Journal of Digital Imaging

The Open-Source Neuroimaging Research Enterprise

Daniel S. Marcus, Kevin A. Archie, Timothy R. Olsen, and Mohana Ramaratnam

While brain imaging in the clinical setting is largely a practice of looking at images, research neuroimaging is a quantitative and integrative enterprise. Images are run through complex batteries of processing and analysis routines to generate numeric measures of brain characteristics. Other measures potentially related to brain function – demographics, genetics, behavioral tests, neuropsychological tests – are key components of most research studies. The canonical scanner – PACS – viewing station axis used in clinical practice is therefore inadequate for supporting neuroimaging research. Here, we model the neuroimaging research enterprise as a workflow. The principal components of the workflow include data acquisition, data archiving, data processing and analysis, and data utilization. We also describe a set of open-source applications to support each step of the workflow and the transitions between these steps. These applications include DIGITAL IMAGING AND COMMUNICATIONS IN MEDICINE viewing and storage tools, the EXTENSIBLE NEUROIMAGING ARCHIVE TOOLKIT data archiving and exploration platform, and an engine for running processing/analysis pipelines. The overall picture presented is aimed to motivate open-source developers to identify key integration and communication points for interoperating with complimentary applications.

KEY WORDS: Biomedical imaging, neuroimaging, neuroinformatics, DICOM, XNAT, research workflow

INTRODUCTION

Biomedical imaging research is a complex endeavor that involves many human and software components. It includes a number of unique requirements that make typical clinical picture archiving and communication system (PACS) and viewing tools inadequate. Neuroimaging research, in particular, demands extensive custom support. Digital imaging and communication in medicine (DICOM) images, the standard format for clinical systems, for example, are converted in most laboratories to research formats like Analyze, NIfTI, and MINC. These images are then run through a battery of processing routines: distortion and inhomogeneity correction, co-alignment, registration into a common atlas space, segmentation, and generation of quantitative measures and statistics relevant to the experimental study. Images generated by these routines are then viewed using tools with unique capabilities to present the images in a meaningful way and to allow user interaction with them. Finally, research studies generally include a range of nonimaging measures (eg, genetics, clinical assessments, neuropsychometric batteries) that must be integrated with the image data.

It is a testament to the vitality of the open-source movement that open-source applications have been developed to meet most if not all of the software requirements for running the neuroimaging research enterprise. Not surprisingly, many of these applications are themselves built using open-source components. Open-source applications are ideal in the research environment because they can be vetted for accuracy and tailored to suit specific laboratory needs. The aims of this paper are to present a conceptual framework that facilitates

From the Department of Radiology, Washington University School of Medicine, 4525 Scott Ave., Campus Box 8225, St. Louis, MO 63110, USA.

Correspondence to: Daniel S. Marcus, 4525 Scott Ave., Campus Box 8225, St. Louis, MO 63110, USA; tel: +1-314-3624562; Fax: +1-314-3626110; e-mail: dmarcus@wustl.edu

Online publication 21 August 2007
doi: 10.1007/s10278-007-9066-z

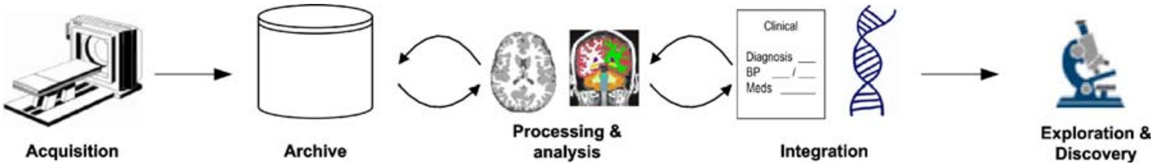


Fig 1. The neuroimaging enterprise workflow.

approaches to integrating the various open-source tools into a seamless enterprise system and to describe a specific example set of open-source tools within the context of this framework. Other tools, such as the many listed on neuroscience tool registries,[1,2] can similarly be integrated into this framework.

THE NEUROIMAGING RESEARCH ENTERPRISE WORKFLOW

Here we model neuroimaging research as a workflow. The advantage of viewing the enterprise as a workflow is that it allows one to consider each stage in the workflow as a software component and to consider the transition between stages as a communication between components. The neuroimaging data workflow as we have modeled it is illustrated in Figure 1. The workflow begins with data acquisition at the scanner. Image files are then transferred to a data archive. In research imaging, as part of the transfer, the image files are often "deidentified" to support patient privacy requirements. Legacy data that are no longer at the scanner can also be transferred into the archive. The next step is for researchers to mark-up the data with annotations and qualitative assessments. Traditionally, this has been the domain of the paper lab notebook, but electronic media are quickly becoming the preferred approach. The image data are then made available for manual and/or automated image processing routines. Resulting quantitative measures and derived images are stored to a database and integrated with nonimaging measures (eg, clinical assessments, genetics). Finally, researchers use discovery and productivity tools to interact with the integrated database. It is worth noting that the workflow is largely unidirectional but that, as derived data are generated during analysis, these data – like the original raw data – enter the workflow at the data mark-up stage.

In the following sections, we detail each stage of the workflow and describe specific open-source tools (summarized in Table 1) that support each stage and, as important, communication between each stage. Whereas our presentation focuses on tools developed in our group, a main benefit of the workflow model is that one could easily replace specific components with other open-source products that may be more appropriate or desirable in other environments. Indeed, in the open-source enterprise, this is particularly crucial because it is unlikely that a single application will meet all of the requirements of the enterprise.

Table 1. Open-source Tools for the Neuroimaging Research Workflow

Application	Workflow Step	Web Site	Core Open-source Components
DicomBrowser	Data capture	http://nrg.wustl.edu/projects/DICOM	dcm4che (http://www.dcm4che.org), ImageJ (http://rsb.info.nih.gov/ij)
DicomServer	Data capture	http://nrg.wustl.edu/projects/DICOM	dcm4che, MIRC (http://mircwiki.rsna.org), Apache FTP (http://incubator.apache.org/ftpserver)
XNAT	Archive, integration	http://www.xnat.org	Turbine (http://turbine.apache.org), Tomcat (tomcat.apache.org), PostgreSQL (http://www.postgresql.org), Velocity (http://velocity.apache.org)
PipelineRunner	Processing and analysis	http://nrg.wustl.edu/projects/pipeline	Saxon (http://sourceforge.net/projects/saxon), XMLBeans (http://xmlbeans.apache.org/)
PlexiViewer	Exploration	http://nrg.wustl.edu/projects/viewer	ImageJ

Data Acquisition and Capture

Despite the academic laboratory origin of MR and positron emission tomography technology, the modern scanner is a high-cost, highly proprietary piece of hardware; it is the one component in the neuroimaging research enterprise that is not amenable to open-source development. Nonetheless, scanner manufacturers have migrated from closed in-house data formats to DICOM, an open industry standard,[3] creating a bounty of opportunity for open-source development of ancillary applications. The DICOM standard is a massive and comprehensive standard that includes specifications for image file formats, data transport, printing, worklists, and querying. Most open-source DICOM tools deal with a subset of the standard. We have built open-source tools that focus on the capabilities that are essential for the research enterprise: receiving images, visualizing images, and accessing/editing image header content. These tools are built on the open-source dcm4che[4] library.

DicomServer was built to support communication between the scanner and data archive. DicomServer captures files sent over the DICOM transport protocol, organizes the received files by study, and parses metadata from the file headers, including where the images were acquired (eg, scanner name, manufacturer, and model), what types of images were acquired (eg, T1, DWI, EPI),

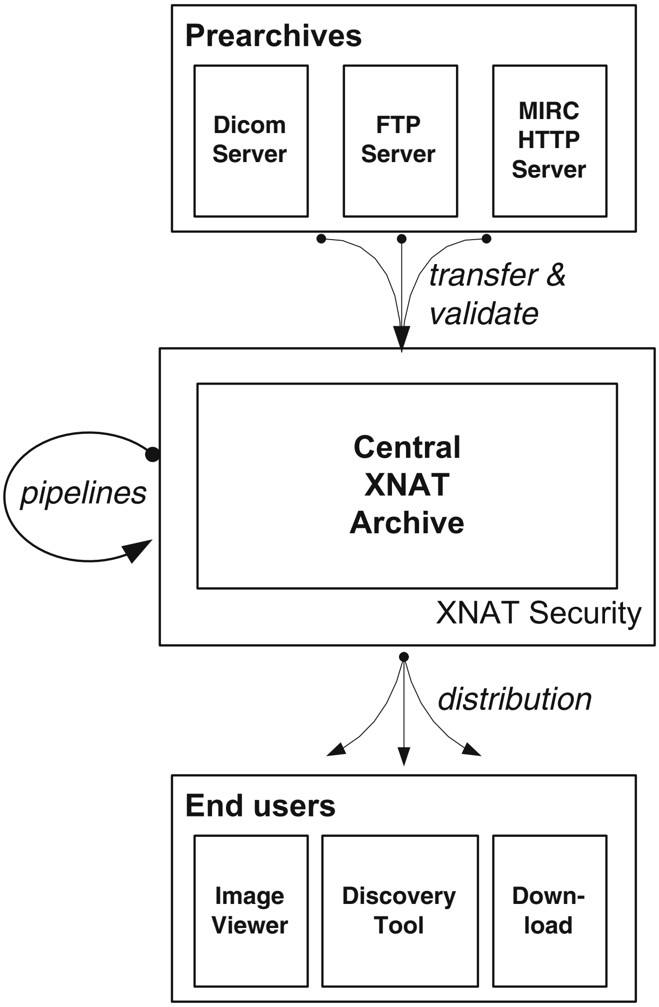
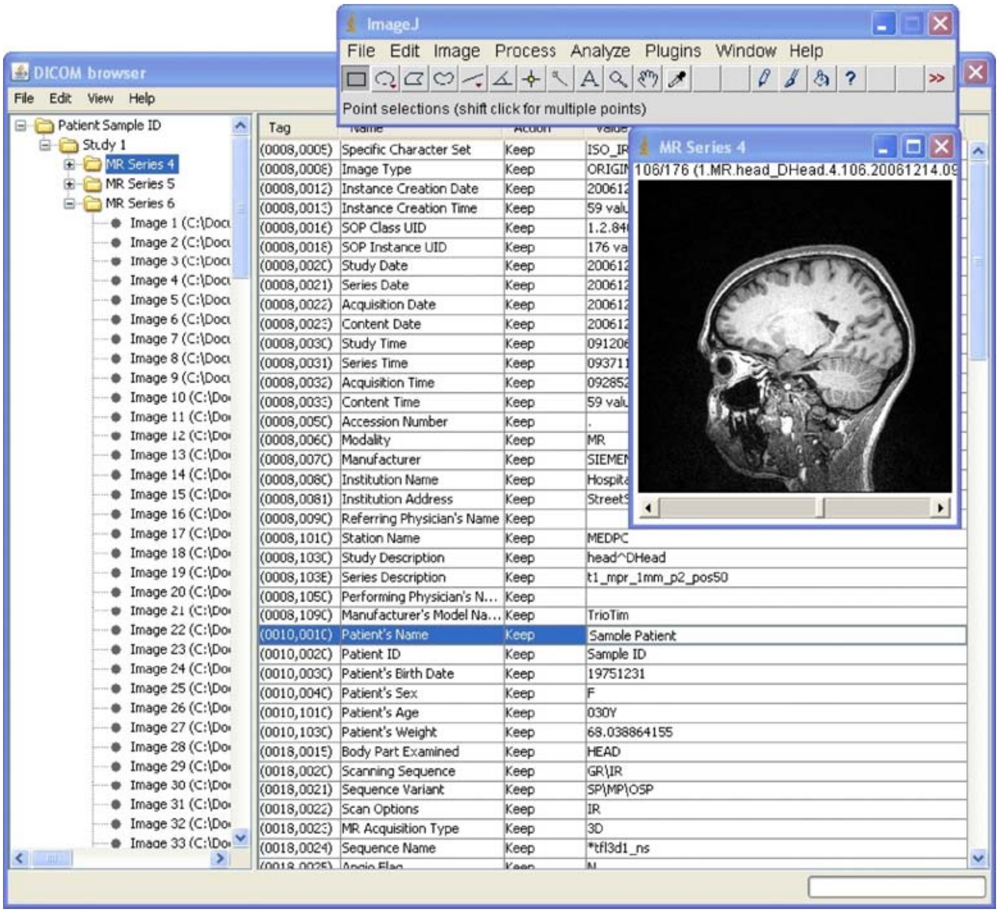


Fig 2. The data capture tools place incoming images into "prearchives" that can be accessed by data archive applications. The archive application securely stores the images and distributes them to various users and applications.



Tag	Name	Action	Value
(0008,0005)	Specific Character Set	Keep	ISO_IR
(0008,0008)	Image Type	Keep	ORIGIN
(0008,0012)	Instance Creation Date	Keep	200612
(0008,0013)	Instance Creation Time	Keep	59 valu
(0008,0016)	SOP Class UID	Keep	1.2.84
(0008,0018)	SOP Instance UID	Keep	176 va
(0008,0020)	Study Date	Keep	200612
(0008,0021)	Series Date	Keep	200612
(0008,0022)	Acquisition Date	Keep	200612
(0008,0023)	Content Date	Keep	200612
(0008,0030)	Study Time	Keep	091206
(0008,0031)	Series Time	Keep	093711
(0008,0032)	Acquisition Time	Keep	092852
(0008,0033)	Content Time	Keep	59 valu
(0008,0050)	Accession Number	Keep	.
(0008,0060)	Modality	Keep	MR
(0008,0070)	Manufacturer	Keep	SIEMEN
(0008,0080)	Institution Name	Keep	Hospita
(0008,0081)	Institution Address	Keep	StreetS
(0008,0090)	Referring Physician's Name	Keep	
(0008,1010)	Station Name	Keep	MEDPC
(0008,1030)	Study Description	Keep	head^DHead
(0008,103E)	Series Description	Keep	t1_mpr_1mm_p2_pos50
(0008,1050)	Performing Physician's N...	Keep	
(0008,1090)	Manufacturer's Model Na...	Keep	TrioTim
(0010,0010)	Patient's Name	Keep	Sample Patient
(0010,0020)	Patient ID	Keep	Sample ID
(0010,0030)	Patient's Birth Date	Keep	19751231
(0010,0040)	Patient's Sex	Keep	F
(0010,1010)	Patient's Age	Keep	030Y
(0010,1030)	Patient's Weight	Keep	68.038864155
(0018,0015)	Body Part Examined	Keep	HEAD
(0018,0020)	Scanning Sequence	Keep	GR\IR
(0018,0021)	Sequence Variant	Keep	SP\MP\OSP
(0018,0022)	Scan Options	Keep	IR
(0018,0023)	MR Acquisition Type	Keep	3D
(0018,0024)	Sequence Name	Keep	*tfl3d1_ns
(0018,0025)	Angio Flag	Keep	N

Fig 3. DicomBrowser allows users to view and deidentify DICOM image files and to send them to DICOM storage providers. Deidentification can be done manually by entering values into the appropriate header fields or automatically using script files.

and what acquisition parameters were used (eg, repetition time, echo time). It stores the received images as files in a "prearchive" on the file system (Fig. 2) and generates an XML document that describes the received files and study content. In contrast to most clinical PACS, which are entirely DICOM-based, these steps in effect neutralize the DICOM images, making them available to general-purpose file operations, databases, and XML tools. This allows the data archive component to more easily accommodate the broader range of data formats and software applications utilized in neuroimaging research.

DicomBrowser is a general-purpose tool for working with DICOM data (Fig. 3). However, it was designed specifically with an emphasis on contributing DICOM images to a research data archive. It includes editing functionality that allows users to remove header content that might compromise the privacy of patients and research participants. Users can create and run deidentification scripts that alter header fields systematically across multiple files. DicomBrowser can also send images to DICOM receivers, allowing users to move studies that are not present at the scanner – for example, studies exported from a PACS – to a research archive.

Whereas DICOM is the dominant data format and transport standard used by scanners, it is insufficient for supporting all of the neuroimaging research enterprise requirements: legacy data and scanners often predate DICOM, DICOM security standards may be too lax (eg, lack of per-user access control), and image processing routines typically generate non-DICOM file formats (eg, Analyze, NIfTI, and NRRD). We have therefore implemented non-DICOM transport mechanisms for capturing data. We have developed custom FTP- and HTTP-based receivers to capture and organize images into a prearchive and to generate XML in the same manner as DicomServer. We have also created a web-based user interface for uploading images. These tools illustrate the value of implementing the enterprise system as a modular workflow of open-source components. The multiple different data capture applications each fill different needs but all comply with the same prearchive architecture, allowing straightforward interoperation with data archiving tools.

Data Archive and Mark-up

We have developed a software platform specifically designed for archiving neuroimaging data. The Extensible Neuroimaging Archive Toolkit[5] (XNAT; http://www.xnat.org) is a Java-based open-source toolkit that includes a file store, a relational database, a web-based user interface, and various services for accessing the data programmatically (Fig. 4). XNAT supports a number of the steps in the neuroimaging enterprise workflow, beginning with archiving and mark-up.

XNAT's user interface provides a view of the prearchive in which the data capture tools put incoming images. XNAT extracts content from the XML metadata document and presents it to the user, giving him/her an opportunity to verify that data arrived as expected and to inspect the data for compliance with study protocols. The user then enters annotations regarding the data acquisition (eg, that the subject sneezed during a scan) and qualitative assessments of the image quality (eg, that the head was positioned poorly). This mark-up serves as the first quality control step in the workflow. Further quality control procedures are implemented by XNAT throughout the workflow. The user also assigns the study to a research subject and research project in the XNAT database.

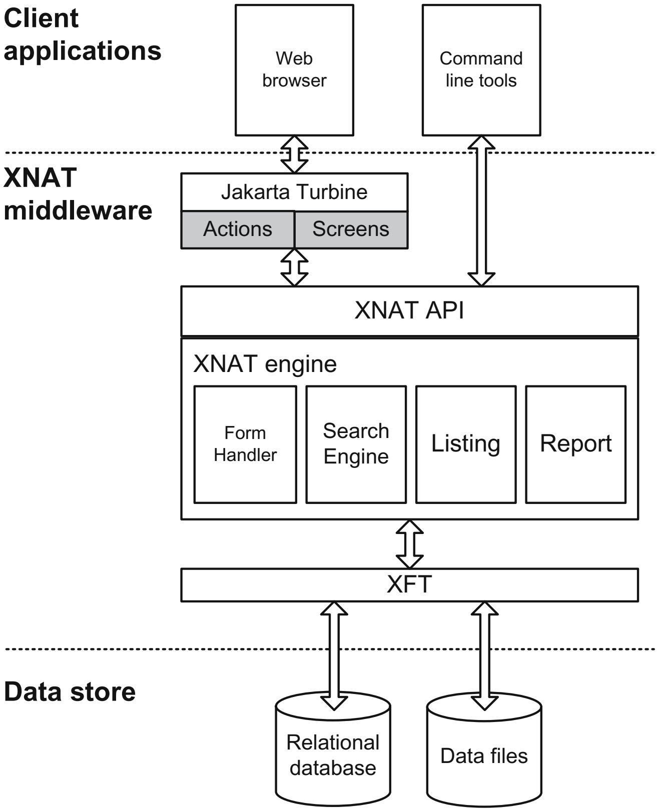


Fig 4. XNAT is a three-tiered application for securely archiving, exploring, and distributing neuroimaging and related data. Its extensible XML data model allows the database to capture study-specific data types.

When the user completes the entry, the images are transferred from the prearchive to a permanent archive, where access to the images is restricted to users directly associated with the assigned research project. The archive itself is simply a directory structure on the server's file system with links to these files written into the XNAT database. By keeping the images on the file system (as opposed to in a database or PACS), the image processing and analysis tools, which are central to virtually all neuroimaging research studies, can easily access them.

Once the images are in the archive, they are available via a number of interfaces. The web application allows users to view, download, and mark-up the images. Command line tools can be used from scripts to query the database and retrieve images. Web services provide a mechanism for application developers to interact with the archive. Throughout these operations, the integrity of the archive is maintained by XNAT's security system. Security is built around the research project. Users are assigned to projects and given roles within those projects. Only users assigned to a project can access that project's data and the user's role determines what type of access they have. A lab manager, for example, would be assigned a role allowing him to enter, edit, and process data, whereas a graduate student would be assigned a role allowing him to view but not manipulate data.

Data Processing and Analysis

Whereas images in the clinical setting typically undergo little processing, research images undergo many – often dozens – of processing and analysis routines. Typical processing steps include reducing scanner artifacts, head motion correction, transformation of data into standard atlas space, compensation of systematic, slice-dependent time shifts, and elimination of systematic odd–even slice intensity differences due to interleaved acquisition. Analysis may include brain region segmentation, atrophy measures, and white matter tract labeling. Each laboratory likely has its own set of routines that it uses, which may include standard packages and local in-house applications. Directing and monitoring the execution of each routine in an organized sequence is critical for producing usable postprocessed images and derived measures. We have built an open-source pipeline tool, *PipelineRunner*, to facilitate this oversight and ensure that all of a project's image data undergo identical processing and analysis.

PipelineRunner executes pipelines defined in project-specific XML specification documents that describe the sequence of tasks that constitute a pipeline. The specification documents include detailed descriptions of the tasks, their associated executables, and input and output data. Typical tasks include retrieving images from an archive, running processing routines, generating quality control snapshot images, updating databases with derived measures, and delivering e-mail notifications. The engine is responsible for monitoring the progress and checking the exit status of each step in the pipeline. The engine sends update notices to listeners, such as the XNAT web application, and sends e-mail notifications to users when a pipeline ends. The engine is capable of pausing and entering pipelines at any step, which is extremely useful if a manual procedure is required or if a task needs to be rerun with different parameters.

In addition to organizing the execution of processing and analysis routines, the pipeline approach provides a key integration mechanism between these routines and the rest of the enterprise workflow. A service within the XNAT web interface allows users to select images from the archive and launch pipelines on them via PipelineRunner. As PipelineRunner executes each step, it writes status messages back to the XNAT system to provide users with regular feedback. As derived images are generated, the images are stored in the archive and links are written to the database with references to the original data. The end result is that acquired data and data derived from them appear as a unified data set to the user.

The pipeline approach also facilitates the generation of quality control measures. PipelineRunner generates a provenance record for each step in a pipeline, detailing exactly what process and version was executed, what parameters were supplied to the executable, what machine the process was executed on, and when the process was run. From a provenance record, a user can exactly reconstruct a derived image's history (and potentially regenerate it). Specific quality control tasks can also be built directly into the pipeline to create snapshot images, generate summary statistics, or pull lines from a log file. These quality control measures can be written to the archive database and made available to users via the web application.

Data Integration

Thus far, the workflow has been composed entirely of images and measures derived from them. Neuroimaging studies typically include a range of nonimage data, including subject demographics, behavioral measures, and clinical assessments. By incorporating these measures directly into the archive database, the full research enterprise is available through a unified interface. This integration allows researchers to generate complex queries across data types and to mine for unexpected patterns in the data.

The XNAT database was designed to allow researchers to easily incorporate an extensible set of nonimage data. This extensibility is enabled by XNAT's core data model implementation in XML Schema.[6] From this core schema, XNAT generates a relational database in PostgreSQL.[7] By adding extensions to the core schema, additional data types that are part of a particular study can be incorporated into the data model. XNAT adds these extensions to the relational database and automatically builds in relations to the core content. These relations provide the key integration mechanism between image data and project-specific nonimage measures. XNAT also automatically generates a substantial amount of software infrastructure to support these custom extensions, including web pages, custom search interfaces, and data access objects.

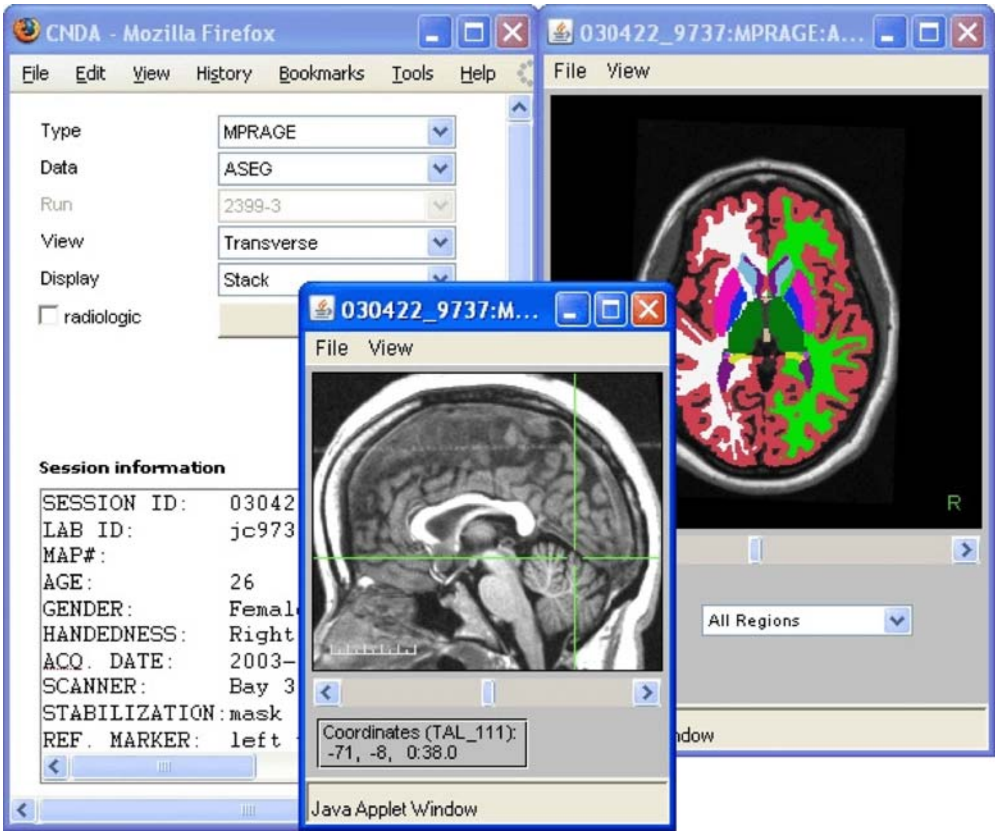
XNAT's search interface is designed to facilitate queries on the integrated database. It allows users to enter search criteria tailored to each data type and to request a result set that joins across data types. For example, an investigator searching an archive that has been extended to capture data in an Alzheimer's disease study could enter a search for female subjects with moderate dementia, no history of stroke, and mild brain atrophy. He could further request that the result set include all clinical and demographic measures and regional brain volume measures. The XNAT search engine would build an SQL query to resolve these criteria and uses the built-in relations to join across the requested data types. The search results would be presented to the user with the option of downloading the associated image data.

Discovery and Exploration

As the above search suggests, the integrated database provides a valuable resource for exploring research study data. Indeed, everything leading up to this stage of the workflow has simply been elaborate preparation for discovery. Discovery tools include data mining applications, image viewers and manipulators, plotting packages, and statistics applications. Here, users tend to rely more on commercial products (eg, Excel, SAS, Matlab) despite the availability of open-source alternatives. Perhaps this is due to the ubiquity and relative low cost of these products (compared to commercial enterprise software). Nonetheless, most neuroimaging-specific discovery tools are open-source or freeware. XNAT itself includes an open-source web-based image viewer that can be extended to support custom image types (Fig. 5). Packages like FreeSurfer,[8] FSL,[9] Caret,[10] and 3D Slicer[11] enable more sophisticated image manipulation and display.

From a workflow perspective, the key to enabling discovery is to simplify the exchange of information and data between the archive and discovery tools. Because tool builders and archive developers are typically independent of one another, it is critical to develop standard interfaces for data exchange. Whereas DICOM has been widely adopted as a standard by commercial vendors and is a vast improvement over the opaque formats previously used by scanner manufacturers, it is still a closed, complicated, top-down standard. Research requirements – the need for 32-bit floating-point representations in functional magnetic resonance imaging, for example – are quite difficult to implement in DICOM. For these reasons, the neuroimaging research community has tended to develop its own formats, often tailored to best suit specific tools. An unfortunate consequence of this tendency is that there is now a proliferation of formats. To combat this, XNAT uses an XML data model (using XML Schema) and web services as a layer of abstraction between the image data and client applications. We are currently working with the Biomedical Informatics Research Network (BIRN)[12] to merge this model with similar efforts to form a widely adopted standard for neuroimaging research data and transport.[13]

The initial outcome of these efforts is promising. As prototype implementations, FreeSurfer and 3D Slicer are being developed to use standard web service interfaces to retrieve images from XNAT archives, generate derived images and measures,



Type MPRAGE
Data ASEG
Run 2399-3
View Transverse
Display Stack
□ radiologic

Session information

```
SESSION ID:    03042
LAB ID:        jc973
MAP#:
AGE:           26
GENDER:        Femal
HANDEDNESS:    Right
ACQ. DATE:     2003-
SCANNER:       Bay 3
STABILIZATION: mask
REF. MARKER:   left
```

Coordinates (TAL_111):
-71, -8, 0:38.0

Fig 5. The image viewer built into the XNAT web application uses Java applet technology to distribute images over the web. Its plug-in design enables developers to create new display modes for neuroimaging images. Here, a sagittal view of a T1 image is shown next to a transverse view of a FreeSurfer segmentation of the T1 image.

and write the generated content back to the archive. The vision of the BIRN and other standards bodies is that a unified neuroinformatics network of databases and discovery tools will evolve to support the emerging high-throughput, collaborative neuroimaging enterprise.

DISCUSSION

In the clinical imaging environment, the scanner and PACS serve as the axis upon which the entire enterprise is based. Images are acquired on the scanner and sent to the PACS. The PACS stores and distributes the data to dedicated work stations and printers for viewing and production of hard copy. Radiologists look at the images and generate qualitative reports. In contrast, the research enterprise relies on complex postprocessing pipelines, generation of quantitative measures, and integration with a range of related measures. In addition, neuroimaging research is becoming increasingly collaborative, with multiple sites acquiring data for a single study and single sites distributing data to collaborators at multiple locations. These differences lead to substantial differences in data capture, archive, and user interface requirements. We presented in this paper a workflow model of the neuroimaging research enterprise, with the intention of establishing these requirements and open-source approaches to meeting them.

The workflow described in this paper provides a model for data in the enterprise. A user workflow can also be modeled and mapped on top of the data workflow. The principle feature of the user workflow is that the user community expands as data move from acquisition and quality control through collaboration and analysis to publication and data sharing. The user workflow can be used to establish requirements for the security, accessibility, and interface requirements of the enterprise. As neuroimaging research becomes more collaborative and data sharing becomes requisite,[14] further study of the user workflow and its interrelation with the data workflow will likely prove valuable.

The XML data model and accompanying web services used by XNAT and other research imaging systems contrast with the DICOM-based approaches used by clinical systems. On the one hand, XML and web services reflect the broader movement in software engineering for representing and communicating data. A larger number of open-source tools are therefore available to support them, and best practices for using these tools to support multisite imaging studies are beginning to emerge. Perhaps most importantly, they can be used in conjunction with the multiple imaging formats that are commonly used by the research community. On the other hand, within the medical imaging community, DICOM is the more widely accepted standard. To facilitate the transition of research methods into the clinic, it will be important to bridge the divergent approaches that now divide research and clinical systems.

Our coverage of the data workflow leaned heavily on open-source tools developed in our group. However, the workflow approach is intended to identify discrete components of the research enterprise that can be supported through independent application development. Our intention in describing our own tools in some detail was to elaborate on open-source approaches to developing discrete modules and communication between modules. Following this approach, the various components could be replaced or mixed and matched relatively easily with alternative solutions that may be more appropriate in other environments. For example, the DICOM tools could be replaced with an institution's existing PACS tools. The file system-based image archive could be replaced with a virtual file system like the Storage Resource Broker.[15] The PipelineRunner could be replaced with a GUI-based tool like LONI Pipeline.[16]

CONCLUSION

As discoveries move from the laboratory to the clinic, one can imagine that the future of radiology may look a lot more like the research enterprise, with quantitative approaches becoming more commonplace and nonimaging measures being integrated into the diagnosis process. Given the depth of open-source software available in the research domain, one could also imagine that industrial-strength versions of these tools may serve as counterparts in the clinical domain. It will likely be an area of active open-source development to bring laboratory and clinic approaches into closer alignment.

REFERENCES

1. NITRC: NITRC website. http://www.nitrc.org. Accessed July 17, 2007.

2. Neuroscience Database Gateway: Neuroscience Database Gateway. http://ndg.sfn.org. Accessed July 17, 2007.

3. DICOM: DICOM web site. http://medical.nema.org. Accessed June 8, 2007.

4. dcm4che: dcm4che website. http://www.dcm4che.org/

5. Marcus DS, Olsen TR, Ramaratnam M, Buckner RL: The extensible neuroimaging archive toolkit (XNAT): An informatics platform for managing, exploring, and sharing neuroimaging data. Neuroinformatics 5:11–34, 2007.

6. W3C: W3C website. http://www.w3.org/XML/Schema. Accessed June 8, 2007.

7. PostgreSQL: PostgreSQL website. http://www.postgresql.org/. Accessed June 8, 2007.

8. FreeSurfer: FreeSurfer website. http://surfer.nmr.mgh.harvard.edu/. Accessed June 8, 2007.

9. FSL: FSL website. http://www.fmrib.ox.ac.uk/fsl/. Accessed June 8, 2007.

10. Caret: Caret website. http://brainmap.wustl.edu/caret/. Accessed June 8, 2007.

11. 3D Slicer: 3D Slicer website. http://www.slicer.org/. Accessed June 8, 2007.

12. Biomedical Informatics Research Network: Biomedical Informatics Research Network website. http://www.nbirn.net/. Accessed June 8, 2007.

13. Keator DB, Gadde S, Grethe JS, Taylor DV, Potkin SG: FIRST BIRN: A general XML schema and SPM toolbox for storage of neuro-imaging results and anatomical labels. Neuroinformatics 2:199–212, 2006.

14. NIH: NIH Data Sharing Policy webpage. http://grants.nih.gov/grants/policy/data_sharing/. Accessed June 8, 2007.

15. Storage Resource Broker: Storage Resource Broker web site. http://www.sdsc.edu/srb/index.php/Main_Page. Accessed June 8, 2007.

16. LONI Pipeline: LONI Pipeline web site. http://www.loni.ucla.edu/Software/Software_Detail.jsp?software_id=2 Accessed June 8, 2007.