The *Edition Production Technology* (*EPT*) and the *ARCHway* and *Electronic Boethius* Projects

Kevin Kiernan (kiernan@uky.edu)

University of Kentucky, English

Dorothy Porter (dporter@uky.edu)

University of Kentucky, Research in Computing for Humanities

Alex Dekhtyar (dekhtyar@cs.uky.edu)

University of Kentucky, Computer Science

Ionut Emil Iacob (ionut@ms.uky.edu)

University of Kentucky, Computer Science

Jerzy W. Jaromczyk (jurek@cs.uky.edu)

University of Kentucky, Computer Science

Neil Moore (neil@s-z.org)

University of Kentucky, Computer Science

Session Statement

This session is based on the collaborative research and interdisciplinary education that have gone into the development of a generic *Edition Production Technology* (*EPT*) for building image-based electronic editions of damaged Old English manuscripts and, by extension, any representation of digitized cultural materials for contemporary users.

The concept of an effective, modular, extensible, *Edition Production Toolkit* (*EPT*) arose from the difficulties encountered while producing an electronic edition of the Beowulf manuscript. To solve these problems for the *Electronic Boethius* project, we set out to create a modular Java and XML software framework, including an edition production management system, a native XML database, graphical user interfaces, and a suite of editorial tools, customized to the needs of textual scholars in the humanities. The goal was to allow for the efficient assembly of complex scholarly editions from high-resolution digital facsimiles and XML-encoded texts, apparatus and ancillary materials. While our general approach was sound and the *Electronic Boethius* project in a few months developed a suite

of Java editorial tools operating under an XML framework, and successfully used these tools to begin the edition, the extensible development of the *Electronic Boethius* toolkit was hampered by the lack of computer science expertise in software engineering.

We were accordingly fortunate to attract computer scientists to join in the *ARCHway Project*, which deeply involved them and their students in an interdisciplinary effort to create an overarching technology for image-based electronic editions. Guided by the *Eclipse* programming environment, *ARCHway* has established an infrastructure for collaborative research and teaching between computer science and the humanities. Our interdisciplinary teams, working together at each stage, have designed formal methodologies for collaborative teaching and research, based on practical goals. *Eclipse*, our chosen programming environment, maintains an open-standards architecture with modular, extensible, interoperable components to coordinate research and development of novel methods, tools, and associated technologies in a teaching and learning environment involving undergraduate and graduate students. *EPT* has guided the definition and coordination of well-encapsulated collaborative student projects from semester to semester in specified research projects related to documenting, editing, storing, accessing, and searching image-based electronic editions.

The complementary projects allowed our research teams to pursue these shared goals from both a specific and practical standpoint, with the *Electronic Boethius* and the *Electronic Beowulf* projects, to a more general and theoretical standpoint, with the *ARCHway Project*. In the following papers, we first present how specific problems of preparing an electronic edition from damaged Old English manuscripts help to define the range of tools required in the *EPT*; we then show how the computer scientists designed and implemented an *EPT* with specific components crucial to image-based, document-centric editing; and finally, we present the *EPT*'s architecture, centering on the utilities that constitute its underlying infrastructure.

Using EPT to Build an Image-Based Electronic Edition of Alfred's Boethius

Kevin Kiernan and Dorothy Carr Porter

EPT is well suited to build an Image-Based Electronic Edition (IBEE) of Alfred the Great's Old English version of Boethius's *Consolation of Philosophy*. There are two surviving Old English manuscripts of this text, but they present the text in very different ways. The first, the tenth-century BL MS Cotton Otho A. vi, is the only prose and verse translation. The other complete manuscript is a later, twelfth-century, entirely prose version in Oxford, Bodleian Library MS, Bodley 180. There is also an indispensable, post-medieval source, however, in a seventeenth-century transcript and collation of the two

manuscripts by Francis Junius, an edition in the making now preserved in Oxford, Bodleian Library MS, Junius 12. In 1731 the earlier Cottonian MS was badly burned in the terrible Cotton Library fire, but ultraviolet discloses much of the seemingly lost text and Junius's transcripts and collations preserve most of the rest, while Bodley 180 provides critical variants. No modern editions have taken full advantage of these rich and diverse materials, and the two "standard" editions, supposedly based on Otho A. vi, respectively present a prose edition, stripped of the verse, and a verse edition, stripped of the prose. To provide a base text for XML encoding, the editor compiled a reconstructed version of Otho A. vi by reinserting the verse sections where they belong. At this point it is ready for *EPT*.

The purpose of an image-based edition is to reveal as openly and fully as possible the primary sources underlying the modern edition. Traditional print editions tend to conceal these sources by radically reformatting their structures, by providing modern punctuation, by underplaying their damaged states, by erasing their scribal peculiarities and semantic cruces through printed emendations and conjectural restorations, and by generally relegating the complex evidence the manuscripts hold to concise, uncomplicated, textual notes. An image-based edition makes these concessions, as well, to render an alien text accessible to today's readers, but it also provides ready access to the ultimate sources by linking, for example, all textual notes to the manuscript context.

In this presentation we will illustrate how the *EPT* provides the means for describing the manuscript using XML markup, and relating the folio and areas of the folio to the text that resides on that folio. The *EPT* enables us to associate images and sections of images with the relevant markup (folios with folio markup, damaged areas with damage markup, letters with markup describing the letter form, etc.), while at the same time associating text with whole or portions of single images, or multiple images of the same folio taken under different lighting conditions (daylight, ultraviolet, fiberoptic).

By providing support for pervasive, complex, image-based encoding, the *EPT* inevitably exacerbates the problem of overlapping markup. Iacob and Dekhtyar address the *EPT*'s approach to this problem in greater detail in this session; here it is enough to say that the *EPT* does support overlapping markup using multiple DTDs, and that if the DTDs are well-designed the humanities scholar need not worry about the semantics of XML markup, and can concentrate instead on the main tasks of editing, such as the semantics of an Old English text.

The *EPT* includes three fundamental tools for image-based encoding. The *ImagText* tool, working in cooperation with *xMarkup* and *xTagger* tools, supports general image-based encoding, allowing the editor to tag any element defined in the DTDs. *xMarkup* reads the DTDs and automatically creates templates that the editor configures, assigning meaningful or otherwise convenient aliases for elements, attributes, and attribute values, and arranging the elements into logical editorial groups, such as Start Edition, Condition, Codicology, Paleography, Restoration, with all their subsets of tags, which will very likely differ from the organization of the DTDs. Through *ImagText*, the editor views images side-by-side with the corresponding text (viewed through *xTagger*), and describes them with reference to one another using the templates provided by xMarkup. Otho A. vi has suffered severe damage, both from fire and from later preservation attempts. Thus, much of the tagging relates to manuscript condition, highlighting where the manuscript is damaged and linking it with the transcript or edition. This information ensures that the final edition will show clearly what text in the edition comes directly from the manuscript, what text is slightly damaged, and what text is damaged to the point of illegibility and thus either copied from another manuscript or otherwise restored by the editor.
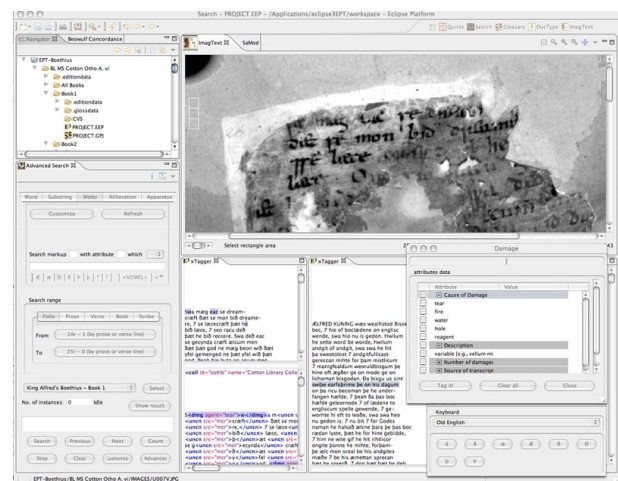

Figure 1. A snapshot of EPT illustrating image-based encoding through ImagText, xMarkup, and xTagger (including an XML view). The figure also shows the Keyboard panel and Search Tool.

The OverLay allows the editor to encode the text in reference to multiple images of the same folio. Images taken under special conditions, such as ultraviolet fluorescence, often provide clearer textual evidence than those digitized in natural light. Through OverLay an editor can examine minute differences between digital images, laying one image on top of another, selecting a section of the image, and using a slidebar to change the transparency of the top layer, moving between them. The editor can save the combined images created by OverLay and open them in ImagText, marking them up there as they would any other image.

The DucType is an example of a markup template that has been configured to deal with the specific encoding problem of describing letter forms. Paleography, the study of handwritten texts, is central to the scholarship of medieval manuscripts. Description of letterforms and the style of individual scribes

have traditionally been limited to general descriptions in manuscript catalog entries, or in introductions to manuscript facsimiles. Using XML, we are now able to incorporate paleographical description into the edition content character-by-character, providing individual letters with their own descriptions. This advanced markup will enable users of the finished edition to search for letters based on specific characteristics.

Specialized templates require specialized configuration. The editor configures the Letter template using the Letter Template tool, through which he assigns meaningful aliases to element and attribute names, adding attribute values as he discovers new letterforms in the manuscript. The Letter Template tool also enables the editor to clip and save sample letters, which the editor can reference later, comparing them to other letterforms in the manuscript.

Although the *EPT* enables the editor to practice image-based electronic editing, it is the DTDs that provide the underlying structure for describing the manuscript. We design our DTDs as extensions of TEI, defining some new elements and adding new attributes to existing TEI elements. We will illustrate how we are using TEI for image-based encoding, and how our extensions allow for a more complete manuscript description than TEI alone. We will discuss how we adapt TEI elements for image based encoding, and we will also describe our new attributes, which assist the *EPT* in supporting links between text and image. We will also introduce some our new elements, including `<offset>`, an empty element which marks an area in the manuscript where text from the facing page has bled onto the folio, in cases obscuring the manuscript text, and `<offsettext>`, which marks the text on the facing page corresponding to the offset. The `<offset>` and `<offsettext>` regions can then be compared using OverLay. We will also discuss our markup for paleographical description and the restoration of text visible under special lighting, not visible under regular lighting.

The *Electronic Boethius* Project is funded by a Collaborative Research Award from the National Endowment for the Humanities and the Andrew W. Mellon Foundation, and is sponsored by The British Library and the Bodleian Library, Oxford, who are providing digital images of the relevant documents. We are working in collaboration with the complementary print-based *Alfredian Boethius* project at Oxford, directed by Malcolm Godden.

Bibliography

Primary Sources

British Library MS Cotton Otho A. vi.

Oxford Bodleian Library MS Bodley 180.

Oxford Bodleian Library MS Junius 12.

Editions

Krapp, George Philip, ed. *The Paris Psalter and the Meters of Boethius.* The Anglo-Saxon Poetic Records 5. New York: Columbia University Press, 1932.

Robinson, Fred C., and E.G. Stanley, eds. *Old English Verse Texts from Many Sources: A Comprehensive Collection.* Early English Manuscripts in Facsimile 23. Copenhagen, Denmark: Rosenkilde and Bagger, 1991.

Sedgefield, Walter J., and E.G. Stanley, eds. *King Alfred's Anglo-Saxon Version of Boethius, de Consolatione Philosophiae.* Oxford: Clarendon Press, 1899.

Secondary Sources

Bauman, Syd, and Terry Catapano. "TEI and the Encoding of the Physical Structure of Books." *Computers and the Humanities* 33 (1999): 113-127.

Clark, James, ed. *XSL Transformations (XSLT) 1.0.* W3C Recommendation, 16 November 1999. Accessed 2005-04-07. <http://www.w3.org/TR/xslt>

Johansson, Karl Gunnar. "Computing Medieval Primary Sources from the Vadstena Monastery: Arguments for the Primary Source Text." *Literary and Linguistic Computing* 19.1 (2004): 93-104.

Kiernan, Kevin. "Image-based Electronic Editing of Alfred the Great's Boethius." *Making Sense: Constructing Meaning in Early English.* Ed. Antonette diPaolo Healey and Kevin Kiernan. Forthcoming. (In progress, expected Richard Rawlinson Center Series, Medieval Institute Press, 2006.)

Kiernan, Kevin, Jerzy W. Jaromczyk, Alex Dekhtyar, Dorothy Carr Porter, Kenneth Hawley, Sandeep Bodapati, and Ionut Emil Iacob. "The ARCHway Project: Architecture for Research in Computing for Humanities through Research, Teaching, and Learning." *Literary and Linguistic Computing* (Forthcoming in 2005). (Special issue, papers from Joint International Conference of the Association for Computers and the Humanities and the Association for Literary and Linguistic Computing, 2003.)

Kiernan, Kevin. "The nathwylc Scribe and the Beowulf Palimpsest." *Poetry, Place and Gender: Studies in Medieval Culture in Honor of Helen Damico.* Ed. Catherine E. Karkov and Nancy van Deusen. Kalamazoo, MI: Medieval Institute Press, Forthcoming in 2005.

Kiernan, Kevin, W. Brent Seales, and James Griffioen. "The Reappearances of St. Basil the Great in British Library MS Cotton Otho B. x." *Computers and the Humanities* 36 (2002):

7-26. (Image-based Humanities Computing. ed. Matthew Kirschenbaum.)

Kiernan, Kevin. "Digital Facsimiles in Editing: Some Guidelines for Editors of Image-based Scholarly Editions." *Electronic Textual Editing.* Ed. John Unsworth, Katherine O'Brien O'Keeffe and Lou Burnard. Modern Language Association and the TEI Consortium, 2005.

Kiernan, Kevin. "Odd Couples in Ælfric's Julian and Basilissa in British Library Cotton MS Otho B. x." *Beatus vir: Studies in Anglo-Saxon and Old Norse Manuscripts in Memory of Phillip Pulsiano.* Ed. Kirsten Wolf and A.N. Doane. Tempe, AZ: Medieval and Renaissance Texts and Studies (MRTS), Forthcoming in 2005.

Lecolinet, Eric, Laurent Robert, and Francois Role. "Text-image Coupling for Editing Literary Sources." *Computers and the Humanities* 36 (2002): 49-73. (Image-based Humanities Computing. ed. Matthew Kirschenbaum.)

Prescott, Andrew. "'Their Present Miserable State of Cremation': The Restoration of the Cotton Library." *Sir Robert Cotton as Collector: Essays on an Early Stuart Courtier and His Legacy.* Ed. C.J. Wright. British Library Publications, 1997. 391-454.

Seales, W. Brent, James Griffioen, Kevin Kiernan, C.J. Yuan, and Linda Cantara. "The Digital Atheneum: New Technologies for Restoring and Preserving Old Documents." *Computers in Libraries* 20.2 (February 2000): 26-30. Accessed 2005-04-07. <http://www.infotoday.com/cilmag/feb00/seales.htm>

Sperberg-McQueen, C.M., and Lou Burnard, eds. *Guidelines for Electronic Text Encoding and Interchange; XML-compatible edition.* Chicago and Oxford: TEI P4, 2001. XML conversion by Syd Bauman, Lou Burnard, Steven DeRose, and Sebastian Rahtz.

Unsworth, John. "Reconsidering and Revising the MLA Committee on Scholarly Editions' Guidelines for Scholarly Editions." *Panel on "New Directions for Digital Textuality." 2001 Conference of the Society for Textual Scholarship.* 19 April 2001. Accessed 2005-04-07. <http://www.iath.virginia.edu/~jmu2m/sts2001.html>

Yergeau, Francois, Tim Bray, Jean Paoli, C.M. Sperberg-McQueen, and Eve Maler, eds. *Extensible Markup Language (XML) 1.0 (Third Edition).* W3C Recommendation, 4 February 2004. Accessed 2005-04-07. <http://www.w3.org/TR/2004/REC-xml-20040204//>

Building Tools for Image-Based Electronic Editions

Alex Dekhtyar and Ionut E. Iacob

The *EPT* serves to organize the raw materials of digital scholarship – digital image and text files – and, using specialized encoding, builds these materials into a usable electronic edition. This edition will include a wide variety of editorial information, including organizational description of both physical (books, folios, lines) and semantic (sentences, words), glossarial and metrical description, and description of the condition of the physical object, notably how that condition interacts with the text on the page. For this reason, it is vital for a successful IBEE that the *EPT* enable the editor to create links between the images and text.

The eXtensible Markup Language (XML) is preferred by the humanities computing community as data support for electronic text encoding, most notably through Guidelines of the Text Encoding Initiative. Although XML does not well capture complex text structures (its strict hierarchical organization severely limits its usefulness in describing, for example, both physical and textual organization in a single file), its relative simplicity recommends it over more powerful but complex representations. Moreover, XML is well supported by software processing tools, from databases, parsers and editors (supporting syntax coloring and on-the-fly validation) to query engines and XML transformations. Many good XML editors are available at no, or very low, cost, which makes XML an even more attractive choice for humanities text encoding.

Building an electronic edition is a tedious enterprise. The editor using traditional XML software must encode editorial information while remaining mindful of XML syntax and the limits imposed by its use. A misplaced tag can keep an XML file from validating, and often an editor will have to choose between encoding different aspects of the manuscript text or risk overlapping markup (for example, the physical organization of a folio – the lines as they appear on the page – may conflict with the sentence structure of the text). Things become more complicated when images are involved. The editor has to keep track of images and record relationships between text and image, not just relating entire folios to the text on that folio, but identifying corresponding regions of text and image. The unfortunate result of this process is that as the complexity of the encoding increases, the editor must concentrate on the syntax of encoding rather than on the details of the text of the manuscript or edition. Our goal was to design tools that allow the editor to concentrate on the act of editing, rather than focus on issues of XML syntax and validity.

As James Clark points out, there are two main classes of XML editors: *text editors* and *structural editors*. The key difference between these two kinds of editors is the way markup is

introduced. Structural editors focus on data-centric encoding, and the editing process begins with markup. The human editor adds content to an encoding template, in a manner similar to entering items in a database. This is in contrast to text editors, which focus on document-centric editing and begin with the textual content (`PCDATA`). The editor inserts markup into (or deletes it from) the content one tag at a time. The text editor approach is much preferable for humanities editing in general and image-based encoding specifically, as it gives the human editor control over exactly what markup is entered where in the text. This control is important for image-based editing, as it facilitates the recording of image-text relationships by allowing the human editor to select specific sections of text and, with the right software support, relate that text to the corresponding sections of image. Another issue that arises in document-centric encoding is that the XML document may not be valid during the editing process: the order in which the editor introduces the markup in the text may depend not on the requirements of the DTD, but rather on the modus operandi of the human editor (which in turn depends on the semantics of the features to be encoded).

Thus, an image-based XML editor has to have the following features:

- Hide the XML syntax if requested. The focus of the human editor should be on text semantics and how images and text are connected. Instead of displaying the complete XML, show where markup exists by highlighting the relevant text in the display. The editor may at times wish to examine the XML encoding. In that case, the XML editor should provide a system for filtering out unwanted markup, showing only those elements that the editor wishes to see.

- Allow text markup by enabling the editor to select the range of content to be marked up and the tag (and attribute values) to be inserted. Among tag attributes, at least one is dedicated to link text and corresponding image or image region.

- Provide support for the editor to connect the markup with the corresponding manuscript image and a specific region in the image. While the editor selects the related areas, the information for mapping the image to the text should be saved automatically by the software — the editor should not have to concern himself with creating image maps or noting image coordinates.

- Assure document well-formedness and provide support for (partial) validation in such a way that it is transparent to the human editor. Imposing validity constraints for update operations might be too prohibitive in text encoding applications: not every update operation (or a set of consecutive update operations) yields a valid document. The software takes further update decisions based on the current status of a document. At the same time, it is important to be able to verify at each moment of time that the current XML

fragment is 'on track', i.e., that the human editor has not committed any structural error while introducing the markup (in which case markup deletion is required). We call this *potential validation* and we designed and implemented an algorithm for checking potential validity of document-centric XML documents.

- Provide support for searching for both text and structure, and for searching the encoding of image features described in the XML markup. There are three main types of searches that the editor can perform in an IBEE. First, the text search, through which the editor can search for a string of characters in the edition content. Second the structural search – this information describes how various text and image features are interrelated (words in certain lines or sentences, holes on the folio in the middle of sentences, etc.). And finally, image feature searches. Given a specified region on the image, the software will find all encoded features related to that region or, conversely, will find all image regions corresponding to a given text range or descriptor (for example, find all image regions with corresponding damage markup).

The architecture of our image-based XML editor is presented in Figure 2.
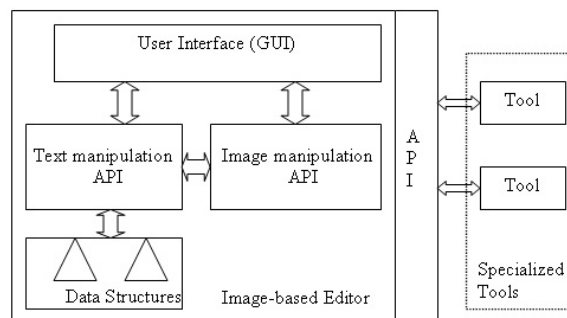

Figure 2. Image-based XML editor

In this paper, we will describe how we designed and built the *EPT* and its individual components to incorporate those elements that we found most important for image-based, document-centric editing.

Bibliography

Brown, Michael S., and W. Brent Seales. "The Digital Atheneum: New Approaches for Preserving, Restoring, and Analyzing Damaged Manuscripts." *Proceedings of the First ACM/IEEE-CS Joint Conference on Digital Libraries.* New York: ACM Press, 2001. 437-443.

Brown, Michael S., W. Brent Seales, Kevin Kiernan, and James Griffioen. "3D Acquisition and Restoration of Medieval

Manuscript." *Communications of the ACM: Special Issue on Digital Libraries.* May 2001.

Clark, James. "Incremental XML Parsing and Validation in a Text Editor." Presentation at XML 2003, Philadelphia. December 2003.

Hayes, Deborah. "Glossing Damaged Manuscripts: an Example from Ælfric's Lives of Saints." Presentation at Digital Resources for the Humanities (DRH01). University of London, London, UK. 10 July 2001.

Kiernan, Kevin, Alex Dekhtyar, Jerzy W. Jaromczyk, Dorothy Carr Porter, and Ionut Emil Iacob. "Edition Production Technology (EPT) and the ARCHway Project." *DigiCULT.Info* 8 (August 2004): 36-38.

Seales, W. Brent, James Griffioen, Kevin Kiernan, C.J. Yuan, and Linda Cantara. "The Digital Atheneum: New Technologies for Restoring and Preserving Old Documents." *Computers in Libraries* 20.2 (February 2000): 26-30. Accessed 2005-04-07. `<http://www.infotoday.com/cilmag/feb00/seales.htm>`

Sperberg-McQueen, C.M., and Lou Burnard, eds. *Guidelines for Electronic Text Encoding and Interchange.* Chicago and Oxford: TEI P4, 2001.

Yergeau, Francois, Tim Bray, Jean Paoli, C.M. Sperberg-McQueen, and Eve Maler, eds. *Extensible Markup Language (XML) 1.0 (Third Edition).* W3C Recommendation, 4 February 2004. Accessed 2005-04-07. `<http://www.w3.org/TR/2004/REC-xml-20040204//>`

Yuan, C.J., and W. Brent Seales. "Guided Linking: Efficiently Making Image-to-Transcript Correspondence." *Proceedings of the First ACM/IEEE-CS Joint Conference on Digital Libraries.* New York: ACM Press, 2001. 471. Accessed 2005-04-07. `<http://www.infotoday.com/cilmag/feb00/seales.htm>`

The ARCHway Software Infrastructure: a platform and utilities for building electronic editions

Jerzy W. Jaromczyk and Neil Moore

In this paper we will discuss the implementation of *EPT*'s architecture, specifically focusing on those utilities that form its underlying skeleton or infrastructure. These utilities support the consistent management of diverse sources of data while providing an extensible framework for building and organizing the editorial tools discussed in the previous papers.

The *EPT*'s architecture is based on the plugin, an encapsulated and independent software unit that "plugs into" a larger whole, extending its functionality. An editorial workbench built from many individual tools gives the editor the freedom to pick and choose tools for the tasks at hand. We selected *Eclipse* as the platform for both the development and deployment of the *EPT* because it seemed it would well support such a free and configurable approach to the design of the editorial workbench. In this presentation we will briefly describe the *Eclipse* platform and the functionality and implementation of a selection of *EPT* plugins that provide the infrastructure for accessing data, organizing and annotating resources and defining different models for electronic editions.

Eclipse is an open-source platform designed to serve as an extensible integrated development environment (*Eclipse Platform Technical Overview.*). Originally developed by IBM, *Eclipse* is now maintained by the Eclipse Foundation and an extensive user's community. Although initially intended for software development, *Eclipse*'s open-ended plugin-based architecture allows users to extend it to support an unlimited variety of other tasks including software deployment.

The organization of *Eclipse* is a collection of plugins: loosely-coupled software components, often developed independently of one another, which communicate with each other and with which users can interact using well-defined interfaces. Much like the more familiar web browser plugins (such as the Macromedia *Flash* plugin, Sun's *Java* plugin, and Adobe's *Acrobat Reader* plugin), *Eclipse* plugins 'hook into' so-called extension points defined elsewhere in the application, extending and enhancing the application's existing functionality as well as adding completely new features. However, *Eclipse* differs notably from most other extensible software. Most significantly, the platform itself is built from scores of plugins which themselves extend other plugins and provide additional extension points; almost everything in *Eclipse* is a plugin. This may be contrasted with, for example, web browsers, where plugins extend an underlying monolithic software system and rarely interact with one another.

The extensibility of the plugin architecture will be a tremendous benefit to the users of the *EPT*. Humanities researchers have various needs and editorial styles; with a plugin system, they can create a personal editing workbench containing only the tools they need, without losing the advantages of a coherent interface and uniform access to data. Furthermore, scholars with specific needs unforeseen by the *EPT* developers may collaborate with programmers to develop their own editing tools or modify existing ones. The *EPT*'s plugin architecture allows users to develop new tools separately from and independent of the *EPT* and then plug them into the *EPT* extension points, providing a seamlessly integrated experience. In effect, such plugins become an integral part of a customized version of *EPT*, on equal footing with the many tools that make up the *EPT* proper.

The *EPT* organizes its plugins in a series of layers, with each layer building upon, using, and extending the layers below. We

will discuss three of these layers in our presentation. The bottommost layer is called the Data Layer. The plugins making up the Data Layer provide a consistent set of operations for managing, reading, and storing various types of edition data files such as images, configuration files, textual transcripts, marked-up edition documents, and XML document type declarations (DTDs). The Data Layer provides a single interface for accessing all data, regardless of where that data is stored — in the local file system, a database (see Dekhtyar et al.), or a remote site. Plugins called data source drivers extend the Data Layer by providing functionality to access resources through different means. Currently the *EPT* contains two data source drivers, one for accessing files located within the file system of the computer running the *EPT*, and one for accessing resources stored on a remote web server, using the HTTP protocol for the World Wide Web. The flexibility of the Data Layer framework allows the user to implement a wide variety of data source drivers; users could build drivers that transparently compress or encrypt data, drivers to maintain data in a relational database, and many others.

On top of the Data Layer sits the Project Explorer, which provides a higher-level view of the resources comprising an electronic edition project. Project Explorer provides a user interface for viewing and managing the logical structure — the model — of a project (as opposed to its physical structure, as an edition project may take its components from several different data repositories). Project Explorer provides a rich set of extension points so that other plugins may contribute resources to the Explorer view. Such contributions provide various actions (e.g., launch a tool, display an image, etc.), which operate on the model, and on the resources themselves.

The Resource Registry, built atop the Data Layer and Project Explorer, enables the user to organize, categorize, and manage collections of similar resources. For example, one collection might contain all the manuscript images comprising the electronic edition. Each collection has a schema, a list of attributes applicable to all resources in the collection. The editor defines collections and their schemas in the Resource Registry, and then adds resources to those collections, describing them by specifying the attribute values for each resource. For example, the schema for manuscript images might contain attributes describing the folio name (038v, for example); the image format (JPEG, GIF, TIFF, etc.); the type of lighting used when digitizing the image (e.g., overhead white light, ultraviolet light, or fiber-optic backlight); the provenance of the image files; and so forth. The Resource Registry contributes items to the Project Explorer, arranging the resources by a user-defined ordering of their attributes. Other plugins can issue queries to the Resource Registry asking for resources with certain attributes, for example all the ultraviolet manuscript images which are in the JPEG format.

The utilities described above form part of the infrastructure for the *EPT*. They provide a workbench within which a user can arrange various specialized tools, such as ones described in the other papers in this session, in convenient combinations capable of solving complex tasks in the production and presentation of image-based electronic editions.

Bibliography

Dekhtyar, Alex, et al. "Database Support for Image-based Electronic Editions." *Proceedings, 10th International Workshop on Multimedia Information Systems (MIS 2004), August 25–27, 2004, College Park, MD.* 2004. 147-156.

Eclipse. Accessed 2005-04-12. <http://www.eclipse.org/>

Eclipse Platform, Technical Overview. Accessed 2005-04-12. <http://www.eclipse.org/whitepapers/eclipse-overview.pdf>

Jaromczyk, Jerzy W., and Sandeep Bodapati. "An Architecture Promoting Collaborative Research, Teaching and Learning." *Proceedings, Joint International Conference of the Association for Computers and the Humanities and the Association for Literary and Linguistic Computing, May 29–June 2, 2003, Athens, GA.* 2003. 10.

Jaromczyk, Jerzy W., and Neil Moore. "Geometric data structures for multihierarchical XML tagging of manuscripts." *Proceedings, 20th European Workshop on Computational Geometry, Seville, Spain, March 2004.* 2004. Accessed 2005-04-14. <http://www.us.es/ewcg04/Articulos/jaromczyk.ps>

Kiernan, Kevin, Jerzy W. Jaromczyk, Alex Dekhtyar, Dorothy Carr Porter, Kenneth Hawley, Sandeep Bodapati, and Ionut Emil Iacob. "The ARCHway Project: Architecture for Research in Computing for Humanities through Research, Teaching, and Learning." *Literary and Linguistic Computing* (Forthcoming in 2005). (Special issue, papers from Joint International Conference of the Association for Computers and the Humanities and the Association for Literary and Linguistic Computing, 2003.)