TAPoR: Five views through a text analysis portal (COCH/COSH Allied Association Session)

Geoffrey Rockwell *(georock@mcmaster.ca)*
McMaster University
Stéfan Sinclair *(sgsinclair@gmail.com)*
McMaster University
James Chartrand *(jc.chartrand@mcmaster.ca)*
OpenSky Solutions

A. Session Introduction

The *TAPoR* project started as a project to create a portal where users could manage texts, tools and then run tools on text. The Alpha version of the *TAPoR* portal nicely demonstrated the potential of this simple workbench paradigm. *TAPoR.2* builds on the individual project paradigm to make the portal useful for research communities. It does this in a number of ways:

1. We have developed a Try It first encounter interface for use by new users, casual users, and just-in-time users. This interface has been developed in close coordination with usability researchers, though it is now going into extensive testing.
2. *TAPoR.2* allows user information to be saved for groups or made public in a fashion similar to community information portals like *del.icio.us* (<http://del.icio.us>) and *CiteULike* (<http://www.citeulike.org>). Some types of information have always been intended for public viewing like the News built into *TAPoR* from the beginning. We have not only extended the sharing model to all types of information managed, but we have added communal editing to selected types of information, especially documentation, with a wiki editing-like interface.
3. We have extended the project paradigm to allow interfaces to be created that can be integrated into other projects and web sites. Thus advanced users can create projects that are styled to look like part of a different project.
4. We have developed a tool developers interface so that tools as web services can be added and documentation quickly entered. We have also used the community building features of the portal to develop *TA!DA!* or the *TAPoR Developers Association* – a site for the developer community.
5. We have developed *TEA*, the *TAPoR Engine of Association*, which is designed to help the serendipitous exploration of texts, references, links, people, projects and tools. *TEA* combs and visualizes topic maps which associate items across users.

In this session we are going to present the portal from five views that move from a conventional first encounter view of a tool portal to an inverted view of the portal as a research community association engine. These five views will be presented as three coordinated papers.

B1. TAPoR: First Encounters

Geoffrey Rockwell

The first paper will demonstrate the first encounter interface, Try It. Woven into this presentation will be a discussion of the usability research and testing that led to this interface hypothesis. It is our hope that this encounter interface will be of use to novices and advanced, but casual, users. It is an interface that doesn't require a portal account so it can be used occasionally and it is optimized for ease of use and successful results.

Rockwell will then demonstrate the basic user account paradigm for people who want to use the portal for sustained text analysis projects. He will demonstrate how from a first encounter once can get a myTAPoR account with which to organize links to texts, organize tools, and manage projects.

B2. TAPoR: Developing Encounters

Stéfan Sinclair

The second paper will demonstrate and discuss the Tool Developers interface and the community tools designed to assist developers. In this context Sinclair will discuss the first *TAPoR* "hackers ball" funded by the Social Science and Humanities Research Council of Canada through a grant led by Stéfan Sinclair. He will also discuss the technical design of the underlying tool broker and the data interfaces that allow results to be saved to a Data Bench for use as an input text for a different tool. This component of the presentation will end with a blatant attempt to enlist attendees in *TA!DA!* so we can enrich the tools collection.

The portal must bring together the text analysis community. In particular, the portal must make it as easy as possible for researchers who have existing tools, or want to write new tools — in their preferred programming language — to make the tools available through the portal. Web services provide a standard language and protocol to enable communication between different programming languages, and therefore are a very appropriate vehicle for connecting text analysis tools together through the portal. Further, most programming languages provide tools to publish existing program code as web services with little or no modification, and little extra setup. In some cases the tools will take an existing program function and create the entire infrastructure needed to make the function available over the internet: the web server, the code to listen for remote requests and translate them into calls to the local program code, and code to package the results up and return them to the original caller.

Text analysis tools provided as web services are easier to combine in simple ('piped') combinations, but can also be combined in very sophisticated arrangements (using scripting) — without requiring that the user learn new programming languages or run through elaborate setup procedures.

B3. TAPoR: Community Encounters

James Chartrand

The third paper will discuss the underlying technologies deployed in the portal so as to show how the portal can be rethought as a community association engine. We chose *Apache Cocoon* as our web development framework for the portal. *Cocoon* satisfies several of our objectives. *Cocoon* provides a basic portal implementation geared towards custom development. *Cocoon* is open source. Much of *Cocoon* is made up of code donated from large scale software projects; code that has gone through numerous development cycles on large systems. *Cocoon* is actively maintained and supported by hundreds of developers. *Cocoon* is therefore stable, secure, and scalable. In addition, *Cocoon* runs on Java and therefore, can run without modification on *Linux*, *Windows* and the *Mac*, allowing new projects to install the portal with ease.

The portal must provide a uniform and single point of access for text analysis tools, but must also engender an online community of knowledge. We chose Topic Maps for knowledge management because they are adaptable, simple, and standards based. Topic Maps can be thought of as a very rich index. An index that doesn't just point into texts, but can describe relationships between almost any object or idea. In our case, the relationships are between texts, between tools, between texts and tools, between projects, between projects and tools, between projects and users, between users and texts, and so on. Topic Maps also make the portal more adaptable to the needs of other projects outside the text analysis community.

In the context of underlying technologies James Chartrand will demonstrate the portal again, but now from the view-point of how it can be used to develop a research group or project taking advantage of the incorporated technologies. He will demonstrate the deep skinning features that allow users to create views that suit their research, their groups, or their projects. In this context he will illustrate how the *TAPoR* portal, is, from one perspective, just a web of associations between links, notes, tools, and topics.

C. Issues

There are a number of key issues that underlie all three papers.

i. Peer review of tools and academic credit. In a panel organized for the ACH/ALLC 2003 in Athens Georgia by Stéfan Sinclair on "Peer Review of Humanities Computing Software" we presented some models for how review of tools could be supported. *TAPoR* as a public portal that gives access to tools elsewhere that run as web services can be site for the review and documentation of software tools. We will present a documentation interface that allows public comments and reviews of tools that could serve some of the need for a peer review system.

ii. Open source. A popular paradigm for the creation and maintenance of community tools is to release them as open source under one of the various licenses available. We will discuss the way in which the portal as software is open source and the ways individual tools can be made available or protected. Likewise we will discuss the need for authentication for selected texts which cannot be made available openly.

iii. Humanities software development. The portal must, fundamentally, meet the needs of a research community. Needs which aren't, by definition, yet completely defined as research evolves. To that end, we have adopted an "agile" development process that involves regular meetings and storytelling. This approach has proven extremely effective. We have avoided getting bogged down in over-analysis and excessive documentation, and at same time have been able to adapt development cycles to meet the evolving needs of the project. Adaptability is particularly important for a research project like this where midstream research outcomes can lead to new paths, or close others.

iv. Stories. The 'story' is the fundamental unit of work in our process. Stories are informal descriptions of how the end-user would like to use the portal. Stories can be written in whatever style makes sense for the user. Stories and other documentation is kept in the *TAPoR* Wiki which is a shared development space. The stories are then broken down by the Open Sky Solutions team into 'tasks' that are assigned time estimates.

v. Adaptability. An important objective of the project is to enable other projects to adapt the portal and to contribute to its development. We have, therefore, organized the development process around standards that make it straightforward to not only download and install the portal, but to setup the development environment. Our goal is to ensure continued development of the portal.

D. Conclusions

The *TAPoR* Portal is fundamentally conceived of and designed to be an extensible, network-based research environment. As such, it has been crucial to devise mechanisms for enriching the portal by allowing developers and users to encounter the portal, use it, and adapt it for others. It is worth emphasizing how this approach differs from the development of text analysis tools of the past, such as *OCP* and *TACT*, that are essentially pre-defined workstation-based programs. *TAPoR*, by contrast, seeks to accommodate unknown and unanticipated resources. Such flexibility requires considerable engineering to ensure compatibility between disparate texts and tools. We will present a model for such flexibility, but recognize that it will need testing and scrutiny to become genuinely useful.