Beyond The Trowel's Edge: Provenance-Based Collective Archaeological Interpretations

Michael O. Jewell
Electronics & Computer Science
University of Southampton
Southampton, SO17 1BJ
moj@ecs.soton.ac.uk

Tom Frankland
Faculty of Humanities
University of Southampton
Southampton, SO17 1BF
tf4e10@soton.ac.uk

Enrico Costanza
Electronics & Computer Science
University of Southampton
Southampton, SO17 1BJ
ec@ecs.soton.ac.uk

Graeme Earl
Faculty of Humanities
University of Southampton
Southampton, SO17 1BF
graeme.earl@soton.ac.uk

Luc Moreau
Electronics & Computer Science
University of Southampton
Southampton, SO17 1BJ
l.moreau@ecs.soton.ac.uk

ABSTRACT
Archaeologists interpret finds from the moment that they are recovered, but many of the notes and annotations created during excavation and analysis are discarded. We introduce PoN, or PATINA of Notes, an interpretation capturing web application informed by the work practices of archaeologists and integrated tightly with provenance. PoN preserves annotations, allows for their layering and grouping, and gives archaeologists the ability to visualise how their collective interpretation of artifacts has evolved from the early stages of excavation to the final report and beyond.

1. INTRODUCTION
In recent years, digital technologies, and in particular web-based technologies, have empowered and enhanced much research in the humanities. Archaeology in particular has benefited from these developments, with web-based tools offering new ways for archaeologists to collect, store, and analyse data and then disseminate their findings with both specialists and the wider public. Examples of these include archaeological recording systems, virtual research environments and semantic technologies. While this integration has presented many benefits to archaeological research there are still many problems, such as the large amounts of processing and curation of existing archaeological records and databases that is often necessary. For example, often archaeologists will make paper notes in the lab and field that are edited and annotated multiple times before they are even transcribed into digital form.

There have been recent attempts to introduce paperless recording systems at several archaeological sites, such as at Pompeii[3]. However while these interventions improve the accuracy and accessibility of records on site, they lack the infrastructure and/or mechanisms within the software to capture the processes that lead to the creation of the data. Furthermore, they do not at present show how data captured at an excavation can evolve as archaeological research progresses. We feel that this evolution is significant, as changes to the data can be representative of changes to the latest interpretation for the entire excavation. This shortcoming can also be seen to relate to another issue in archaeology, namely the visibility of individual archaeologists in the final report. Contributions to the archaeological knowledge about a site that arise through the efforts of individual archaeologists engaging in excavation, are often not recognised in the final report. By making the changes that occur to the data apparent, we hope that this might also offer opportunities for the contributions of archaeologists, who don't directly contribute to the final report, to be made increasingly visible.

We have addressed this problem with an approach that combines expertise from archaeology, HCI, and web science, into a solution for the grassroots capture of archaeological interpretations. Archaeological research places interpretation at its centre, with archaeological objects and places being understood by analogy to other known examples. Technologies employed previously fail to sufficiently represent the complexities of these interpretative processes. Through interaction with the archaeological community, we determined user requirements via focus groups, interviews, and paper prototypes, and we devised provenance use cases using the PrIMe software engineering methodology[9]. These provenance use cases led to the Xeros data model, and the PoN (PATINA of Notes) web application was designed using the paper prototypes to abstract this complex provenance model into a system for collaborative interpretation that is both intuitive and valuable to archaeologists.

This paper provides contributions to HCI, archaeology, and web science. In HCI terms, this relates to the design process, and how to expose provenance to users. This contribution

overlaps with the contribution from provenance, which relates to the data model and how this is exposed to users. From the archaeological/digital humanities perspective, the contribution of this paper relates to recognising that notes from early research processes have a value to archaeologists, and that technology can be used to capture interpretation in the early stages of research.

We first discuss the design process involved in creating the PoN system, specifically the interactions with archaeology researchers that informed the provenance use cases and prototypes. We then show the intervention itself, and describe how the application meets both the needs of the archaeologists and our provenance requirements through the use of the Xeros data model. We finally provide some initial evaluation of the system from a deployment with student archaeologists.

2. RELATED WORK

Currently, archaeological recording systems, such as IADB[14] (Integrated Archaeological Database) and ARK[4] (Archaeological Recording Kit) support the capture, storage and dissemination of archaeological data. ARK, inspired by the ideals of the post-processual movement in archaeology, allows for the recording of different interpretations and aims to empower the reader to become the interpreter. It also supports per-item versioning, but does not provide for the layering of interpretations or versioning over an entire archive. In contrast, PoN makes these versions immediately apparent to the user, and encourages exploration of this data.

Several projects have investigated interfaces for note taking, ranging from systems focussed on personal information management (PIM) to management of handwritten notes and shared notes. This work generally focuses either on the organization of notes, through grouping and collation, or on the augmentation of notes content through pattern recognition. A relatively early example is NotePals [2], a system that enables users to take notes on a mobile device and later share and organize them through a desktop Web interface. On the Web, notes for multiple users can be merged, grouped by event or associated to external documents, such as pdf files or html pages. As an example of note augmentation, Jourknow [16] is a system to record and organize personal information. It requires users to write notes in a structured format that can be parsed and subsequently processed to extract, for example, location, time and contact information. This information can subsequently be used to trigger automatic, context-aware reminders. For a more detailed review of personal information management systems we refer the reader to [6], a survey spanning almost three decades.
PoN, the system we propose in this paper, is different from previous work in notes management systems for its focus on the evolution and stratification of content, in addition to its organization in groups and sharing.

Wikis are web applications that allow users to quickly and collaboratively edit web pages [7]. They support collaborative accumulation of information and they normally keep track of individual page edits, maintaining a "page history." Their application to information management in organizations [1] and in large online communities (e.g. Wikipedia) [17] has been studied within the fields of Computer-Supported Collaborative Work (CSCW) and HCI.
Similar to Wikis, in our work we keep track of user editing information, but we also place a strong and explicit emphasis on extending one piece of information with another, which is currently lacking in Wikis. Moreover in PoN, versioning is maintained at the level of the entire database, rather than per individual items (e.g. pages).

In computer science, provenance is defined as a record of the entities, activities, and people involved in producing a piece of data or thing, which can be used to form assessments about its quality, reliability or trustworthiness[12]. Its concepts have been applied successfully to workflow technology, used by e-Scientists, with VisTrails[15] and Taverna[18] producing provenance 'traces' - a representation of all dependencies amongst products generated by the workflow execution. Many different representations of provenance exist, and the Open Provenance Model[11] (OPM) was designed to be a technology-agnostic model to allow for the exchange of provenance between systems. OPM was one of the main references to the W3C PROV[13] standard, which is used within PoN. For further information, a comprehensive survey paper is available[10].

3. DESIGN PROCESS

To inform the design of the intervention, a number of in situ interviews were carried out in the archaeology department of the Faculty of Humanities at the University of Southampton. Each interview took place in the labs where the researcher worked, and sometimes moving between different labs. The interviews were documented through a combination of handwritten notes, audio recordings and photographs. The audio recordings were then manually transcribed and thematic analysis was conducted on these transcriptions. The results of this analysis were then used to inform the design of focus groups and further interviews, described later.

The interviews revealed many details of the work practices that occur while archaeologists are engaged in 'post-excavation research', a phase of research which occurs when archaeologists return from fieldwork. Post-excavation research typically involves the scientific analysis of physical *artefacts or* finds obtained through fieldwork, and the synthesis of the results of this analysis with the textual and drawn records to create a holistic interpretation of the site. Our initial interviews were conducted with a professor, lecturer and PhD student who specialised in the identification and analysis of archaeological ceramics, and an osteoarchaeologist (an archaeologist who specialises in the analysis of human remains).

One established practice we observed is for researchers to use a strict unique ID system for all objects coming from an excavation. This ID provides an explicit link between an artefact and information documented during excavation, which includes the find location and the context for its discovery. The IDs are generally quite verbose, and include the site of origin (e.g. "Great Chesterford, Juvenile, 70" - could include photo here), defining a hierarchy of artefacts and contexts from which they originate. Such hierarchy can be represented in a tree structure, as illustrated by the diagram in Figure 1.

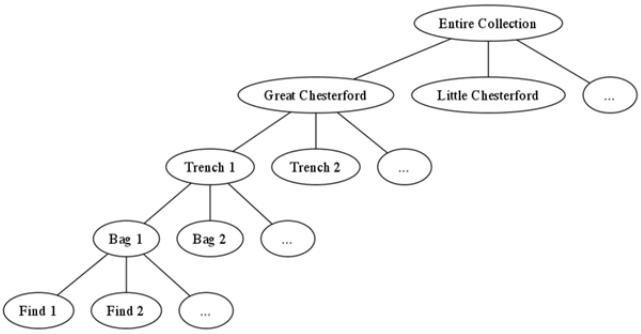

Figure 1: A hierarchy of artefacts and contents.

In some instances artefacts are grouped together from the moment they are excavated, kept in a bag with a tag associated with it, indicating a feature number (e.g. "trench 5") and context (e.g. "GC2705"). Such finds are re-grouped in the sorting process: pottery sherds belonging to two different contexts may be identified as belonging to the same pot, which can be partially reconstructed. In this case, the reconstructed pot is assigned an ID number associated with the individual sherds which usually still have their ID number written on them. A tiny section of any of these sherds may be sent along with others for further analysis (e.g. petrographic, lipid) and each takes a new number, associated with the ID of the sherd/pot it belongs to. This grouping process and the related association of IDs is very important for tracking back the provenance of the artefacts. In other instances, finds may be grouped together at much later stages, for example researchers may create assemblages of finds and analyse them collectively to try and demonstrate a given hypothesis (e.g. the evolution of a pottery style).

We observed a variety of practices to record information in the labs: from systematic data entry, such as filling out a printed recording form that was later transcribed to a computer spreadsheet, to informal notes on the back of an envelope. Part of this information may be lost in subsequent, more formal and higher level records of the research, such as reports or academic papers, that later build on it. However, during the interviews informants discussed with us their experiences of surreptitiously finding the notes of other researchers, and how these were often useful to their analyses. They also described how they would pass their notes between one another for annotation and comment. Therefore most of the informants agreed that making raw notes available might be useful, especially if they were made easy to access.

Based on focus groups and interviews we developed the hypothesis that the notes and data entries that researchers collect while they are in the lab are an important trace of the research process, and we are interested in the structure of this information, and the opportunities it provides for search, browsing and visualization. We believe that the notes are already structured in a rather precise fashion, even though not necessarily always explicitly. We think that the structure of the notes goes beyond simple association to artefacts. We suppose that it involves also relations among notes, such as versioning and layering. By versioning we mean that a note may be an edited or refined version of an earlier note (similar to wiki pages). By layering we mean that a note may be added to one or more earlier notes, to express agreement, disagreement, to include additional or more specific information, or to elaborate deduction and interpretation. Figure 2 illustrates the more rich structure that we are expecting to find through our intervention. Here black arrows indicate association with artefact, red arrows indicate versions and the blue arrows indicate layering. This layering and versioning process may be carried out by an individual researcher, on his own notes, or by different researchers in a conversational manner. Because information is already linked to specific artefacts, these artefacts can become pivotal elements for navigating and sharing information. Often artefacts are already classified as originating from a particular site, from a particular period of time, or from a particular type of object or species (e.g. vase, horse).

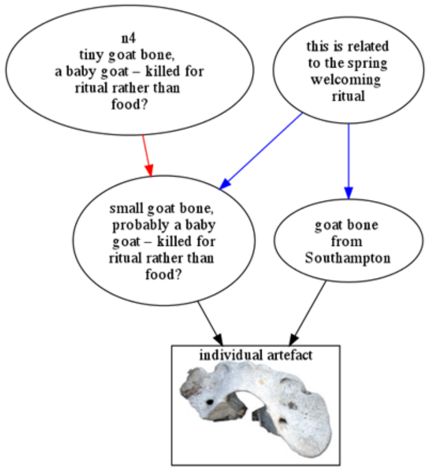

Figure 2: A possible layering of notes, with notes attached to finds as well as other notes. Black arcs indicate reference to an artefact, red arcs indicate information edit, blue arcs indicate information extension or interpretation.

3.1 Provenance Use Cases

A question that one often hears about provenance is "how much should we record"? Too much may lead to performance concerns, whereas too little may reduce the usefulness of provenance. Against this background, PrIMe[9] is a software engineering methodology to design provenance-enabled applications. The gist of PrIMe is to identify use cases exposing the need for provenance-like behaviour, to instrument the code to record a sufficient amount of provenance information, and to implement functionality leveraging recorded provenance to address use cases. In using PrIMe as we designed the PoN application, we integrated provenance tightly in the data model, so as to support acquisition of multiple archaeological interpretations.

From the interviews and analysis of archaeological practice, we identified the following provenance use cases. Using PrIMe, these helped to expose provenance in the PoN system:

1. A user should be able to revise item metadata.

2. A user should be able to group items, add an item to a group, or remove an item from a group.

3. A user should be able to revise structured data entries.

4. A user should be able to revise notes.

5. A user should be able to layer notes, to express alternative interpretations.

6. A user should be able to group notes, add a note to a group, or remove a note from a group.

7. A user should be able to view and navigate versions and layers of notes, as well as versions of groups.

8. A user should be able to roll back the system to a previous state and replay the actions that led to the creation of interpretations.

4. ARCHITECTURE

We wanted to study and understand the structure of the notes archaeologists take in their labs via an intervention, and explore ways to leverage this structure to browse and visualize the collected notes. As such, we designed PoN: a web-based system that allows users to take notes in electronic format, with particular emphasis on the structure among notes, possibly trying to elicit such structure and make it explicit.

PoN allows users to take and browse notes, with particular emphasis on eliciting capturing and displaying the structure among notes described above. The main function of the system is to allow users to record and browse notes associated with physical artefacts, as artefact-oriented note taking is the most prominent practice we observed. The system should elicit note-to-note relations, yet without getting in the way if a user wants to take a simple note (unrelated to other notes).

PoN was implemented to be used both on the desktop and on mobile phones: users responsible for recording finds would use the system via a browser on a desktop; users working on or near the trenches would make use of the mobile phone interface to quickly annotate as they worked. To ease development, we made use of responsive web design[8]: media queries are used to progressively enhance the site within different viewing contexts. As such, the site adapts itself to the viewing environment, thus allowing it to work well on desktop, tablets, and mobile phones. By using a fluid layout, different screen resolutions are also supported.

We further complemented this design by providing a native audio recorder for Android devices. This provides a more efficient workflow than applications available by default on the phones: a user chooses to attach an audio note, presses one button to record the note, then a final button to upload it to the server. This simple interface is better suited to environments where the user requires a fast turnaround, so the archaeological process is not disturbed. Android users can also upload media via a traditional web interface, and can thus attach photos or video, data files, or audio recorded via other applications.

The PoN web application has three core components: the front-end, built using HTML, CSS, and Javascript; a RESTful API; and our provenance backend, Xeros, which persists its data structures into a MySQL database.

The client-facing side of PoN is comprised of a number of widgets: small self-contained bundles of HTML, CSS, and Javascript, which are downloaded on demand and inserted into the interface. This provides the ability to update a region of the site without having to reload the entire page, thus allowing for faster feedback from the UI. An event-driven approach ensures that each widget is up to date when the state of the system changes. A responsive grid is created, with each widget placed within this. As a result, items can be scaled or hidden when the dimensions of the page alter.

The client-side code communicates to the backend via a RESTful interface. This exposes services that can create, retrieve, edit, and delete items; importers and exporters for CSV and PROV; functions that can obtain the state of the system in a form that is quick to render to the user; and authentication handling functionality. This approach also allows for integration with other software being developed under the PATINA project.

Finally, the actions made in the PoN front end are persisted via the Xeros provenance module. This maintains the graph structure that represents the accumulation of interpretations of time.

4.1 User Interface

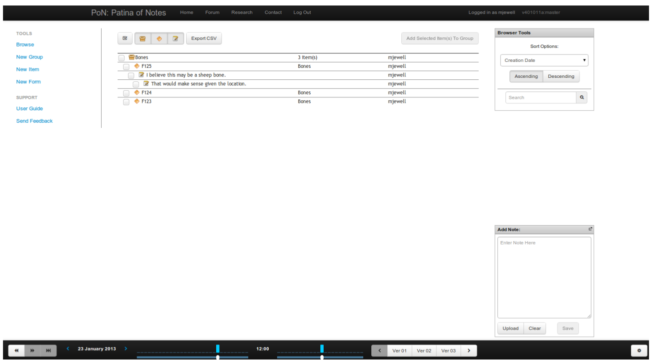

Figure 3: The main view of PoN, showing the hierarchy of a group, item, and notes, and the history view at the bottom.

The user interface design process was iterative: paper prototypes were produced and evaluated with researchers over short sessions (about half-hour). This iterative process allowed us to ensure that the interface was usable and appropriate for its specific audience. The PoN interface (see Figure 3) allows for the creation of a variety of different entities: items, recording forms, notes, and groups. An **item** denotes something that a user might want to interpret: it could be a piece of pottery, a bone, an area of a trench (a 'context'), or even the site as a whole. An item can be created with just an ID, such as F123 in the case of a find. These IDs can be edited, typically in case of transcription errors (see Use Case 1).

A **recording form**, defined as a set of typed fields, allows for users to add structured **data entries** to items. For example, a Pottery form might included fields such as weight and height. While viewing an item, the user can then select this form and enter those details. Fields can be whole or decimal numbers, text, URLs, dates, or yes/no selectors. Data can be edited using the same forms, satisfying Use Case 3.

Items can then be placed into a **group**. These are shown using a view that is very similar to a traditional spreadsheet, or recording sheet (see Figure 4). From this view, new items can be created, and data can be added or edited by entering/altering values in the cells. Selecting a different form then shows the tabular data for that specific form. Users can also select items from the main view, from which they can add them to groups. This group functionality satisfies Use Case 2.

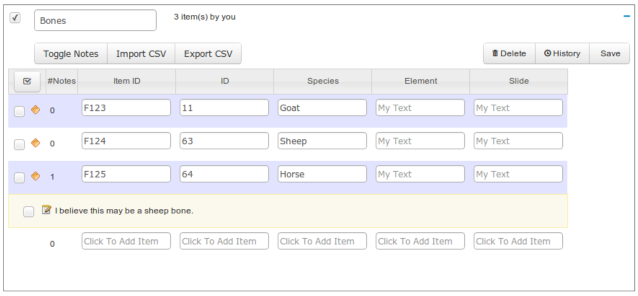

Figure 4: The group view in PoN, shown in a spreadsheet-like form that is similar to an archaeological recording sheet.

The final type, and that most pertinent to PoN, is the **note**. This is a piece of text, written using the Markdown[1] syntax, and thus able to include formatting, links, and embedded images. A note can be attached to items, groups, or other notes within PoN, and can be edited via the UI (addressing Use Case 4). The flexibility of note attachment is ideal for interpretation building, both via specialization and alternation, and addresses use cases 5 and 6.

For example, User A might select a bone and a sherd of pottery, and add a single note suggesting that they are from an animal sacrifice. User B may then attach a note to this interpretation, indicating that there is some extra evidence at the site that suggests that it may be a sheep bone (see Figure X). In this case, the note created by User B is a specialization of the original note, refining the original interpretation. There may be a case, however, where User B believes a different interpretation, and therefore creates a note attached to the two original items with their own suggestion (see Figure Y). This note is therefore an alternative to the original interpretation.

Whenever an entity is added, edited, or deleted in PoN, the previous versions are explicitly kept as part of the record associated with the new version, as described in the Xeros data model. For any item, note, or group, it is possible to see the edits that have taken place during its lifetime via an 'edit view'. This also provides the means to show the differences between versions, similar to a wiki, which is valuable when comparing longer notes.

[1]http://daringfireball.net/projects/markdown/

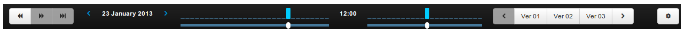

Figure 5: The history view, showing the version navigation interface and timeline.

The history view (see Figure 5) provides the means for a user to navigate to any point of a workspace's lifetime, with each day's activity indicated by a sparkline. This view allows for the replay of actions in the system, and shows how interpretations have accumulated and been edited over time (i.e. Use Cases 7 and 8). Users can browse history by time: a specific day can be selected, and a timestamp is then chosen from the set of modifications made during that time. The next section will describe the Xeros model, which is core to the PoN implementation.

5. THE XEROS MODEL

The Xeros model, named for the eXtension, Edition, and Reduction operations that are core to its definition, is responsible for preserving the state of the PoN system. The three operations allow for the building of interpretations, the edition of entities, and for the non-destructive removal of states of knowledge[5]. No entities are ever removed from the graph, instead operations on the graph create new states that bypass entities where required.

Each entity in a Xeros graph has a positional co-ordinate $\bar{c}$, with components (x, e, r) for extension, edition, and reduction. We also define the vectors $\bar{x}$, $\bar{e}$, and $\bar{r}$ as $(1, 0, 0)$, $(0, 1, 0)$, and $(0, 0, 1)$ respectively.

The edges in a Xeros graph extend the W3 PROV model[13], thus allowing for a degree of interoperation: every entity in the system is a prov:Entity with a URI, and can thus be referenced externally to the system.

This section will describe the core operations of the Xeros model, and then discuss the other features of the approach which are beneficial to PoN. For further definition we direct readers to the paper formalizing the model[5].

5.1 Core Operations

The extension, edition, and reduction map directly to functionality within PoN: respectively, users can add notes and data to items, edit the content and metadata of entities, and remove knowledge from the system. A creation operation is also provided, which simply creates entities within the graph.

In the context of Create, Read, Update, and Delete (CRUD) functionality, the creation operation maps to Create; extension and edition can be seen as Update operations, as they create new versions of created entities; reduction is a form of Deletion, though this operation provides partial functionality, with deletion (see Section 5.2.2) providing for the remaining cases. The Xeros operations are further supported by completion operations, which are detailed in earlier work.

5.1.1 Extension

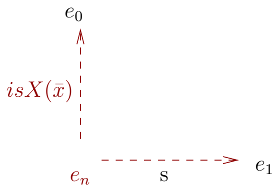

Figure 6: Extending e_0 with entity e_1, thus creating an extension entity e_n which is an extension of e_0 and related to e_1 by the provided relationship s. Created elements are in red.

The extension operation provides the means to associate one entity with another. For example, attaching a note to an item, or an item to a group. Given the original entity e_0 and the entity to attach e_1, a new extension entity e_n is created which has an extension edge $\xrightarrow{isX}$ to e_x, and has a relationship r with e_1. In the case of attaching a note to a find, r would be a $\xrightarrow{hasNote}$ edge (see Figure 6).

In PoN, an alternate interpretation is created by extending the same entity more than once. If a single item has two notes attached, it has two interpretations (see Figure 7). However, if we instead extend item e_n with a new note e_2, we are further specializing the interpretation: the note e_2 adds extra knowledge to the state in which the item is extended with e_1 (see Figure 8).

Figure 2 contains both alternation (e.g. the two notes added to the original artifact) and specialization (the notes added to these two notes).

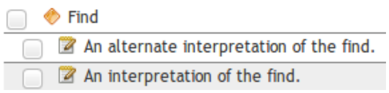

Figure 7: A find with two alternate interpretations. Here, an extension operation has been applied twice to the original find.

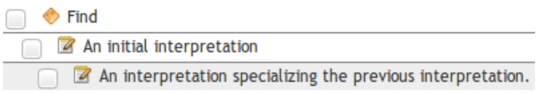

Figure 8: A find with two notes, one conveying extra information about the other. In this case, a second extension operation has been applied to the resultant entity from the first.

5.1.2 Edition

The process by which the content of an entity is edited is a simple graph operation: in editing entity e_0, a new entity e_n is created, and an edition edge $\xrightarrow{isE}$ links the edition to the original entity (see Figure 9). In PoN, edition operations apply to the text of notes, values of data entries, and to the metadata of items and groups.

Xeros also makes use of 'internal editions' which occur as side-effects of the completion operations. These are created to ensure that the graph remains consistent when other entities are dependent upon the edited entity.

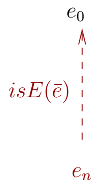

Figure 9: Editing entity e_0 to a new version, e_n.

5.1.3 Reduction

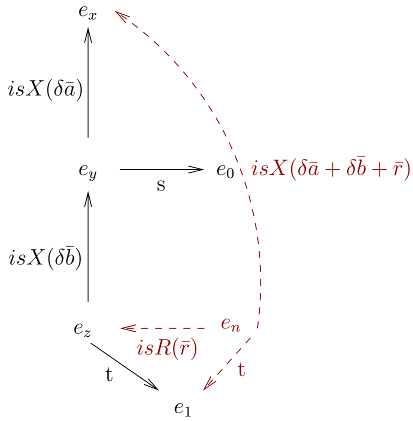

Figure 10: Reducing the state of knowledge e_y by creating a reduction entity e_n, bypassing e_y and directly extending e_x.

Reduction allows for the removal of a state of knowledge within the Xeros graph. If, for example, a measurement on a find is discovered to be incorrect, the state of knowledge asserting this can be removed. As entities are never actually deleted from the graph, new entities are created that skip the omitted state. This process requires two existing consecutive extension entities: the first of these is removed by reducing the second (see Figure 10).

The reduction operation is more complex than the extension and edition operations, and is documented in more detail in an earlier paper.

5.2 Model Features

With the core operations in place, we made use of the Xeros model to provide further functionality within PoN. The universe model, discussed next, makes use of the stored provenance to allow for both the PoN history view and a non-destructive form of deletion. We also use the graph characteristics in Xeros to allow for multi-user support.

5.2.1 Universe Model

In order to allow efficient access to the graph structure that is operated upon by Xeros, we use the idea of a list of 'universes'. A universe refers to the operations that created the entities that have no incoming edges in the Xeros graph: i.e. the minimum set of entities from which all other entities can be reached. So, given an initial universe U_0 containing the creation operation of an entity e_0, a subsequent edition operation on this entity would result in a new universe U_1, containing just the edition operation. Thus, every universe

is effectively a 'snapshot' of the state of the system at a point in time.

The universe approach is key to the history view in PoN: each version shown is a universe, and the timeline is a list of all universes that are available. It also allows for the deletion of entities from the system, described next.

5.2.2 Deletion

While reduction allows for the removal of a state of knowledge from the graph, it requires two consecutive extensions to have occurred. As such, it is not suitable for the deletion of a find with no notes, or a group with no items. However, we do not wish to delete the entities from the graph itself. To handle this, we find the operations that have occurred on the entity and apply them in reverted form: effectively the inverse operation.

To remove e_0 which has been edited to e_n, the inverse edition operation is performed that reverts e_n back to e_0, and then the inverse creation operation removes e_0. By applying these operations, a new universe can be created that has that part of the graph 'rewound'. The graph itself is not modified, but the updated universe no longer refers to the operations that were applied to e_0.

5.2.3 Multi-User Support

The core Xeros operations lend themselves well to multiple users. The extension operation is always propagated through edited entities: if a user adds a note to a find while another user is editing it, the note will be added to the updated find via the completion operations. Parallel edition is less trivial: we notify an editing user if an item has been altered while they have been editing it, and offer them the opportunity to override the change made by the other user, or to keep the other.

5.3 Model / Interface Mapping

The Xeros model is abstracted behind the PoN user interface, and as such the client application functionality is mapped onto the Xeros functions. The note creation functionality maps onto an entity creation followed by an extension, as does the addition of data in an item's recording form and the addition of an item to a group. Notes and data entries can never be unattached, so the creation and subsequent extension of these is performed as a single transaction.

Edition is supported by the various entity views: when editing the ID of items or groups, the content of notes, or the values in a data entry, an edition operation is performed. This is kept unobtrusive in the UI, with only the URL altered to refer to the new entity ID. Editions are made visible via the edit view, which shows, wiki-style, the changes made to a single entity. This list is built by following the edit chain in the Xeros graph.

Reduction is primarily used when detaching an item - so when removing an item from a group. The state of knowledge reflecting that the item was added to the group is bypassed, thus the item remains in the system but is no longer part of that group. When deleting an entity, reduction and reversion are employed: if an item within several groups is deleted, reduction removes it from the groups before its operations are 'rewound' to indicate that it is not in the updated universe.

The universe 'chain', as mentioned earlier, is used to provide the history view at the bottom of the PoN interface. Any version navigation makes use of this, including choosing to view an old version of an item from the edit view. The main view of PoN, showing the hierarchy of entities, is built from the current universe, and even the other entity views make use of the universe as an efficient way to gather the entities required to build the view.

6. INITIAL EVALUATION

To provide an initial evaluation of the PoN system, we performed a deployment at a student archaeological dig site in Tidgrove over the course of two 1-week periods. For many participants this was a first dig: they would be excavating two 'test pits' at the site; performing geophysical surveys; sketching the site; and washing and analysing finds excavated at prior digs.

For the first week of the deployment, a laptop was set up running the PoN desktop interface. This was located in a portacabin on site for shelter, and powered by a diesel generator. While the students were performing initial excavations, an assistant was tasked with adding find data into the system. This served two purposes: firstly, to create some data available for annotation in the second phase; and secondly to test both the hardware and software and locate any issues. A wireless router was also set up outside the portacabin, and mobile phone range was tested as preparation for the second phase.

During the second week, the mobile implementation of PoN was evaluated in two environments. For the first, one of the supervising archaeologists was given a mobile phone during a finds analysis situation, and this was supplemented by both a tablet and laptop running PoN. While the archaeologist analysed stones found at the dig site on a prior occasion, the items were entered into PoN using the laptop, and interpretations were recorded on the mobile phone. The tablet was used in a more journalistic role, recording video of the interpretations. One interesting case arose when a find was found to be similar to one from a different dig: it was simple to create an item for this reference find in PoN, and create the interpretation attached to both finds (see Figure 11). As such, PoN was capable of pulling in supporting evidence for an interpretation, even when the evidence was from a different situation.

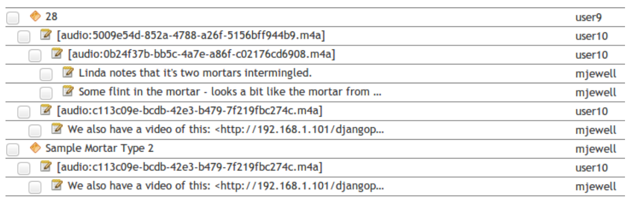

Figure 11: An excerpt from PoN, where Find 28 and Sample Mortar Type 2 have been linked by a single audio note.

For the second environment, phones were given to students who were excavating on one of the two new trenches at the site. At this point excavated items were not being given IDs, so interpretations were created for the region of the trench, or context. In one pertinent case, one of the student archaeologists found an interesting piece of building material in the context, and recorded an audio note. A supervisor then visited the context and added their own interpretation of the item as a specialization (see Figure 12), explaining more about what was discovered:

Student: "Underneath our layer of flint we found a really big bit of tile. We're not sure if it's CBM (Ceramic Building Material)... I think it's roofing tile, due to the way it's been constructed. Also it has a hole at the top."

Supervisor: "It needs a wash to be absolutely sure - it looks post-medieval by the thickness, and we've got a nice mill-hole... It's a handmade one so it's early post-med. Whether or not it's medieval is something you're going to have to check when you wash it, but you don't normally get medieval tiles around here. It's something you'll see on status buildings occasionally."

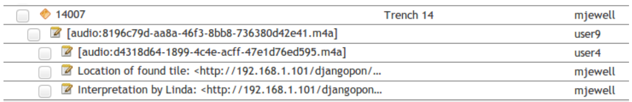

Figure 12: An excerpt from PoN, showing the interpretation of the tile made on Context 14007 in Trench 14. The response by the supervisor, Linda, is added to that interpretation, along with two videos.

7. CONCLUSION

In this paper, we have described the design approach, implementation, and data models behind a novel system for knowledge gathering: we are able to capture and visualise the evolution of interpretation as users encounter and analyse artefacts, and thus enable archaeologists to become more prominent in the narration of a dig site. The Xeros model preserves this graph structure, allowing users to browse through the history of their knowledge structures, while the user interface exposes the graph in an intuitive fashion. This was possible due to the use of a multi-faceted design approach, incorporating HCI and computer science methodologies to inform the creation of a system that captured both archaeology and provenance requirements.

Development on the PoN system will continue, with a focus on the implementation of workspaces. Users will initially have a single, private, workspace, corresponding to a single chain of universes in the data model. However, they will then be able to create more workspaces, and alter permissions on these workspaces to share them with other PoN users. These permissions can be tailored per-user: they may have read-only access, the ability to add notes, the ability to create new items, or the ability to edit existing items. Workspaces will be valuable for future archaeological evaluation: a workspace could be created for each trench at a dig site, with some archaeologists able to create new items and others able to add interpretations.

We are also deploying the software to commercial archaeologists, who are evaluating the mobile and desktop interfaces of PoN. This will allow a more focussed evaluation, with ongoing usage providing valuable feedback. We hope that continued use at a dig site will capture how interpretations naturally accumulate, and will inform us as to features and integration that the users require. This deployment will also provide an opportunity to integrate the PoN software with existing archaeological tools, thus lowering the barrier to entry for users with established workflows.

8. ACKNOWLEDGEMENTS

This research is funded in part by the EPSRC and AHRC PATINA project through the RCUK Digital Economy programme (EP/H042806/1), and in part by the EU DESIRE network.

9. REFERENCES

[1] C. Danis and D. Singer. A wiki instance in the enterprise: opportunities, concerns and reality. In *Proceedings of the 2008 ACM conference on Computer supported cooperative work*, CSCW '08, pages 495–504, New York, NY, USA, 2008. ACM.

[2] R. C. Davis, J. A. Landay, V. Chen, J. Huang, R. B. Lee, F. C. Li, J. Lin, C. B. Morrey, III, B. Schleimer, M. N. Price, and B. N. Schilit. Notepals: lightweight note sharing by the group, for the group. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, CHI '99, pages 338–345, New York, NY, USA, 1999. ACM.

[3] S. Ellis and J. Wallrodt. The paper-less project: The use of iPads in the excavations at Pompeii. In *39th Annual Conference on Computer Applications in Archaeology*, 2011.

[4] S. Eve and G. Hunt. ARK: A development framework for archaeological recording. In *Layers of Perception. Proceedings of the 35th International Conference on Computer Applications and Quantitative Methods in Archaeology (CAA)*, pages 1–5, April 2007.

[5] M. O. Jewell, E. Costanza, T. Frankland, G. Earl, and L. Moreau. The Xeros data model: tracking interpretations of archaeological finds. In *IPAW 2012 - 4th International Provenance and Annotation Workshop*, June 2012.

[6] M. Kljun and A. Dix. Pim research protoypes ideas and what has been transferred to mainstream software. November 2010.

[7] B. Leuf and W. Cunningham. *The Wiki way: quick collaboration on the Web*. Addison-Wesley, 2001.

[8] E. Marcotte. Responsive Web Design. *A List Apart*, 306, May 2010.

[9] S. Miles, P. T. Groth, S. Munroe, and L. Moreau. PrIMe: A methodology for developing provenance-aware applications. *ACM Trans. Softw. Eng. Methodol.*, 20(3):8, 2011.

[10] L. Moreau. The foundations for provenance on the web. *Foundations and Trends in Web Science*, 2(2–3):99–241, November 2010.

[11] L. Moreau, B. Clifford, J. Freire, J. Futrelle, Y. Gil, P. Groth, N. Kwasnikowska, S. Miles, P. Missier, J. Myers, B. Plale, Y. Simmhan, E. Stephan, and J. V. den Bussche. The Open Provenance Model core

specification (v1.1). *Future Generation Computer Systems*, July 2010.

[12] L. Moreau, P. Missier (eds.), K. Belhajjame, S. Cresswell, R. Golden, P. Groth, G. Klyne, J. McCusker, S. Miles, J. Myers, and S. Sahoo. The prov data model and abstract syntax notation. W3c first public working draft, World Wide Web Consortium, Oct. 2011.

[13] L. Moreau, P. Missier (eds.), K. Belhajjame, S. Cresswell, R. Golden, P. Groth, G. Klyne, J. McCusker, S. Miles, J. Myers, and S. Sahoo. The PROV Data Model and Abstract Syntax Notation. W3c first public working draft, World Wide Web Consortium, Oct. 2011.

[14] M. Rains. Towards a computerised desktop: the integrated archaeological database system. In *Computer Applications and Quantitative Methods in Archaeology*, pages 207–210, March 1994.

[15] C. E. Scheidegger, H. T. Vo, D. Koop, J. Freire, and C. T. Silva. Querying and re-using workflows with vstrails. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, SIGMOD '08, pages 1251–1254, New York, NY, USA, 2008. ACM.

[16] M. Van Kleek, M. Bernstein, D. R. Karger, and m. schraefel. Gui — phooey!: the case for text input. In *Proceedings of the 20th annual ACM symposium on User interface software and technology*, UIST '07, pages 193–202, New York, NY, USA, 2007. ACM.

[17] F. B. Viégas, M. Wattenberg, and K. Dave. Studying cooperation and conflict between authors with history flow visualizations. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '04, pages 575–582, New York, NY, USA, 2004. ACM.

[18] J. Zhao, C. Goble, R. Stevens, and D. Turi. Mining taverna's semantic web of provenance. *Concurr. Comput. : Pract. Exper.*, 20(5):463–472, Apr. 2008.