kinds of artifacts recovered from the excavation of a historical 19th century farmstead in Nova Scotia. The result of this state of affairs is that most archaeological projects still rely on paper record keeping. Further, the longer an archaeological excavation has been running, the more paper is generated. As a result, it becomes more difficult for the project’s archaeologists to locate and analyze specific subsets of data. This problem is greatly intensified for archaeologists not associated with project itself who might be interested in the data being recovered. Not only are they unfamiliar with the specific system of data collection, data management, and data archiving techniques of the project, but the data itself (in the form of reams of paper) may be located in a box half way around the world.

This should not suggest that archaeological projects have not experimented with storing data in databases, quite the contrary. However, these databases are designed specifically for individual archaeological projects, and are, as a result, quite unique. Given this, there is very little chance that one data entry and archiving system could be adapted for another archaeological project. Further, the systems themselves are rarely, if ever, networked (and designed for robust and elegant user interaction). This means that both non-project archaeologists and project archaeologists who are not in the same location as the database wouldn’t be able to access the data. This is unfortunate, as the lack of standardized data and interoperability of data sets and data systems has made the kinds of “bigger picture” trend spotting which is the hallmark of both field archaeologists and theoretical archaeologists difficult.

The evolution of the web, and more specifically web services, offers an opportunity to solve some of these problems. It is within this context that we propose a Web 2.0 Archaeological Knowledge Management system dubbed iAKS (Interactive Archaeological Knowledge Management System). As outlined in this paper, iAKS is specifically designed to take advantage
of web services and Web 2.0 technologies in order to create a robust, scalable, stable, and re-usable system that solves many of the data collection, data archiving, and data analysis problems encountered by many archaeological projects. iAKS will feature a very flexible setup and install model that will allow archaeologists to customize the specific types of data that they might want to collect and archive. In addition, iAKS will have a variety of client/server model, making it an appropriate tool for a wide variety of archaeological settings ranging from a small-scale single site fieldschool to large multi-site excavations. Further, iAKS would feature a variety of connectivity models, making it an appropriate tool for projects that have regular network connectivity and those that do not. iAKS would also be designed with a keen sense of usability, thereby making it appropriate for a broad user base. Finally, iAKS will feature a robust data visualization system, allowing current (and future) archaeologists to browse and creatively visualize data. The ultimate goal of iAKS is to create an interactive system that archaeologists can use to not only to collect, archive, and analyze excavation and artifact data, but to access and visualize data remotely from anywhere in the world.

**Proposed iAKS Architecture**

The Web 2.0 movement has seen the development of a variety of useful and very powerful technologies that have helped the Internet evolve from a one-way information medium populated with walled off silos of information to a platform for building and deploying Rich Internet Applications. The iAKS project plans to harness this medium and its associated technologies to deliver an interactive, data-centered web and client-based application to help make data collection, data archiving, and analysis easier for archaeologists.

In early 2007, Adobe will be introducing a product named AIR (Adobe Integrated Runtime) to developers. AIR is a cross-operating system runtime that allows web developers to deploy applications to the desktop utilizing their existing skills (HTML, JavaScript, AJAX). By taking advantage of AIR, the iAKS client will be able to run on virtually any desktop PC, access the local file system and resources when a connection to the server can’t be established, and run as a traditional in-browser web application all from the same code base. This new technology is vital to the success of iAKS because it enables its operation under the wide variety of system and network conditions often found on archaeological projects. In addition to the flexibility of AIR, iAKS will also leverage the power of Adobe’s Flex platform. Flex, a rich Internet application framework based on Flash, will allow iAKS to present data in detailed, multi-colored graphs, charts, and tables giving the end user almost immediate feedback and analysis of their data.

Currently, a prototype UI can be seen at [http://iaks.cas.msu.edu](http://iaks.cas.msu.edu).

**iAKS Client/Server Models**

One of the main goals of the iAKS project is to create a system that can be used by archaeologists in a wide variety of settings. iAKS can be configured to operate efficiently in different situations ranging from small archaeological projects to large, multi-site excavations. The iAKS Manager (server), which is used to manage users, set up projects, store data, and publish findings can be set up using five different client-server models:
- **One Client/No Server Model.** This configuration leverages the power of AIR and allows the iAKS client to store all project information and data on the users local hard drive. This model is ideal for small projects or educational settings where minimal multi-user collaboration is needed.

- **One Client/One Server Model.** This is the most basic of the client/server models that incorporates the iAKS Manager system into the configuration. With the server installed, this model will allow the client to store data remotely, create multiple users for projects, and share data through web service APIs. Both the iAKS Manager and the client application will store project data in the same format making it easy to migrate from local storage to server storage.

- **Many Clients/One Server.** This client/server model allows iAKS to manage multiple client applications accessing the same set of project data. This facilitates multiple users being able to access the system through different clients from different locations. iAKS will manage each running client and sync to the project that user selects upon logging in to the system. This configuration also gives iAKS the ability to manage a pool of client applications, assigning roles and permissions for a particular project.

- **Many Clients/Many Projects/One Server Model** (figure 1). In this configuration, the iAKS manager tracks multiple projects each with multiple client applications. This is the most complex setup and would be used on a large project that might span multiple excavation sites and have many users accessing the data locally and from remote locations via the iAKS client.

Finally, because at its core iAKS is a web application, users will be able to run the client from within a web browser and have access to all their projects and data remotely. This can be a flexible option for teams that need remote access, or a starting point for an iAKS-based online service that offers a "lite" version of the full installation available only through the browser.

---

**figure 1.** iAKS Many Clients/Many Projects/One Server Model

**iAKS Setup Model**

One of the most daunting hurdles for designing a single knowledge management system usable and useful in a wide variety of archaeological settings is the lack of standardization among the types of data that different archaeologists collect. For example, when it comes to stone tools, one archaeologist may record the length,
width, thickness, and weight, and type of a tool. However, another archaeologist may collect these measurements and many others (such as platform width, platform thickness, number of dorsal scars, blank type, retouch type, etc.). As a result, a knowledge management system for field archaeology with hard coded methods of data entry is wholly useless in a wide variety of archaeological settings.

In an effort to solve this problem, iAKS will include a flexible setup model that will allow archaeological projects to configure exactly the types of data (and the associated methods of data entry) they would like to include in their iAKS system. For example, an excavation of an 18th century historic fort in England would have absolutely no need to collect and archive data on stone tools (because there wouldn’t be any). However, they might have the need to collect, archive, and analyze data on pottery. On the other hand, an excavation working on an early Neolithic site in the Egyptian Western Desert would have nowhere near the same kind of need to collect, archive, and visualize data on pottery. However, they would have a very pressing need to collect, archive, and visualize a vast array of data pertaining to stone tools. As a result, the knowledge management for the first project will look radically different from that of the second project.

In order to solve this problematic need for systems that vary radically from project to project, the core feature of the iAKS setup model will be a robust library of data types (and their associated data entry mechanisms). The library will be created based on consultations with different archaeologists working in a variety of areas (geographical, temporal, methodological) in order to get a deep sense of what kinds of data archaeologists collect, and how they might want to enter, archive, and visualize that data in iAKS.

During the setup process, which will be moderated by a user friendly wizard, the project archaeologists will be able to pick and choose the types of data (and associated data entry mechanisms) that are important to their excavation, all of which will be drawn from the aforementioned library. The result will be that the project archaeologists could directly customize exactly how their iAKS clients will be configured. During the setup process, archaeologists will also be able to take a modular approach to setup. For example, one archaeologist may include a “Ceramic Data Module” (and then customize it further with specific data types and associated data entry mechanisms), while choosing not to add the “Lithic Data Module” and the “Faunal Data Module” based on their excavation’s specific needs. This setup model will also configure the iAKS client interface itself, creating a custom user experience for each project.

In addition, a secondary goal of iAKS will be to make an API available so archaeological projects can extend the types of data (and the methods in which that data might be entered) included in the initial setup, thereby allowing archaeological projects to further configure and personalize their iAKS system.

It is very important to note that the setup model is identical regardless of the client/server model. The process by which a large archaeological project (with multiple sites) using the iAKS server will configure their iAKS clients will be exactly the same as the process by which an individual archaeologist working on a smaller scale (a grad student working on a small portion of an
archaeological project, for example). The only difference is that during the setup and configuration of projects using the iAKS Manager, the server will be connected during the setup and configuration, while the setup and configuration process for a single client install will be completely self-contained.

The setup and install process will create an XML file that will either sit on the server (if the archaeological project was using the client/server model) or the client (if the archaeological project was using the single client model). The XML file will essentially act as a configuration file for that project's iAKS interface. The result will mean that whenever a new iAKS client connects to the server of a project that has already been configured, the server will push the XML file to the client, configure the interface, thereby making it unnecessary to go through the lengthy setup process anew. In the case of a single client setup (where, as mentioned before, iAKS is setup on a single machine without the connection to a central server) the XML file can be exported and moved to another machine, thereby speeding a new iAKS installation.

Ultimately, the goal of the setup model is to create an installation and configuration process that is flexible enough to result in a system that is both usable and useful for the widest possible range of archaeological projects.

iAKS Distribution Model
The final, but no less important, aspect of iAKS is the manner in which it will be distributed. While all of those involved in the iAKS project are firm believers in the philosophy of the open source movement, releasing the iAKS client and iAKS Manager as open source software under a license such as the GNU GPL is impractical at best. The primary reason for this lies in the fact that the individuals who will be using the system would be archaeologists, and would most likely not have the necessary technical skills required to alter the software. In addition, iAKS is designed for such a specialized niche that it would most likely not garner the same developer community that is associated with other, more popular, open source software.

Given this, several distribution scenarios (none of which are mutually exclusive) are plausible:

- The iAKS client and iAKS Manager would be distributed freely and user hosted. In much the same way as the WordPress blogging system, an organization would be formed to host and distribute the software as well as build and host a user community.
- Free hosted service for those who might not have a server appropriate for mounting the iAKS Manager. A business model could be built around this scenario that offers upgraded services for a fee. This is not dissimilar to the business model of 37Signals or Blogger.
- The iAKS source code (both for the client and the iAKS Manager) could be licensed for those wanting to radically modify the system.

As mentioned before, the release on an iAKS API would facilitate the extension of the system without the need to license and edit the source code.

iAKS Online
In addition to the iAKS client and iAKS manager, the iAKS project will include a robust online community based site that will act as a central (and semi-public) repository into which iAKS users can upload the
archaeological data from their local iAKS installation. Once uploaded by individual iAKS users, other community members can either search and browse the data online or download and import it into their own local iAKS installation for standalone analysis or inclusion into an existing iAKS dataset. For the purposes of security, iAKS online will have a two-tiered system of access. The first tier will be open and accessible by the general public, and will feature data that is filtered by the original contributor. This way, sensitive data, such as site location or exact artifact provenience, can be hidden from the general public. The second tier will feature full and complete data sets, and will only be accessible by those professional archaeologists who have registered with the site and whose credentials have been verified.

The interoperability of data between an iAKS installations and iAKS online as well as between individual iAKS installations is made possible by the fact that all iAKS data is stored as XML. The structure and logic of the XML used to format and store iAKS data is based loosely on the ArchaeoML Schema developed by David Schloen and used by the University of Chicago OCHRE project. This is important as iAKS, while primarily a tool for knowledge management and data visualization, becomes a vehicle for the standardization of archaeological data. All those who employ iAKS (in any of its various capacities) will benefit greatly from their data being standardized.