

Less is More: Lightweight Ontologies and User Interfaces for Smart Labs

J. G. Frey^{}, G. V. Hughes, H. R. Mills, m. c. schraefel, G. M. Smith, David De Roure*

Electronics and Computer Science, University of Southampton, SO17 1BJ, United Kingdom

^{*}School of Chemistry, University of Southampton, SO17 1BJ, United Kingdom

E-mail contact: hrm@ecs.soton.ac.uk

ABSTRACT

The Smart Tea project to date has focused on the issues and surprises of using a human-centred design and development approach from the front end interaction design to the back end systems architecture.

Keywords: Human computer interaction, ontologies.

INTRODUCTION

To date, the computer science focus on e-Science research has been to develop appropriate grid services and structures to support massive computation across shared dynamic resources. This approach is particularly valuable if the data being shared across the grid is generated digitally. This is not always the case. Synthetic organic chemists, for instance, who work in wet laboratories record their experiments in paper log books, and publish papers on paper.

So, what do these chemists want from e-Science? The general answer described by the CombeChem project has been “Publication at Source” [1]. Publication at Source represents the ability for chemists to publish all aspects of the data they produce, from experimental results to published papers, as soon as the data itself is generated. Such publication may mean sharing data within one’s laboratory, or within one’s international community.

The Smart Tea work within the Combechem project has been constructing an end-to-end system to support this kind of publication at source, for synthetic organic chemists in particular. By addressing the challenge of the synthetic organic laboratory, which is the least digital field in the chemistry domain, we have the opportunity to evolve a model to benefit all aspects of chemistry (and potentially all groups in e-Science) by starting from what may be called the “worst case scenario.”

There are four components to the project:

1. experiment plan construction and capture,
2. recording the live experiment in the laboratory: the digital lab book,
3. producing multiple views of the experiments for

multiple re-use publication contexts, and

4. the overall ontology and architecture to integrate each of the above three components.

In this paper, we describe the methodology that has evolved as part of the development process for each of these components. We touch on the evolution of this method briefly here, in order of component development.

THE DIGITAL LAB BOOK – DESIGN ELICITATION

The fundamental challenge of taking the synthetic chemistry laboratory into the digital domain is the paper lab book. Synthetic chemists record their experiments into paper log books (Figure 1, left). We needed a way to let chemists capture their experiments digitally, as naturally as they do with paper. Without this digital support, the majority of the data they create would not be readily available for access.

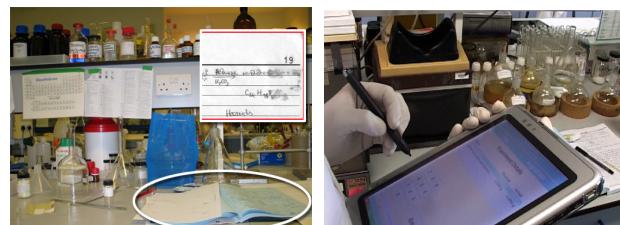


Figure 1. The old and the new: Paper and digital lab books.

While there have been previous attempts to replace the paper note book (see [12] for a review), take-up of such systems has been poor in the area of synthetic chemistry. Drawing on Human Computer Interaction models for investigating practice and artefacts, we carried out an investigation into the lab book, both as recording device and as cultural/political artefact in a laboratory environment. We needed to understand the affordances of the paper book in order to replicate those experiences.

One particular challenge in this space was in terms of design methodology: most approaches assume that the tasks to be modeled in the software take place in an observable period of time, are structured, and are understandable, once explained, to the designer. In the synthetic chemistry laboratory, we learned that experiments can take from

hours to months to complete, that they are loosely structured, and that the decision-making process requires a high degree of domain knowledge and experience. So, while we were able to observe the laboratory and interview chemists about their use and the roles of the chemistry lab book, when it came to analyzing when a recording was made in a lab book, or what kinds of constraints might be placed on that process, our lack of domain expertise made effective analysis impossible. To address the problem – to be able to interrogate the experimental recording process – we developed a new design elicitation technique, called Making Tea.

Making Tea is a multi-stage process based on the concept of learning by analogy [4]. Analogy is usually used to help introduce new concepts through a discussion of known concepts [2]. Students may be introduced to the concept of atoms first as analogous to a raisin pudding. More properties of atoms can be introduced by a comparison to the solar system, with particular emphasis on orbits of planets around the sun. Each stage of the developing analogy reflects the leveraging of previous understanding to support new understanding: both in terms of how the new model is like the previous model, as well as how the new model is unlike the previous model.

Making Tea was developed particularly to address the gap described above between design team knowledge and the domain expertise required for the interaction process to be modeled. In our case, we wanted to understand the synthetic chemist's experimental process. In particular, we wanted to understand the chemist's recording process physically – in terms of what activities took place where in the laboratory that required recording, from measuring chemicals to mixing compounds – and abstractly, in terms of what *specifically* they recorded about an event, how they recorded it during an experiment (drawing, notes, references), and why they recorded these particulars rather than others. While field studies and task analysis would help us understand the context of use and the functional process of performing a recording task, most of us on the team felt that such observations left us at one remove in understanding sufficiently the real-time decisions being made in carrying out an experiment, and in deciding what to record. This would also leave us less able to engage the chemists effectively about their practise.

One option would be for us to enroll in a first year chemistry course, but we neither had the time for this, nor the need to become chemists, since we were more interested in the role of the recording process in an experiment than in the experiment itself. We also did not want to become “experts” per se, and hencepossibly miss certain assumptions in a practice that an expert may not question or express. We therefore needed a technique that would get us closer to that process, so that we could both appreciate the subtleties of what we would observe, and engage in conversations with chemists at a high enough

level that we could make the most constructive use of their valuable time – and ours. Our previous experience, for instance, working with complex expert systems for engine design, had suggested that expert walk-throughs of processes were not the most efficient way to address the kind of problem we faced. Experts can get passionate about explaining low-level detail that, while important for their work, does not necessarily illuminate effectively the core issues of the process to be modeled. For instance, a chemist in an expert walk-through could tell us “we now add acetyl chloride to benzene to perform a Friedel-Crafts acylation,” and we would all nod our heads. With the tea experiment analogy, we could ask whether or not it really *is* important, in terms of the experimental process, to add the milk to the cup before the tea or vice versa.

The analogy of Making Tea as an experiment would potentially provide us with the leverage we sought for understanding the attributes of the process we wanted to translate from the physical to the digital. It would also provide a mechanism for experts to communicate with us in a focused and productive way for the goal at hand, and to maximize contact time both observing and engaging with the experts.

Making Tea evolved as a multi-stage process that we used throughout the design cycle. The stages are,

- defining an appropriate analog,
- validating the analog with domain experts,
- using the analog to interrogate practice, and
- using the analog to test translation of practice from the physical to the digital.

This last stage of translation is an iterative one, where design revisions are checked against the model repeatedly. The advantage of Making Tea is that it gave us a process that was observable and repeatable within a reasonable period of time. The familiarity with tea meant that we could, in the design elicitation phase, use terms which were readily understood by both chemists and designers. The analogy was also effective in eliciting where making tea was *not* like an experiment. These comments were as valuable as where the model did match the experiment.

Once we had developed mockups, in the design reviews with chemists, we ran through each mockup by making tea with them, as a kind of synthetic chemistry in the small. Based on feedback from the chemists, which increasingly became more sophisticated, less constrained by Tea as our own understanding grew, we developed more robust prototypes. These we tested again with the making tea experiment, but then with more complex examples of “real” experiments.

By Making Tea with chemists, we developed a working discussion space where we could ask questions or describe problems in a common parlance that also engendered a kind of social quality to the exchanges. The common language

of making tea reflected common goals among the stakeholders. This language aspect, we think, is not unlike what Bødker [1] sees as the value of common language/language acts among design participants to enable design.

Most particularly, our design approach, from investigation to development, has been grounded in working with real chemists throughout the design process. One design theme repeatedly emerged from each of these contacts: we usually needed a lighter approach than the designs/functions we were favoring. It is an information system designer's tendency to add forms, functions and constraints to ensure the usability of the data that is captured, often at the cost of the usability of the system by its practitioner. We had to pull back from much of this attitude – the attitude that says, we can capture temperature data really easily, let's put in temperature data – in order to connect with what the chemists were telling us is important to them, which frequently turned out to be an existential exercise of how to design for variable, context-dependent interactions.

The result of developing a method better to support designers' ability to interrogate highly expert, loosely structured, potentially high-duration tasks has been so far the development of a well-received prototype of a usable digital lab book system[5, 6]. Based on the lessons learned about recording process here, we have more recently been using the Making Tea experiment to help us develop the ontologies, architecture and experiment planning application.

DESIGNING THE INTERFACE

It is an obvious goal in HCI when replacing any existing system that a new system must either be no more onerous to use than the existing one, or else the additional complexity must have immediate and obvious benefits to the user. Most of the existing lab book systems which we have identified fail either on the point of general accessibility and searchability of the data, or on this axiom of usability. Some systems aim to be no harder to use than paper notebooks; most attempt to provide large benefits with the addition of increased complexity. In the latter case, many of the systems are designed as desktop applications for ordinary PCs, which is at best unhelpful in the context of a synthetic chemistry laboratory, where space is usually at a premium. Most of the commercial systems which we (and the chemists which we spoke to) were aware of were page-based, required the use of a mouse and keyboard, and dealt with well-structured documents that were more suited to the form of an academic paper or report than a note-taking system.

As we became more familiar with the modes of operation of the chemists that we were working with, it was increasingly obvious that most data entry was either very simple or totally free-form. It might be possible to record the data entered by the synthetic organic chemists using a

strongly-structured entry form and database storage, but the entry forms for the synthetic laboratory that we observed would not translate well to the other branches of chemistry with which we were familiar (for example, the experiments performed by physical chemists).

In the synthetic laboratories, it was clear that while an experiment was being set up, the chemist switched context between the apparatus, the workbench, and their notebook frequently and rapidly. This indicated that the primary goal in the design of a laboratory-side user interface had to be simplicity, as any form of complexity would increase the cognitive load on the chemist by forcing them to remember their context in the data-entry/notebook application as well as in the experiment. After the experiment has been performed, however, the chemist will organise, analyse, and write up their experiment at their desk outside the laboratory. This part of the process can use more complex interfaces to manage the data which has been collected in the laboratory.

After the process of elicitation and interface design, we made simple mock-ups of the proposed user interfaces. The feedback that we obtained from the chemists to the proposed interfaces was positive, but was always pushing towards simpler interactions and more flexibility. After a few iterations, we finally developed the user interface applications described in the following section.

THE WEIGH-STATION

The weigh-station (see Figure 2) shows the dry materials to be weighed on a scales, and would be situated next to the scales in a laboratory. The liquid measure shows the other ingredients to measure. The chemist can instantly see what needs weighing and how much to weigh. Once they have weighed or measured the amount, they enter the precise, final amount using the oversized numeric keypad and press the “enter” button. The data is added into the experiment record. The interface is oversized and easily read from a distance, allowing for the lack of accuracy caused by using the pen interface whilst wearing gloves.

THE BENCH

The bench (see Figure 3) displays the steps of the chemist's experiment. These are listed in the top half of the screen, with the bottom half being an area allowing for free sketching. Our inspiration was that of e-mail applications showing the message headers at the top of the window and the message body at the bottom. The user would follow a step in their plan, write any notes with the pen and tick a box to indicate completion of the task. The bottom pane is simply a drawing area and the data is stored as vector data. We do not utilise the handwriting recognition tools of the tablet nor do we provide a keyboard. We found that chemists just need a place to write a few cursory comments on what has happened when they perform a process. They also love to sketch – on anything to hand, not just their lab books – so by not providing a highly structured interface

we allow them to sketch and write freely.

Figure 2. [LEFT] The Weigh-station Tablet Application, showing the list of reagents, with planned and actual amounts.

Figure 3. [RIGHT] The Bench Tablet Application. The top half of the display shows the planned steps to be performed. The lower half shows hand-written annotations for the selected step.

TLC PLATE IMAGE MANAGEMENT

Another client application that has been developed for this project is a system for taking digital photographs of Thin Layer Chromatography (TLC) plates and managing their integration into the experiment record. During our investigations into the way lab books are currently used we observed how common it was for chemists to use TLC plates, and how much effort was needed to draw a life sized replica of the plate into the lab book. TLC plates fade over time, and so the common practice is to draw a picture of the plate into the lab book. A simple rig was built using a standard digital camera to take photographs of TLC plates. The software allows users to add metadata to the image as well as providing calibrated tools for marking up the image with features such as the baseline, solvent front and spot locations.

THE PLANNER AND THE PUBLISHER

Our current efforts are now focused at both ends of the experimental process: planning it, and publishing it (in the Publish@Source sense). Chemists have to fill in a COSHH (control of substances hazardous to health) form for safety requirements (see Figure 4). We have leveraged this necessity into a virtue for our planning approach: we modify the COSHH form to include extended descriptions of the steps to be taken in the process. Our current challenge is to determine the best way to support the chemist recording that process so that the system will know which kind of reflux, of the several dozen variants, a chemist may mean, without having the chemist tab through several dozen possible selections. The system has to work for the chemist; not the chemist for the system.

Similarly for publication, we are investigating how best to represent the multiple contexts in which publication, can be displayed. Again, we are working with the chemists, and finding, once again, that less is more.

COSHH ASSESSMENT FORM			
Record No.			
SUBSTANCE NAME	PHYSICAL FORM	QUANTITY	NATURE OF HAZARD
Water	liquid	1000ml	None
Dextrose	Solid	<20g	possible irritation to eyes and skin
Caffeine	Solid (pow)	<1g	Harmful if swallowed, induce vomiting
Milk	liquid	<20ml	No particular hazards

NATURE OF PROCESS
liquid extraction of caffeine, followed by combination with dextrose to produce a sweet drink

Is there a less hazardous substance? **No**
 If so, why not use it?

CONTROL MEASURES REQUIRED *No specific measure required*
 (Local exhaust ventilation, personal protection, etc.)

Figure 4. COSHH Form, showing the hazards of making tea

The current planner application (Planner0) (see Figure 5) is a very simple web-form application written in PHP. It allows the basic structure of an experiment to be generated from scratch or from copying an existing experiment. The main metadata for the experiment, such as the experimenter's name and a high-level description of the experiment, can be entered or updated, and more detailed data about the experiment, such as the lists of ingredients and steps can be created or modified.

The current Planner0 interface does not allow the creation of a non-linear experiment plan (i.e. where more than one step can be done in parallel), but this is not a serious problem for our test environment of a synthetic organic research laboratory, where the individual experiments rarely have parallel execution paths.

Figure 5. The Prototype Experiment Planner. At the top is the experiment metadata. In the middle, the list of ingredients for the experiment. At the bottom, the descriptions of the steps to be performed.

After we implemented the user interface of our system, with a simple backend storage system, we tested it by asking a small number of synthetic organic chemists to plan and perform a real experiment in their laboratory, using our system instead of a normal paper lab book.

They found the system very easy to use with a minimum of training, and in observations of the system in use it was clear that after a short period of familiarisation in the laboratory, the fact that it was an electronic interface was mostly forgotten. In the follow-up interviews with the testers, they were extremely positive about the system, claiming that “it’s so good to know that your data’s safe,” and “I’d rather be where we’re heading [to the electronic lab-book], than where we are now.” The fact that the chemists perceived even the basic prototype system as immediately useful to them is extremely encouraging for further research and development in this area.

ONTOLOGIES AND ARCHITECTURE

In order to represent to the system the experimental processes described in the log book, we needed to develop an ontology. Making Tea gave us two guidelines for this part of the architecture design process: firstly, does what we build reflect the intentional measures or actions of the chemist, like add *SOME* of this rather than a specific amount, and secondly, can what we design represent at least the tea experiment? These challenges were important for the early stages of the architecture design we developed from what we had learned in Making Tea.

From our knowledge of working with chemists on their process of both planning and recording an experiment, it soon became clear that we needed a multilevel approach for description: a layer to reflect the plan of the experiment (informed by the COSHH form), a layer to describe the record of the experiment that the chemist makes and a layer to describe the provenance information at the much more detailed scale of which services were called when by the computers while recording the experiment details for the chemist.

The primary goal in the development of the ontology has been to represent the “human scale” of information which is recorded and read primarily by chemists, rather than the highly detailed information which would be read primarily by computers. What has become clear dealing with real chemists is the need to express intentional concepts like “some” or “it’s like this – sometimes” which plays havoc with notions of writing templates to describe processes. Instead, we have needed to develop an approach which can reflect well-defined flexibility. Such flexibility has been informing the development of our architecture and our approach to services.

THE ONTOLOGY

We have developed an ontology in RDFS to encompass the major phases of an experiment: planning the ingredients, planning the procedural steps, and recording the experiment. The ontology uses a high level representation of work flow: Processes have inputs and outputs. A process can be an activity performed by a person such as refluxing, or it can be the running of a software application. In the former case the output is a substance in the reaction

chamber, in the latter case it will be a file or a collection of files. In this case the RDF will record the URI to the output files. These can be stored in other, more appropriate, database systems in a distributed manner. In the case of the experimental procedure the URI to the result of a process is a more abstract concept of little direct use to the chemist but forms the input to the next process along with any new reactants.

Information recording using the ontology is a representation of the human-scale activities of scientists performing experiments. The requirements have been gathered from our work with chemists in the laboratory. We have also incorporated requirements from the chemistry aspects of the CombeChem project so that concepts such as a process can mean either an act performed by a human or the running of a piece of software. The initial design was inspired by analysing the experiments performed by chemists during the tablet interface trials as well as some simple procedures such as the preparation of aspirin. We quickly noted that there was a spine of activity running through the record in which a process led to a product which in turn formed the input to the next process step. In the laboratory this maps to the adding of ingredients to a reaction vessel and the actions performed on it to result finally in an end product.

This process-product pairing is important, both in terms of the physical reality of an experiment, and also in software where each computational process results in output files. Recording these intermediate outputs allows us to link to each outcome from the final report and provide far greater opportunities for other scientists to reproduce experiments. Previously all that remained from an experiment would typically be the analysis of the final product. Our ontology is designed to make it easier for systems such as Ebank [6] to retrieve all of the intermediate data and results vital to reproducing a procedure.

MAJOR CONCEPTS OF THE ONTOLOGY

Our ontology separates out a plan from the record of an execution of that plan. There is no reason, from the point of view of the ontology, why a plan cannot have more than one execution trace, although we do not currently allow for that in our software. The separation between plan and record is an important one, since the two may not match exactly. An experimenter may perform additional steps over and above those in the plan, for example if something unexpected or unplanned happens (say, an interim result is a solid, rather than a liquid as expected, and must be dissolved to continue).

The two main components which the ontology models are Materials and Processes. A *Material* may be either a data set, or a physical sample of a chemical (possibly some anonymous and indeterminate mixture partway through a reaction scheme). A *Process* may be a purely in-silico process (a computation); a purely in-vitro process (a

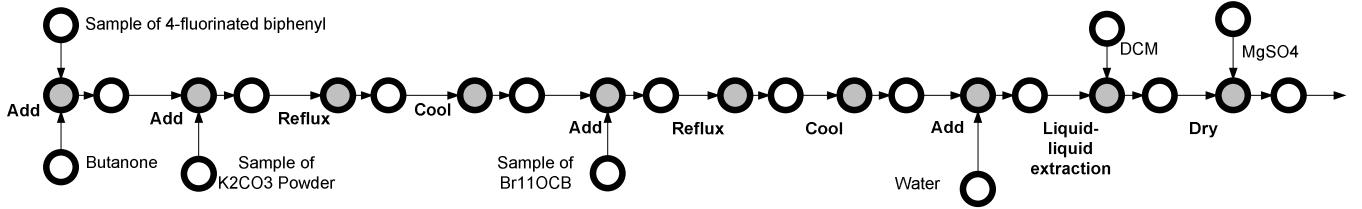


Figure 6. Process-Product spine of events in an experiment performed during the tablet evaluations. It shows the main ingredients being added (far left), and the processes performed on that mixture. Note the alternation of processes (grey) with products (white) along the experiment path. In this diagram observations and annotations have been removed for clarity.

chemical reaction); or a hybrid process such as the measurement of a spectrum, which takes a substance as input, and produces a piece of data.

The ontology has a hierarchy of different process types – for example, it separates in-vitro processes into broad classes of “Separate”, “Mix” and “React”, with each of these being subdivided into different types: Separate has subclasses including Filter, LiquidLiquidExtract and ColumnChromatography. These may be further subdivided to give more detailed descriptions of the processes being used, as required. Extending the ontology to include additional types of process is a job for the domain experts (in this case, chemists), who are much better able to identify what classifications are useful. Extending the ontology in this way does not require a high degree of expertise in ontology creation, as it is primarily a matter of classification of processes. There are tools available which can aid the domain expert in modifying an ontology [11].

To use the ontology’s Materials and Processes, observe that every Material may be the input to or the result of a Process. Some Materials will be used as input to several processes (if they are data, or if they are a large sample of substance split up into smaller samples). Some Processes may have many Materials as inputs (or as outputs – consider a filtration or fractional distillation process). Thus, the main part of the experiment consists of a network of nodes, always alternating between Materials and Processes. A measurement, which may be of some property of a substance, or of the state of a process, or even an arbitrary annotation made by the experimenter, has three parts to it: a measurement *process*, which is the act of making the measurement, a measurement *material*, which is the URI representing the result, and optionally a type/value pair, representing the data of the measurement in the case that it is a simple (one-value) observation with a unit, such as a weight or a temperature.

ONTOLOGY EXAMPLES

We have developed a diagram style during this project to aid us in visualising the ontology and any example experiment models. These diagrams aid us in writing code to navigate along paths in the graphs.

The URIs in the sample experiments were created manually with descriptive names to aid debugging. In the live system

the URIs are allocated by a simple web service to generate unique URIs. Fragments of such diagrams are included here to illustrate the experiment model structures. In the diagrams the shaded circles depict processes, hollow circles depict substances, triangles represent the making of observations and squares represent literal values. Each circle, triangle or square is a node in the RDF with its shortened URI represented in bold, black text. The class name of each node is given in italics and arrows represent relationships.

THE PROCESS-PRODUCT PAIRING

There is a spine running through an experiment in the form of a series of process-result pairs. Figure 6 shows the nodes from one of the experiments carried during evaluation of the Tablet PC interface software. In the first process step butanone is added to a sample of 4-fluorinated biphenyl resulting in a mixture in the reaction vessel. Potassium carbonate is added before the vessel is heated. The diagram does not show annotations and observations made by the chemist. Examples of these are described below.

ANNOTATIONS

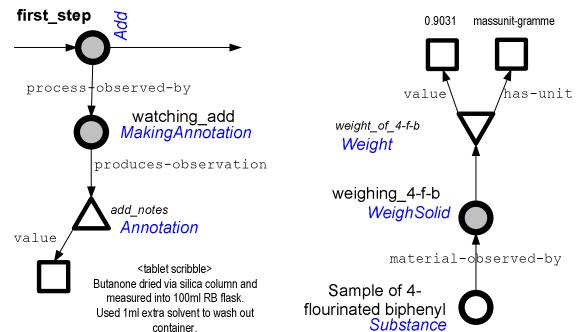


Figure 7. [LEFT] RDF graph of the annotation of a process (*first_step*), showing the annotation process, the annotation URI itself (triangle), and the data of the annotation (squares).

Figure 8. [RIGHT] RDF graph of the observation of a process, in this case the weighing 4-fluorinated biphenyl.

The experimenter made an annotation using the Tablet PC during the reflux and the graph fragment is shown in Figure 7. From the *first_step* node there is a *process_observed_by* property to a *MakingAnnotation* node. This represents the actual act of making the annotation and results in an observation node *add_notes* which has a literal string containing the tablet’s sketch data (see Figure 2 for an

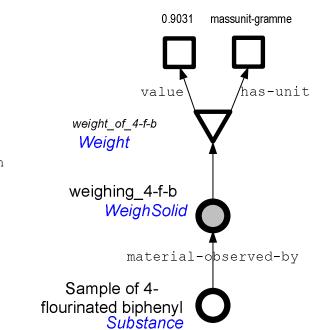


illustration of this type of annotation being entered).

OBSERVATIONS

Some methods of making observations are more complex than others. The ontology accounts for this with SimpleObservation and ComplexObservation classes. Weighing solids, measuring liquids and making annotations are all subclasses of SimpleObservation. Figure 8 shows the graph for a simple observation. Examples of ComplexObservations include mass spectroscopy or the making of a thin layer chromatography plate (TLC). In these processes there are many more parameters to record about the process of making the observation. The ontology does not specify the details of these but the ModelServer will store and retrieve all statements attached to any type of Observation so there is no need to specify carefully the details in the ontology. The output of the TLC application has been designed to integrate into the ontology as a complex observation.

THE METADATA STORAGE SYSTEM

Experiment metadata – the connections between the results and processes in an experiment – is stored in a persistent Jena RDF store [7] backed by MySQL. We have written a system to manage this storage as a Web Service (using SOAP) with an API that serves and manipulates sub graphs of an experiment. The API allows clients such as the Planner and Tablet to retrieve logical components of an experiment such as the textual steps of a plan or the nodes depicting the core record of the experiment process.

QUERYING THE MODELSERVER

The ModelServer API closely matches the ontology and has evolved to meet the needs of the front end applications. For example the call `getExperimentMetaData()` will return the top level experiment node along with all of its properties such as the experiment name, the name of the experimenter and the first planned process node and final planned product node. From this information a client application can use subsequent API calls to navigate the experiment graph.

The method `getRecord()` requires the ModelServer to walk down the graph from the top level Experiment node to the first process record node and along the process-product pairs. It cannot assume that any of the nodes exist as the record of the experiment may be in any state.

The Web service methods return triples as a complete RDF graph. This requires the clients of the web service to be able to manipulate RDF. In the case of the tablet it is a Java application that utilises Jena to load the RDF returned by the Web service and work with the triples.

MAKING CHANGES TO THE MODEL

As the experimenter works, the record of the experiment needs to be updated by the client applications. For example, recording a new weight for an ingredient, or adding a new note to an annotation. The method for adding or updating

triples in the store is simple yet powerful: a client application supplies two RDF models containing the triples to be removed and the triples to be added to the experiment model, and calls a function in the API to make the required changes to the triple store.

A simple example is for the tablet to update a text annotation from the bench drawing area. The sketch data is a single long string stored in a single literal. The tablet client must supply a model containing the triple of the old sketch and a model containing a triple for the new sketch. This may seem like a lot of work for one simple update, but good software engineering can simplify the process. The real power of this technique arises when the clients manipulate and store whole sub-graphs of an experiment. The planner does precisely this, building and manipulating whole RDF graphs as the user creates and edits the plan of the experiment. When the user presses the Save Plan button the planner updates large sub-graphs of the experiment model in one transaction. The tablet is also able to manipulate whole sub-graphs of the model as required. As the user performs the experiment and ticks off steps from the bench the tablet will create the record structure as well as annotation structures for the sketch pad.

BACKGROUND

In [12] we review lab book replacement systems both research based and commercial. In this paper we highlight a few of the other projects using ontologies to represent experiments and scientific results.

The Microarray Gene Expression Data Society [3] is concerned with describing samples and processes used in microarray gene experiments. Their first achievement was a format for the minimal annotation of an experiment. From there they have created an object model to describe experiments. This is accompanied by a tool set to aid developers to convert outputs from systems into their formats, enabling data exchange. The formats are now becoming the de facto way to publish data in the field.

The experiments that we have concerned ourselves with happen mainly in the laboratory, but many are run using computers only. Managing the results of such experiments is of increasing importance in eScience and grid computing. The Earth System Grid project [10] is a major collaboration to create an ontology that describes the processes and data found in such large scale distributed software systems. The major high level concepts include pedigree – the line of ancestry of a data from creation through various transformations, scientific_use – how the scientist used the data including parameters and configuration, other concepts include datasets; services; and details of access to services.

The Collaboratory for Multi-scale Chemical Science (CMCS) [9] is a very large project building a toolkit of resources to aid in multi-scale combustion research. The portal-based project encompasses a vast amount of work in

individual systems to share resources and results in the community. CMCS is a powerful approach to managing diverse data formats produced from different analysis systems and tools such as electronic notebooks.

One of the major components of the CMCS project is SAM (Scientific Annotation Middleware) [8]. SAM is a set of middleware components and services designed to aid storage of data or metadata in its native form and provide tools to map metadata from one form to another. Mappings are written in XML so that an arbitrary combination of analysis systems, underlying data stores and electronic notebooks can be used together through the one portal. The system is based upon a WebDav implementation adding metadata management and notebook services layers.

The CMCS and SAM projects are by far the most advanced work in electronic notebooks and the distributed storage of diverse experimental data. Where their approach differs from the work described here is in our use of an ontology to describe the nature of an experiment and that our client/server API is designed to build a structured RDF graph of an experiment plan and record.

CONCLUSION

In conclusion, we have designed and developed a system for recording laboratory notes, based on experiences of several different chemistry laboratories. We have produced innovations in not only the design methodology, but also in the lightweight style of the user interface, and in the human-scale representation of the information being recorded, leading to a system which supports the experimental process end-to-end, from planning of the experiment to publication at source of the process and results. User-acceptance testing of the system in a live wet-laboratory environment was extremely positive, particularly in the light of the severe adverse reactions of chemists to existing systems.

Current and future work will concentrate on the presentation of the information for an experiment in an easily-read and easily-navigated form for Publish@Source dissemination. Other areas which are in need of additional work are to improve the planner interface to allow for more complex process plans; the development of service wrappers to record in-silico processes and results in the same way as we currently record in-vitro processes; and developing or using a suitable ontology describe the inputs and outputs of different process classes.

ACKNOWLEDGEMENTS

This work is supported by the Combechem project: (Structure-Property Mapping: Combinatorial Chemistry and

the Grid), grant GR/R67729/01 from the UK Engineering and Physical Sciences Research Council. We wish to thank the chemists from the synthesis laboratory of Dr Martin Grossel. The TLC slide image annotation application was written by Will Davies.

REFERENCES

1. Bødker, S., Ehn, P., Knudsen, J., Kyng, M., and Madsen, K. *Computer support for cooperative design*. Proc. of CSCW (1988): 377-394.
2. Chklovski, T. *Learner: a system for acquiring commonsense knowledge by analogy*. Proc. of Knowledge Capture, 2003, 4-12.
3. Stoeckert Jr, C.J., Causton, H.C., and Ball, H. A., Microarray databases: standards and ontologies, *Nature Genetics*, 32 Supplement – Chipping Forecast II, (December 2002), pp. 469-473.
4. Douglas, S.A. and Moran, T.P. *Learning text editor semantics by analogy*. CHI, 1983, 207-211.
5. Frey, J. G., De Roure, D. and Carr, L. Publication at Source: Scientific Communication from a Publication Web to a Data Grid. *Proceedings of Euroweb 2002*, British Computer Society, Swindon (2002) 88-90.
6. Heery, R., Duke, M., Day, M., Lyon, L., Coles, S., Frey, J., Hursthouse, M., Carr, L., and Gutteridge, C., Integrating research data into the publication workflow: eBank experience. PV-2004, 5-7 October 2004 – Frascati, Italy.
7. Jena – A Semantic Web Framework for Java. <http://jena.sourceforge.net/>
8. Myers, J.D., Chappell, A., Elder M., Geist A., and Schwidder, J., Re-Integrating the Research Record. IEEE Computing in Science and Engineering, 2003. 5(3): p. 44-50. Available at <http://www.scidac.org/SAM/>.
9. Pancerella, C.M., Myers, J.D., et. al., "Metadata in the Collaboratory for Multi-Scale Chemical Science", Proceedings of the 2003 Dublin Core Conference: Supporting Communities of Discourse and Practice-Metadata Research and Applications (DC 2003), September 28-October 2, 2003, Seattle, Washington.
10. Pouchard, L., Cinquini, L., Drach, B., et. al., Exploring Ontologies in ESG, Symposium on Cluster Computing and the Grid (CCGrid 2003), Tokyo, Japan, May 12-15, 2003.
11. Protégé Project. <http://protege.stanford.edu/>
12. schraefel, m. c., Hughes, G., Mills, H., Smith, G., Payne, T. and Frey, J. (2004) Breaking the Book: Translating the Chemistry Lab Book into a Pervasive Computing Lab Environment. In Proceedings of CHI 2004, Vienna, Austria.
13. schraefel, m. c., Hughes, G., Mills, H., Smith, G. and Frey, J. (2003) Making Tea: Iterative Design through Analogy. Submitted to Proceedings of Designing Interactive Systems, 2004.