Neuroinform (2010) 8:231–249
DOI 10.1007/s12021-010-9078-6

Federated Web-accessible Clinical Data Management within an Extensible NeuroImaging Database

I. Burak Ozyurt · David B. Keator · Dingying Wei · Christine Fennema-Notestine · Karen R. Pease · Jeremy Bockholt · Jeffrey S. Grethe

Published online: 22 June 2010

Abstract Managing vast datasets collected throughout multiple clinical imaging communities has become critical with the ever increasing and diverse nature of datasets. Development of data management infrastructure is further complicated by technical and experimental advances that drive modifications to existing protocols and acquisition of new types of research data to be incorporated into existing data management systems. In this paper, an extensible data management system for clinical neuroimaging studies is introduced: The Human Clinical Imaging Database (HID) and Toolkit. The database schema is constructed to support the storage of new data types without changes to the underlying schema. The complex infrastructure allows management of experiment data, such as image protocol and behavioral task parameters, as well as subject-specific data, including demographics, clinical assessments, and behavioral task performance metrics. Of significant interest, embedded clinical data entry and management tools enhance both consistency of data reporting and automatic entry of data into the database. The Clinical Assessment Layout Manager (CALM) allows users to create on-line data entry forms for use within and across sites, through which data is pulled into the underlying database via the generic clinical assessment management engine (GAME). Importantly, the system is designed to operate in a distributed environment, serving both human users and client applications in a service-oriented manner. Querying capabilities use a built-in multi-database parallel query builder/result combiner, allowing web-accessible queries within and across multiple federated databases. The system along with its documentation is open-source and available from the Neuroimaging Informatics Tools and Resource Clearinghouse (NITRC) site.

Keywords Data sharing · Federated databases · Neuroinformatics · Neuroimaging data management · Open source

Introduction

Researchers within any given field, and imaging researchers in particular, collect, store, and access their data in physically different and often incompatible ways. Multi-site and longitudinal studies only compound these differences as idiosyncratic methods inevitably accumulate and change over time. Simply aggregating data from each center in one place does not solve the problem, as the investigator-specific understanding of the data coding and storage methods needs

I. B. Ozyurt (✉) · C. Fennema-Notestine
Department of Psychiatry, University of California at San Diego,
San Diego, CA, USA
e-mail: iozyurt@ucsd.edu

D. B. Keator · D. Wei
Brain Imaging Center, University of California at Irvine,
Irvine, CA, USA

C. Fennema-Notestine
Department of Radiology, University of California at San Diego,
San Diego, CA, USA

K. R. Pease
University of Iowa,
Iowa City, IA, USA

J. Bockholt
MIND Institute,
Albuquerque, NM, USA

J. S. Grethe
University of California at San Diego,
San Diego, CA, USA

to be made transparent to other investigators for multi-site data integration. Doing so requires a sophisticated data management system: a comprehensive structure for data storage; interoperability between incompatible, sometimes proprietary data formats; and ontologies to translate concepts from one investigator's system to another, to name a few challenges. The ability to reliably and quickly transfer enormous (multi-terabyte) datasets across distributed neuroimaging sites is also required.

The Function Biomedical Informatics Research Network (FBIRN) focuses on the development and validation of functional magnetic resonance imaging (fMRI) techniques in large-scale, multi-site clinical neuroimaging studies of schizophrenia (Keator et al. 2008). In collaboration with the BIRN Coordinating Center (BIRN-CC) and the Morphometry BIRN (MBIRN) program, the FBIRN has developed a distributed network infrastructure to support the creation of a federated database consisting of large sample structural and functional MRI datasets (sMRI, fMRI). These data have been contributed by and at the same time controlled by the consortium members in an effort to assess the major sources of variation in sMRI and fMRI studies conducted across sites, including instrumentation, acquisition protocols, challenge tasks, analysis methods, and anatomical variations. The projects develop models that permit the analysis of combined data by considering this variability (Friedman and Glover 2006; Friedman et al. 2008; Keator et al. 2008; Magnotta and Friedman 2006). The FBIRN's second major goal is clinical: to use the technology developed herein to study changes in brain function in the development, progression, and treatment of neuropsychiatric illness (Brown et al. 2009; Ford et al. 2009; Kim et al. 2009a, b; Potkin and Ford 2009; Potkin et al. 2009a, b; Wible et al. 2009).

With the emergence of the neuroinformatics field, there have been several systems proposed to manage neuroimaging and other scientific data, although none of these systems provide for multi-site data federation and collaboration, data storage for clinical and imaging values and files, and data entry management in a single package. SenseLab,[1] developed at Yale University, is a metadata driven system to store scientific data using an entity-attribute-value with classes and relationships representation in a relational database and corresponding web interface (Nadkarni et al. 1999; Marenco et al. 2003). The fMRI Data Center's Data Management Tool[2] uses ontology-based knowledge engineering to provide a fMRI ontology to map experimental data and metadata. Ohio State University's GridPACS[3] system provides distributed data storage and Grid based processing pipelines (Hastings et al. 2005), useful for image storage and processing. The Extensible Neuroimaging Archive Kit (XNAT[4]) developed by the Neuroinformatics Research Group at Washington University at St. Louis relies on an extensible XML schema to represent imaging and experimental data and supports a Postgres database backend (Marcus et al. 2007). XNAT is most similar to the proposed system in the extensibility goals and application scope but lacks federation aspect having centralized repository focus and requires modification of the underlying schema and views for new clinical assessments. In XNAT, the database schema is generated from the extensible XML schema.

The Human Clinical and Imaging Database (HID) and toolkit system introduced here differs from these systems in its aim to address the needs of multi-site collaborative projects, combining both imaging and clinical data hosted in a federated and distributed environment. The HID system provides a framework for optimizing the tradeoffs between high performance, usability, robustness and extensibility. This system relies on an extensible relational database schema based on abstract data types allowing extension by introducing metadata rather than schema changes, which provides for more efficient expansion, and utilizes metadata to guide/construct its user interface dynamically. The introduced system is comprised of two core components:

- The Human Clinical and Imaging Database (HID) schema.
- A data management toolkit (HID toolkit) comprised of an extensible web application/framework (HID application) and the Clinical Assessment Layout Manager (CALM).

This paper is focused on a description of the major HID toolkit components, along with some components of the HID schema at the data model level. The schema descriptions emphasize the four essential parts of the HID schema, namely, Experiment-Subject-Visit Management, Assessment Management, Extended Tuple model and Analysis Workflow/Provenance. This paper is intended foremost as a guide for new end users and secondly as an introduction for developers of large-scale scientific data/image management systems.

Background

Among the first considerations in the design of any neuroscientific data management system is the complexity and richness of the data that needs to be managed and organized. Neuroscientists collect data that is comprised of

[1] http://senselab.med.yale.edu/senselab/.

[2] http://www.fmrifc.org/f/fmridc/dmt/index.html.

[3] http://bmi.osu.edu/areas_and_projects/mobius.cfm.

[4] www.xnat.org.

many types, from the storage of simple tabular data (e.g. derived volumetric measurements for brain regions) in both numeric and textual form to more complex data types such as images, volumes, and time-series data. However, a more daunting problem is how to represent the immense variety of experimental preparations and the data they generate. For example, a researcher might want to manage data collected in his laboratory from human assessments, sMRI studies, as well as neuroimaging studies involving the use of FMRI. In order to be able to comprehensively store these datasets we must be able to "wrap" the experimental data with the information regarding the protocol that was used in collecting the data (e.g., define the FMRI scan acquisition protocol along with the behavioral task description of the FMRI paradigm). One of the problems in specifying and storing the protocols and data that need to be managed is their heterogeneity. Therefore, in designing the data representation for a scientific experiment a very important principle must be taken into account: One cannot hope to describe a priori all the protocols and research data that a researcher will want to incorporate in their own data management systems. Due to this need for continued evolution, the HID toolkit was designed so that it is easily modified to meet a specific researcher's needs without needing major modifications in the system's overall structure.

Data maintenance involves data creation, update and deletion by multiple (possibly) concurrent users where the assurance of the data integrity can be challenging. In a neuroscientific setting, data maintenance, at the core, involves managing experiments, subjects and their visit information. A data management system should allow creation of new experiments and enrollment of subjects to them. Both scanning and clinical visits for the participants, which are often separated in time, need to be captured. Longitudinal experiments where subjects are tracked over a period of time, often with different lag time between visits, also need to be supported. Further complications include the potential enrollment of a participant into multiple experiments, both across time and within a single visit. Each study may represent different modalities, e.g. Positron Emission Tomography (PET) or Electroencephalography (EEG), and during a single visit, a participant may perform cognition tasks followed by one or more scan sessions where they perform structural MRI and FMRI paradigms, possibly followed by another set of clinical assessment sessions. Each of these activities must be recorded in the data management system and differentiated.

Clinical Assessments Management

Maintenance of clinical assessments constitutes an important part of clinical neuroimaging data management. Depending on the type of the experiment conducted, neuroscientists use different sets of questionnaires, for example, to assess the condition of the subjects involved in the experiment. These assessments complement the imaging data for the subjects, provide insight into diagnoses and symptomotology of participants, which may in turn be related to brain structure or function. Hence, an extensible mechanism is necessary to capture and view these evolving sets of clinical assessments, in addition to storing the results. The HID Toolkit system facilitates the transition from paper forms to on-line clinical assessments, and enables the on-line assessments to look as similar as possible to their paper form counterparts for ease of data entry. Whenever a new on-line assessment form needs to be added to the HID system, it can be done quickly, easily, and without programmer involvement. To minimize data entry errors, double entry and reconciliation capabilities also are available in the HID Toolkit.

Image Analysis and Derived Data Maintenance

One of the main goals of multi-site neuroscience studies is enabling large-scale data collection of hundreds, if not thousands, of subjects. By pooling image/clinical data collected, sites benefit from larger population sizes and increased statistical power. Imaging data, particularly FMRI data, can be many gigabytes in size per subject visit. For example, in FBIRN, the raw data (data coming from scanner with standard preprocessing and quality checks) is about two gigabytes per subject visit. Data derived by analyzing the raw images can be multiple times larger than the raw data. Moving this much data around is challenging within a local area network environment and more so in a geographically-diverse distributed environment where even the physical limits of networks needs to be taken into account in design of the distributed query/data transfer interfaces. In neuroimaging studies, the raw data may be analyzed by tools such as FreeSurfer,[5] 3D Slicer,[6] FMRIB Software Library (FSL[7]) or the FBIRN Image Processing Stream (FIPS[8]). These tools require all of the data for analysis to be available in a local/mounted file system, with each tool preferring specific directory structures. Hence a federated data management system needs to be able to combine/bundle image data for analysis tools from multiple sites filtered by user queries.

Generic, structured storage of derived data and metadata describing processing pipelines is an important component of a research data management system (Keator et al. 2009).

[5] http://surfer.nmr.mgh.harvard.edu/.

[6] http://www.slicer.org/.

[7] http://www.fmrib.ox.ac.uk/fsl/index.html.

[8] http://www.nbirn.net/research/function/fips.shtm.

In federated systems, data is often being generated at a faster rate than in individual laboratories and analyses being performed at numerous sites in the collaboratory project. It is therefore important to provide researchers with the capability to contribute resultant analyses to the larger federation in a well-documented and structured way facilitating interpretation and reuse.

Methods

In the light of the requirements described in the previous section, the data model and architecture of the federated neuroimaging data management system introduced in this paper are described in the following sections.

Data Model

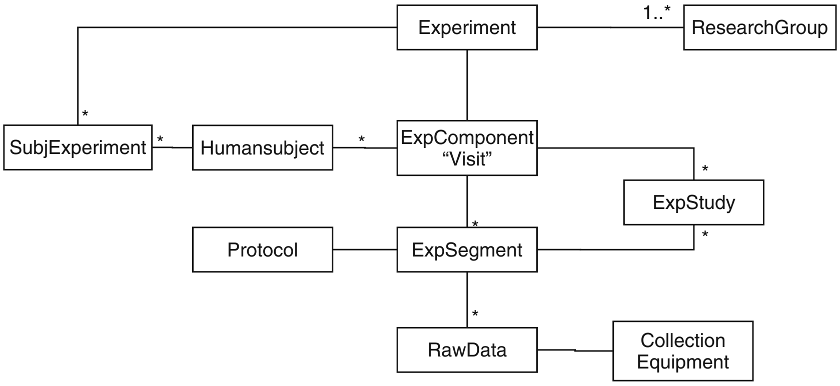
A careful consideration of neuroscientific experiments shows that certain concepts like experiment/project, visit, session, assessment, acquisition protocol reoccur in many research settings. By making a clear distinction between a concept and its multiple realizations and by separating them, an extensible data model can be achieved. This approach allows the data model to remain unchanged while enabling extensibility. For example, there are many clinical assessments available and new ones are added constantly. It is impossible to account for all of the assessments required for all possible experiments that can be conducted. However, by abstracting the assessment concept from its many realizations (different assessments), one can model the assessment concept (metadata structure) statically. Different realizations of the assessment concept, then, can be provided dynamically without requiring a change to the data model. The instantiations of the realizations of the assessment concept (assessment done on a particular subject) are modeled separately from the assessment concept.

A neuroscience experiment is modeled as shown in Fig. 1. A group of subjects (Humansubject) can participate in zero or more experiments. The participating subjects (SubjExperiment) are partitioned into groups such as patient or control (ResearchGroup). Each participating subject can have zero or more visits (ExpComponent). Each visit is further divided into segments (ExpSegment). For an FMRI scan session, the segments correspond to different runs of each FMRI paradigm, for example. The segments can be partitioned into studies (ExpStudy) that group segments, for example, into different modalities (FMRI, PET). Each segment is associated with a protocol. A protocol is a collection of parameters and conditions used to conduct the scan. If a segment belongs to a scanning visit, there are associated image data (RawData). Each RawData entry is further associated with particular collection equipment where the images were acquired.

Clinical data collected during a neuroscience experiment is modeled as shown in Fig. 2. Separation between the concept data model and instantiated data model is further highlighted in this figure. An assessment is a collection of scores (AssessmentScore). A score represents an atomic answer for an assessment question. Scores can form a hierarchy, i.e. a score can have subscores. For many assessment questions, a participant has to choose from a limited set of answers. AssessmentScoreCode objects represent this information. Each score is also associated with a question (AssessmentItem). The assessments done on a participating subject are attached at the segment level and modeled by a StoredAssessment object per assessment instance. Each StoredAssessment object also associated with the conveyor of assessment information (AssessmentInformant) and its status (AssessmentStatus). The particular values for assessment scores for the conducted assessment are partitioned by their data types to facilitate efficient retrieval. If a score value is missing or unanswered for some reason, this information is captured in AssessmentData pointing to the reason (DataClassification).

To attach arbitrary information to a concept realization defined in HID, dynamically, the extended tuple data model in Fig. 3 is introduced. Each concept can be considered as a tuple. Each element of this tuple represents an attribute of the concept. To further enhance the attribute set of a

Fig. 1 Experiment-subject-visit management data model



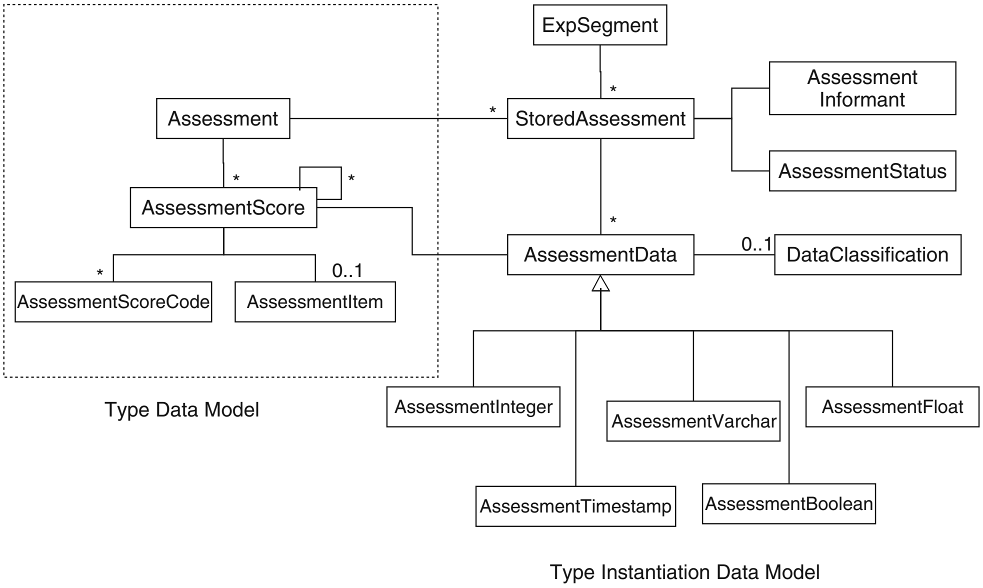


Fig. 2 Assessment management data model

concept dynamically, new attributes can be attached to the concept using the introduced extended tuple approach. The pattern of the extended tuple data model bears a resemblance to the assessment management data model. Here, also, the extended tuple concept model is separated from extended tuple concept realization instance model for extensibility. The TupleClass points to the concept that is extended. The TupleSubClass is used for further refinement of the concept. For example, to indicate the extension of ResearchData concept with analysis result information, TupleClass will point to ResearchData and TupleSubClass to the AnalysisResult concept. The ExtendedTuple concept groups a set of new attributes to be attached to a statically modeled concept. Each extended tuple can have multiple attributes (TupleColumns) including unit information (MeasurementUnit).

An example of extended tuple usage in HID is a representation of data derived from various analyses of neuroimaging data. These types of data depend on the analysis conducted and particular objectives of the research. For example, Freesurfer analysis for brain region volume estimations will have different types of results than the measures of gyral curvature. It is also very rich and ever changing, making it impossible to model statically. The abstraction level of the extended tuple model allows storage of the analysis derived data without changing the data model for each new type of analysis and/or derived information.

Any operation performed on the reconstructed MRI images is considered an analysis or derived dataset. The data model of an analysis process, derived data generated by particular analyses conducted, and provenance information is shown in Fig. 4. The pattern of type (metadata) and type

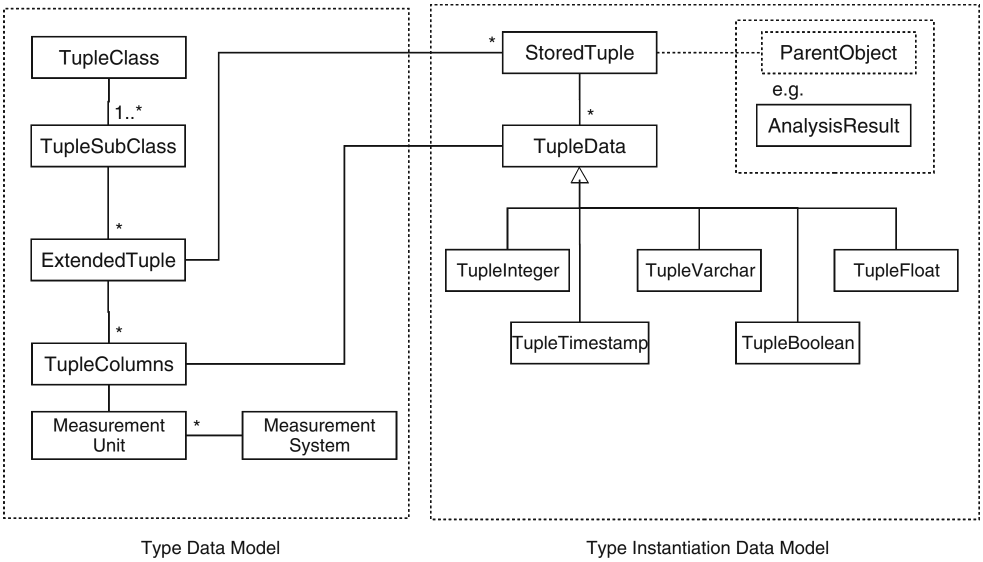


Fig. 3 Extended tuple data model

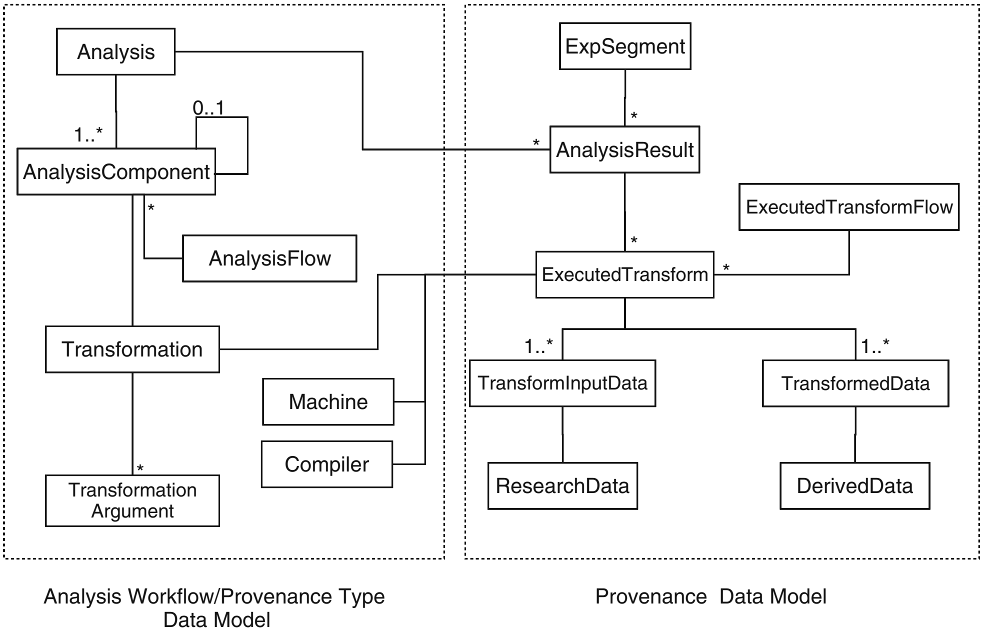


Fig. 4 Analysis workflow/provenance data model

instance separation is also shown in this model. An analysis is composed of one or more AnalysisComponents. An AnalysisComponent is a step in the derived data generation process representing a well-defined transformation (Transformation). A hierarchy of AnalysisComponents can be also defined building higher-level transformations utilizing multiple low level transformations. Each Transformation can be parameterized by TransformationArguments. The AnalysisComponents are linked into workflows/pipelines by using the AnalysisFlow concept. Various analysis types can be introduced dynamically by creating instances of Analysis concept.

An actual instance for a particular analysis on a subject is modeled by the AnalysisResult concept and linked to its subject data at the segment level. Particular transformations performed on an analysis instance are documented by the ExecutedTransform concept. Each ExecutedTransform is associated with a Transformation, which provides its type information. The executed workflow/pipeline, which can be somewhat different based on the condition of the data from the prescribed flow by AnalysisFlow concept instances, is modeled by the ExecutedTransformFlow concept. Together with Machine and Compiler concepts, ExecutedTransform and ExecutedTransformFlow model the provenance data captured about the conducted analysis. The inputs for the analysis are modeled by TransformInputData concept. The references to the essential artifacts generated by the analysis instance are modeled by the TransformedData concept to be used for retrieval purposes. Results of interest from an analysis for query purposes are captured using the Extended Tuple Data model described above.

System Architecture

The HID application (the extensible web application/framework) is accessible to both human users and to client programs wanting to use HID data in a coarse-grained service-oriented manner via web services. The HID application is a three tier J2EE (Java 2 Platform Enterprise Edition) application. The overall architecture is shown in Fig. 5. It has a (thin) client tier, a middle tier (servlet/Java Server Pages (JSP) based) using Apache Struts[9] web framework for human clients, a web service engine for serving client applications, and an information source (IS) tier. The client tier consists of a web browser. The IS tier is the Oracle (or PostgreSQL) database instance(s) and the collection of stored procedures/functions and packages for low-level bulk data upload. The middle tier is the heart of the system forming the HID application. The framework has a layered architecture. The lowest layer, Data Access Object Layer, decouples relational data sources from the framework, providing database agnostic access. This layer consists of code-generated data access objects (DAO) (Alur et al. 2001) for all supported relational database management system (RDBMS) types (Oracle and Postgres, currently). The DAOs can be seen as a simple object-relational mapping, since they map database tables to objects. They provide a data access layer, which is not directly accessed by the presentation layer to reduce inter-layer dependency. The DAOs provide CRUD (create, read, update and delete) operations on database table level and are fine grained. A code generator creates DAO layer

[9] http://struts.apache.org/.

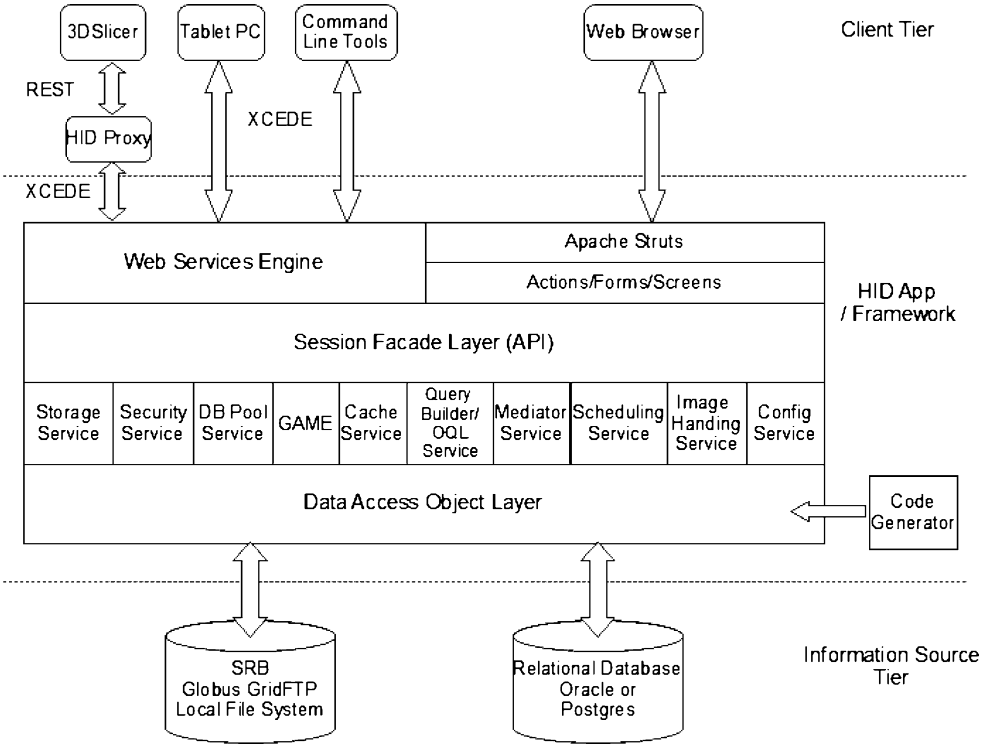


Fig. 5 High-level architecture of HID application/framework

automatically from the HID schema. This tool facilitates database schema update management, provides optimal performance (no runtime overhead) and strong type checking at compile time. The presentation layers, Struts based web framework for human clients and web services engine for client applications, communicate with the data access layer through session facades (Alur et al. 2001), which coordinate operations between multiple DAOs in a workflow and provide a coarse grained simpler interface hiding the internals of the business logic from the presentation layer.

A service layer stands between DAO layer and Session Façade Layer providing crosscutting core services. Some of these services including Object Query Language (OQL) Service and Storage Service, directly communicate with the information resources. The Storage Service only deals with image files thus makes no use of DAO layer. The services provided at this layer are described in the following paragraphs.

The Query Builder/OQL Service provides a lightweight database independent query language represented in value object terms bypassing the DAO layer for optimum performance. This service is also responsible for generic query building support for clinical assessments and derived data. The analysis derived data comes in various forms, which include segmentation/parcellation results (volume/curvature values for cortical/subcortical regions) for MRI images, quality control measures for FMRI scans and statistical map information for multilevel FMRI analyses. Since exposing the underlying low-level query mechanism is not appropriate for end users, a middle ground needed to be established between query builder generality and ease of use. The query interface for the user needs to be simple, intuitive and powerful enough to satisfy most needs of the end user while still being generic enough to facilitate queries of unknown data a priori.

The Generic Assessment Management Engine (GAME) service provides a generic mechanism to collect data via the CALM-generated on-line assessment forms. CALM and GAME complement each other. The former is used offline to layout and generate assessment forms, while the latter presents the online form to the user and manages user data entry. GAME manages the lifecycle of online assessments in an assessment independent way. It persists the entries into the database and facilitates double entry and reconciliation functionality to increase data entry reliability. It also coordinates forward and backward traversal through on-line assessment pages, so that complete data entry of an assessment does not need to occur in one sitting, a particularly useful feature for entering long clinical assessments.

The Security Service provides authentication and authorization services for higher layers. The Database Pool Service provides a non-traditional two-stage connection pool for multi-site federated queries enabling database agnostic query building.

To minimize the effects of network latencies and data transfer bottlenecks, particularly problematic when receiving large image downloads, a distributed object cache service has been developed which caches the most frequently/recently used data. It supports both image data, which are stored outside the HID and remote database queries. Since remote process communication of any sort is usually at least an order of magnitude slower than the corresponding local operation, caching provides significant performance benefits at the cost of increased storage and memory on the server side. To address potential space shortage problems, a storage space aware cache eviction mechanism is utilized, which recovers storage space by removing the least used cached items first and ensuring enough storage space is available for proper operation. Additionally, in a federated setting, server caches can become out of sync. As such a distributed event based synchronization mechanism is provided. The synchronization mechanism is only available for federated database query support, since it needs cooperation between participating systems to publish state change events to their distributed counterparts.

The Mediator Service provides multi-site query and result combining support. For an end user, there should be no perceptible difference between a query to the local database and a query spanning multiple databases. The mediator service provides this functionality by adding multi-instance query capabilities to the Query Builder service. The Database Pool Service provides uniform access to any database in the federation. By use of the distributed cache service, the number of expensive auxiliary remote queries is minimized.

While most of the features provided by the HID application run interactively, near real time, some tasks involve long computational times making them impractical to be handled in a synchronous fashion. A Job Scheduling Service schedules these long running tasks, including image data preparation and batch report style queries.

The Image Handling Service provides image file conversion/manipulation services for image series previewing. The Config Service provides system configuration/user management data services along with an interface for specifying HID installations in the federation which are available for multi-site queries.

In order to be able to easily replace these services with different mechanisms, the abstract factory design pattern (Gamma et al. 1994) is used. The interfaces are the only means by which the layers communicate with each other. Even when the implementation of the interfaces changes drastically over time, if the interfaces remain the same, the layers can talk with each other. The data transfer between the presentation and the data access layer is via coarse-grained transfer objects (value objects) (Alur et al. 2001).

The underlying web application framework for the user interface is Jakarta Struts, a Model 2 architecture based on the model-view-controller (MVC) design pattern (Gamma et al. 1994). Struts intercepts a web request and determines what to display next. It also provides tag libraries for JSPs, validation, internationalization and error handling support. The web services engine provides data services for third party clients.

The imaging data can be stored in various types of file systems either local or distributed. One of these storage mechanisms is the Storage Resource Broker (SRB) system. The SRB provides transparent virtualized middleware for sharing data across distributed, heterogeneous data resources separated by different administrative and security domains (Rajasekar et al. 2002). This storage system is superseded by the Globus GridFTP[10] system in BIRN. The plug-in based mechanism of the HID framework is storage mechanism agnostic. For example, to support usage outside a distributed file system environment, the HID can be configured to use a locally accessible file system and can be changed from experiment to experiment hosted by the system from configuration panel of the HID application.

Clinical Assessment Layout Manager (CALM)

The CALM tool is used to prepare on-line clinical assessment entry forms, create the corresponding assessment meta-data in the underlying database, and associate the assessment scores with the form elements in the on-line form. An assessment form can have one or more pages and a cover page to collect assessment specific meta-data including the date and time of the assessment, informant information and clinical rater information. CALM provides a mechanism for specifying questions, which have multiple answers. Often in human research studies, the subject will choose not to answer some questions, and hence missing values must be properly documented and differentiated from questions that were not asked by a rater.

The main design goal for CALM is to provide a generic clinical assessment form building in a declarative manner without any programming knowledge. The end users of the tool are neuroscience researchers who want to manage their clinical assessments by extending the HID application to support new and various assessments. The tool needs to be expressive enough to visually layout forms very close to or the same as their paper counterparts, to require minimal effort, if there are no layout restrictions, to integrate the laid out assessment with the HID application automatically without any configuration and/or programming, and any existing assessment data for the generated forms should be easily imported into the HID. The declarative end-to-end assessment building/integration capability separates CALM from a generic HTML editor. It provides flexibility similar

[10] http://software.nbirn.org/.

to an HTML editor in the layout, but requires no knowledge of HID application internals and programming as would be required within an HTML editor. When there are no strict layout restrictions for an assessment, a new online assessment can be created in a relatively short time (in minutes for short assessments like NAART, in few hours for long assessment like SANS) by providing the question text, (if any) possible answers and some hints for the layout.

Security

Sharing of biomedical human subject data is highly sensitive and the security requirements are much higher than those for most other database applications. The Health Insurance Portability and Accountability Act of 1996 (HIPAA) confidentiality requirements must be met. A sure way of complying with HIPAA is not storing any personal health information (PHI) in the data sharing system at all, which is the route chosen by FBIRN. To ensure this, data import to the system, which could potentially contain PHI, is done only via separately developed FBIRN open-source upload software that anonymizes image headers and defaces anatomical images.

The HID application provides strong authentication to disallow uninvited access to the data. To enable one of the main goals of BIRN, data sharing, however, it provides a liberal authorization mechanism that allows all registered users read-only access to all available experiments. Any private data needs be held in a separate local HID app instance. Extending the authorization mechanism for more restricted environments is planned for feature releases.

Web Services Facet of the HID Application

While the HID application is intended primarily for research scientists, it also makes available some portion of its functionality to non-human clients as web services.[11] Using this functionality, the HID application can provide data to third party systems for analysis and/or visualization, accept assessment and/or derived data from other tools, and/or connect to complex workflows as a data source and/or sink. The HID application provides Simple Object Access Protocol (SOAP) web services. For clients preferring a Representational State Transfer (REST) style web services, SOAP services are adapted to REST style services by using a lightweight proxy. The layered architecture of the system facilitates the implementation of web services. The web service components need only interact with Session Facade layer of the system.

Implementation

Based on the design and architecture put forward in previous section, the implementation aspects of the devised system are described in the following subsections.

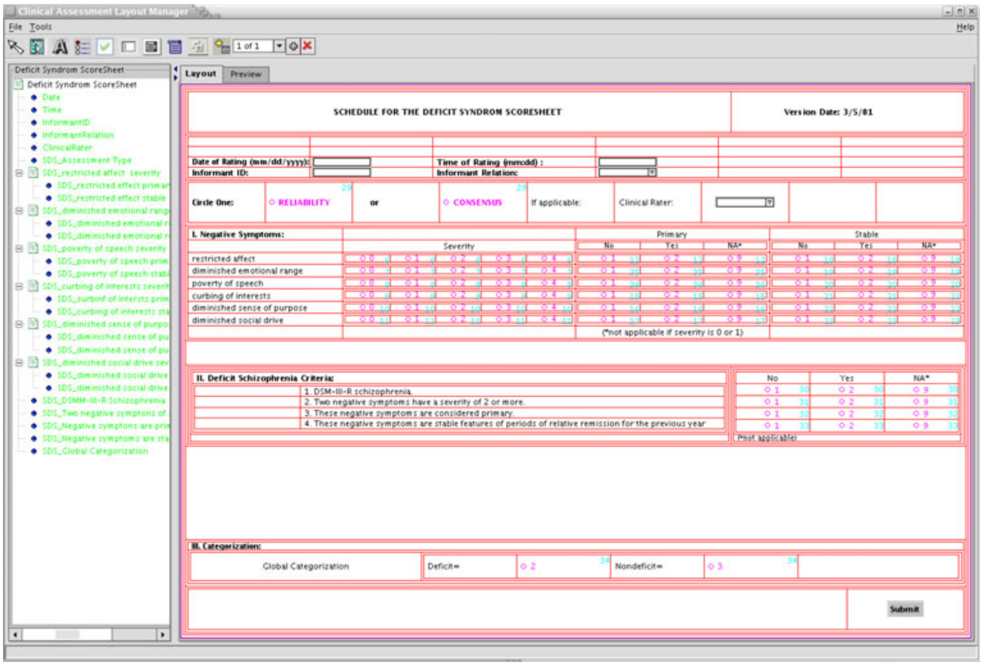
CALM and GAME

The CALM tool (See Fig. 6) provides a layout mechanism based on an arbitrary depth container and display element hierarchy. A display element is a form field, for example, a text entry field, styled text, or embedded HTML. A container is defined as a rectangular area of the on-line form that can contain zero or more display elements and/or containers. Each container has a layout manager which constraints how its children are laid out with respect to each other and their parent container. CALM currently uses a percentage based layout manager to place the children of a container, similar to the <TABLE> HTML tag.

CALM uses the JavaBeans software component model (Hamilton 1997) providing property editors and customizers for the display element and container beans used to build on-line forms. CALM currently supports seven display element types, namely, multi-line styled text, radio buttons, check boxes, text input fields, text area fields, drop-down fields (both static and dynamic SQL backed), and action buttons. To facilitate rapid data entry, user specified “Skip” action buttons are provided to support skipping of questions. Assessments with calculated fields are further supported. CALM uses a generic mathematical expression language to enter formulas for calculated fields during assessment layout and generates logic for automatically populating those fields during data entry. CALM further provides editing aids and the capability to save a portion of a form as a template to be reused in subsequent form creation, increasing productivity and turnaround time.

Since the goal of creating an on-line assessment is to enter data, the form input elements must be associated with their corresponding assessment scores. CALM provides functionality to build a new assessment from scratch or work with an already existing assessment in the HID database. An assessment consists of a hierarchy of scores that are matched to the answers of the questions in the assessment form. The scores can be bound with the form elements in the on-line form by selecting the corresponding score and form element (or form element logical group) using the mouse. A form element logical group is used to combine a set of form elements into a mutually exclusive input component (e.g. a group of radio buttons) to be associated with a single score. CALM automatically generates the handling logic for missing values in the form. CALM persists the layout and assessment association information in an XML formatted file.

[11] http://www.w3.org/2002/ws/.



Deficit Syndrom ScoreSheet

- Deficit Syndrom ScoreSheet
  - Date
  - Time
  - InformantID
  - InformantRelation
  - ClinicalRater
  - SDS_Assessment Type
  - SDS_restricted affect severity
    - SDS_restricted effect primar
    - SDS_restricted effect stable
  - SDS_diminished emotional rang
    - SDS_diminished emotional r
    - SDS_diminished emotional r
  - SDS_poverty of speech severity
    - SDS_poverty of speech prim
    - SDS_poverty of speech stabl
  - SDS_curbing of interests severit
    - SDS_surbinf of intersts prim
    - SDS_curbing of interests sta
  - SDS_diminished sense of purpo
    - SDS_diminished sense of pu
    - SDS_diminished sense of pu
  - SDS_diminished social drive sev
    - SDS_diminished social drive
    - SDS_diminished social drive
  - SDS_DSMM-III-R Schizophrenia
  - SDS_Two negative symptons of
  - SDS_Negative symptons are prim
  - SDS_Negative symptoms are sta
  - SDS_Global Categorization

SCHEDULE FOR THE DEFICIT SYNDROM SCORESHEET

Version Date: 3/5/01

Date of Rating (mm/dd/yyyy):

Informant ID:

Time of Rating (mmdd) :

Informant Relation:

Circle One: ○ RELIABILITY 29 or ○ CONSENSUS 29 If applicable: Clinical Rater:

I. Negative Symptoms:											
	Severity					Primary			Stable		
						No	Yes	NA*	No	Yes	NA*
restricted affect	○ 0 6	○ 1 6	○ 2 6	○ 3 6	○ 4 6	○ 1 12	○ 2 12	○ 9 12	○ 1 18	○ 2 18	○ 9 18
diminished emotional range	○ 0 7	○ 1 7	○ 2 7	○ 3 7	○ 4 7	○ 1 35	○ 2 35	○ 9 35	○ 1 19	○ 2 19	○ 9 19
poverty of speech	○ 0 8	○ 1 8	○ 2 8	○ 3 8	○ 4 8	○ 1 36	○ 2 36	○ 9 36	○ 1 20	○ 2 20	○ 9 20
curbing of interests	○ 0 9	○ 1 9	○ 2 9	○ 3 9	○ 4 9	○ 1 15	○ 2 15	○ 9 15	○ 1 21	○ 2 21	○ 9 21
diminished sense of purpose	○ 0 10	○ 1 10	○ 2 10	○ 3 10	○ 4 10	○ 1 16	○ 2 16	○ 9 16	○ 1 22	○ 2 22	○ 9 22
diminished social drive	○ 0 11	○ 1 11	○ 2 11	○ 3 11	○ 4 11	○ 1 17	○ 2 17	○ 9 17	○ 1 23	○ 2 23	○ 9 23
						(*not applicable if severity is 0 or 1)					

II. Deficit Schizophrenia Criteria:	No	Yes	NA*
1. DSM-III-R schizophrenia.	○ 1 30	○ 2 30	○ 9 30
2. Two negative symptoms have a severity of 2 or more.	○ 1 31	○ 2 31	○ 9 31
3. These negative symptoms are considered primary.	○ 1 32	○ 2 32	○ 9 32
4. These negative symptoms are stable features of periods of relative remission for the previous year	○ 1 33	○ 2 33	○ 9 33
	(*not applicable)		

III. Categorization:

Global Categorization	Deficit=	○ 2 34	Nondeficit=	○ 3 34

Submit

Fig. 6 Screenshot of CALM tool

While the XML persistence document contains all the details of layout and association information and facilitates collaborative online assessment development and assessment dissemination, it is not suited for end user manipulation. To facilitate quick online assessment creation, a simple declarative XML language for creating online assessments is also provided. With slightly more than just the question and answers text, an end user can define a multi-paged assessment in this simple declarative XML language. The CALM GUI understands both declarative and full assessment XML formats, and creates and integrates the corresponding online assessment with the HID web application automatically. CALM can further import Excel workbooks containing assessment metadata worksheets and/or actual assessment data. When only assessment metadata (assessment scores, questions, possible answers and answer types) is provided, the import wizard allows the user to select from eight different question layouts to customize the displayed online assessment form. The automatically created layouts are adequate for most purposes and can be easily fine tuned in the CALM layout editor. If actual assessment data is provided, the CALM import wizard populates the underlying Oracle or Postgres HID instance, making the assessment data available via the HID application for the researchers. This functionality is especially convenient for retrospective assessment data.

Once an on-line form is laid out, an assessment is generated and/or associated with form input elements, CALM generates code for HID application. CALM generates one JSP for each page of the on-line assessment and one Struts Form bean per on-line assessment. It also updates the Struts configuration file adding the new on-line assessment into the HID data management system. In earlier versions, CALM also generated an XForms 1.0[12] compliant document and an XSLT[13] stylesheet to transform the XForms document into JSP. Since the majority of the web browsers do not support XForms, currently JSPs are directly generated.

The lifecycle of a clinical assessment in the HID toolkit is depicted in Fig. 7. While CALM is used offline to build online versions of new assessments, GAME is responsible for managing the lifecycle of the CALM-registered on-line forms within the HID application. GAME automatically recognizes available on-line assessments at runtime from the Java byte code, and queries the CALM generated Struts form beans for meta-data used in managing the on-line-assessment wizard pages.

Query Builder

The end user query interface is based on an extensible Web 2.0 dynamic interface guiding the user in query building. Currently, two generic query builders are available in the HID application. The assessment query builder is responsible for collecting a clinical assessment user query and retrieving/combining results from multiple resources in the federation. The analysis query builder is responsible for collecting user search criteria for parameters on analysis

[12] http://www.w3.org/TR/xforms/.

[13] http://www.w3.org/TR/xslt/.

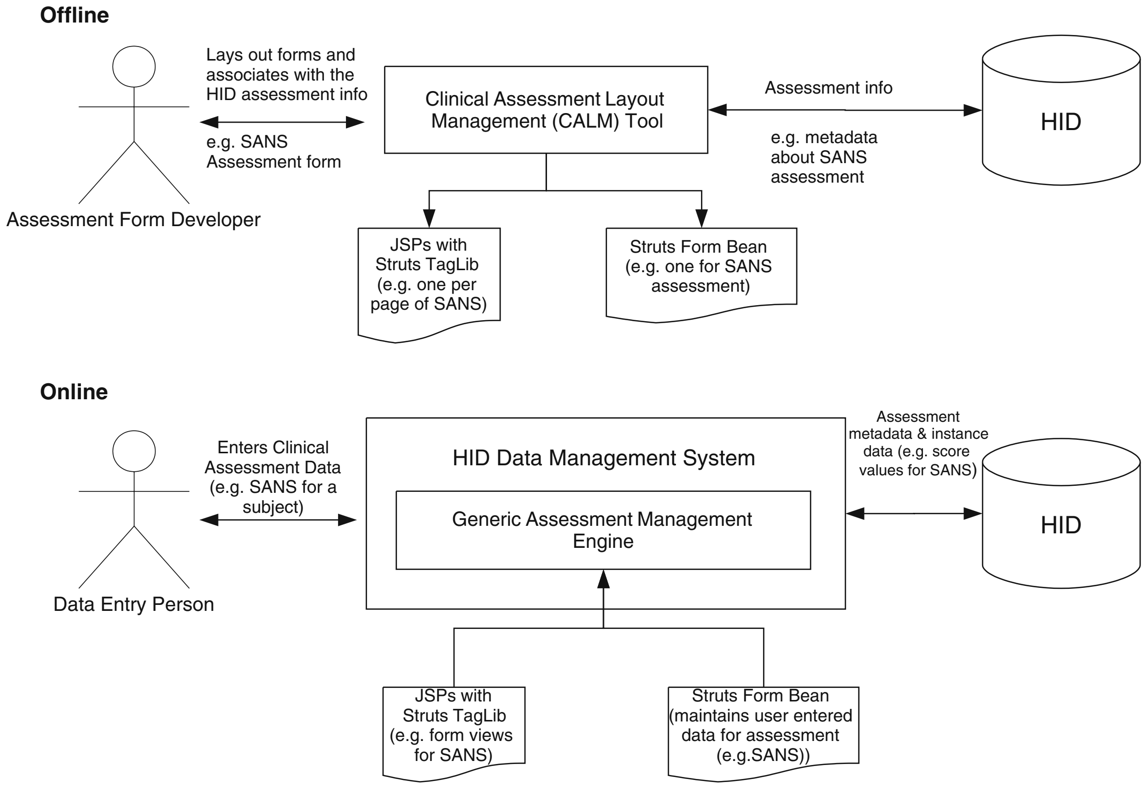

Offline
Assessment Form Developer
Lays out forms and associates with the HID assessment info
e.g. SANS Assessment form
Clinical Assessment Layout Management (CALM) Tool
Assessment info
e.g. metadata about SANS assessment
HID
JSPs with Struts TagLib (e.g. one per page of SANS)
Struts Form Bean (e.g. one for SANS assessment)
Online
Data Entry Person
Enters Clinical Assessment Data (e.g. SANS for a subject)
HID Data Management System
Generic Assessment Management Engine
Assessment metadata & instance data (e.g. score values for SANS)
HID
JSPs with Struts TagLib (e.g. form views for SANS)
Struts Form Bean (maintains user entered data for assessment (e.g.SANS))

Fig. 7 Lifecycle of on-line clinical assessment forms generation/usage process

results and on analyses themselves, and serving the results to the user. A user can also build queries combining both assessments and analysis-derived data.

A user query is expressed internally as an abstract syntax tree (AST) and converted to the corresponding SQL query (ies) by query builder classes specific to the type of the queries that will be built. The query builders use the Visitor design pattern (Gamma et al. 1994) to recursively visit the AST nodes and build a complex SQL query corresponding to the user's query. The query builder classes use registered implementations of ISQLDialect interface to handle relational database management system specific SQL dialects depending on the type of database to which the query will be submitted. A query spanning multiple remotely located and possibly heterogeneous mixtures of databases is made possible by this architecture.

The HID application generates SQL queries tailored for each type of database in the federation, sends generated queries to their corresponding databases in parallel, maximizing throughput, and combines the returned results. The user can navigate through the search results, drill down, and do preliminary univariate (descriptive statistics) and bivariate statistical analysis (regression analysis) with Web 2.0/AJAX based live charts. The results can be exported in CSV format suitable for SPSS or other statistical packages for further statistical analysis.

Imaging and Derived Data

The HID application uses a plugin mechanism to interact with different storage mechanisms. It stores location agnostic pointers (logical names) to the image resources stored in SRB, GridFTP or local and mounted file systems. The end user can query the HID using the web-based user interface for projects and subjects matching the specified criteria and download the corresponding imaging data optionally bundled with the matching clinical data. Using a common "shopping cart" paradigm, the HID application allows the end user to select image data for the subjects of interest and schedule a job for the HID application to create a compressed bundle of all related image files and/or clinical information. Large bundles will consist of multiple parts to facilitate download. The system notifies the user both by email and visually (if still online) when the bundle is ready to be downloaded.

The HID applications' generic job scheduler has an AJAX[14] based web front-end for job notification, monitoring and management. New jobs can be quickly and easily registered with the job scheduler. The job scheduler uses an intermediate cache for preparation of the bundles and to maximize user perceived throughput. Since intermediate

[14] Asynchronous Javascript + XML.

cache storage requirements can be substantial due to the size of FMRI image data sets, the job scheduler monitors the available cache storage and adjusts its cache management strategy accordingly. Cache management strategies are easily extensible and configurable. The default cache management policy to date, however, has been more than adequate for fBIRN usage patterns. Also, an AJAX based image series viewer is provided to allow previewing of structural image series in the web browser.

Generic derived data management provided by HID application consists of three main parts. 1) Import of the derived data and corresponding analysis provenance data presented in XML-Based Clinical Experiment Data Exchange Schema (XCEDE[15]) (Keator et al. 2006) format. 2) Storage of this data in HID using extended tuple data model (See "Data Model") 3) Generic analysis query builder and AJAX based front-end for end user query building. While the derived data management can handle any type of analysis in XCEDE format, XCEDE does not model metadata for presentation layout. This information is necessary to create an intuitive query builder interface for querying a particular analysis. Active development and improvement in this front is ongoing. Currently the analysis query builder view is targeted only for a limited set of analyses.

Web Services

A document centric approach is used for communicating large structured data imports and exports such as clinical assessments between HID applications and client applications. XCEDE is used as the lingua franca to exchange these kinds of data with third party clients. The HID app provides a remote procedure call (RPC) SOAP API and publishes its API as Web Services Definition Language (WSDL) documents through its web interface under the context path clinical/services. Currently, the HID web services target three client types (see Fig. 5);

1. 3DSlicer/Query Atlas (Brown et al. 2004) uses REST based web services to query for image data to visualize/analyze Freesurfer segmentation results. A lightweight proxy server based on Restlet framework[16] is developed to translate REST requests to SOAP requests and SOAP responses to REST responses.
2. The tablet PC based assessment data entry system developed at University of New Mexico uses HID web services to import clinical assessments/register subjects in experiments.
3. Command line clients distributed with the HID data management system to import derived data with provenance and workflow information.

The web services are made available via an open source SOAP server (CXF[17]) integrated with the HID application. Since image transfer over SOAP is not efficient, image transfers are done directly over HTTPS bypassing SOAP.

Data Federation

The HID application was designed to be used as a standalone neuroimaging data management system and in a federated environment for multisite projects. In a federated environment such as FBIRN, each site is responsible for its own imaging and clinical data, and access to the accumulated data across all the sites are coordinated by HID applications running at each site. If there are N sites in a multisite project, each site's HID application registers to N-1 other sites. A site administrator registers a remote site using the HID configuration console by providing remote database connection information. Each site in the federation provides a read-only database user to allow database federation. For distributed cache configuration, N-1 HID application Internet addresses need to be registered using the configuration console. After this one-time registration process, each site can do queries spanning the whole federation using the query builder views provided. There is no difference between the local and multi-site query building process. The user only needs to select the appropriate query selection (local or multisite) from the user interface. The capabilities have been successfully tested in the geographically diverse FBIRN federation and in smaller, locally distributed installations.

For best end user experience, the federated queries should not take much longer than the local queries, which is a major challenge in a geographically diverse federation. To guide the implementation and optimization of the federated query system and identify potential performance bottlenecks, the following tests were devised;

- Query for all subjects having quick mood scale and matching values for two, four and eight scores, locally.
- Query for all subjects having quick mood scales which have matching values for two, four or eight scores between multiple sites from west coast, east coast and mid-west USA.
- Query for all subjects having quick mood scale and demographics scale scores with matching values for one, two and four scores from each assessment locally.

[15] http://www.xcede.org/.

[16] http://www.restlet.org/.

[17] http://cxf.apache.org/.

- Query for all subjects having quick mood scale and having demographics scale with matching values for one, two or four scores from each assessment between multiple sites.

Three FBIRN sites were involved in data federation performance tests. The HID systems at the University of California at Irvine (UCI), Duke, and the University of Minnesota (UMN), as all had similar amounts of assessment data in their databases and located in the west coast, east coast and mid-west USA, respectively. The query performance times including dynamic query preparation, retrieving connection from pool, query execution, result preparation/merging is recorded by a profiling tool (JAMon[18]) embedded into the HID application. The results, provided in "Federated Query Performance Analysis", were instrumental in the optimization of the query system of HID application. When these performance analyses were conducted, none of the site databases had indices besides the primary keys. In the light of this analysis and usage pattern analysis, hot spot queries were detected and indices for database query performance improvements were deployed. The current software distribution includes these indices. During system setup, these indices are applied to the HID schema.

Security

The HID application provides an application level role based security mechanism for authorization, combined with a custom JSP tag library to be used in the presentation layer. Each user can be assigned privileges currently ranging from read-only access to administrator access. The write privileges do not extend beyond the local HID to the federation. Currently there are no experiment level privileges, i.e. any user with read privilege can see all of the experiments. This behavior was deliberate to facilitate anonymized research data sharing. Data anonymization involves stripping image headers and defacing of structural images. A quantitative study (Bischoff-Grethe et al. 2007) indicates that 3D reconstructed defaced images are not identifiable and defacing left the brain intact for analysis. Each subject is assigned an eight digit secure random BIRN ID, before any acquisition. That subject is identified from that point on only by the BIRN ID, which is not correlated with any subject information.

Interface based decoupled authentication and authorization services supports extension and/or replacement of these mechanisms. As of release 2.2, Grid Security Infrastructure (GSI) authentication plug-in module is available for GridFTP support. GSI provides certificate based public/private key authentication. The communication between client browser and web services client program and HID application server is secured by using HTTPS protocol.

Web Access

The HID application web interface provides an intuitive, integrated view of the data stored in the HID. Based on the metadata stored in the HID and the privileges/roles assigned to the authenticated user, the HID web interface dynamically builds and adapts it views. For example, an administrator can access user/database/application configuration panels as shown in Fig. 8a, b. A researcher can use the assessment and/or derived data query builder views (see Fig. 8c, d) to build local or mediated multi-site queries to find interesting datasets for download and/or preliminary statistical analysis. Based on the query results, a researcher can select image series and clinical assessments to download either using the fine grained shopping cart view (Fig. 9a) or coarse-grained batch download job submission view. The batch download functionality was developed to facilitate image download in a dynamic, federated environment. Because a user cannot be guaranteed that all HID installations will transfer images at the same rate, a user could potentially wait many hours to get a large FMRI dataset from geographically distributed HIDs. The dynamic job management view allows the user to login to the application at a later time and check the status of image download (Fig. 9b). When the job is complete, the bundled image (and clinical assessment) dataset is ready to transfer from the web server to the local client machine. Additionally, a researcher can drill down into search results to view clinical and acquisition protocol information and/or preview acquired image series (Fig. 9c).

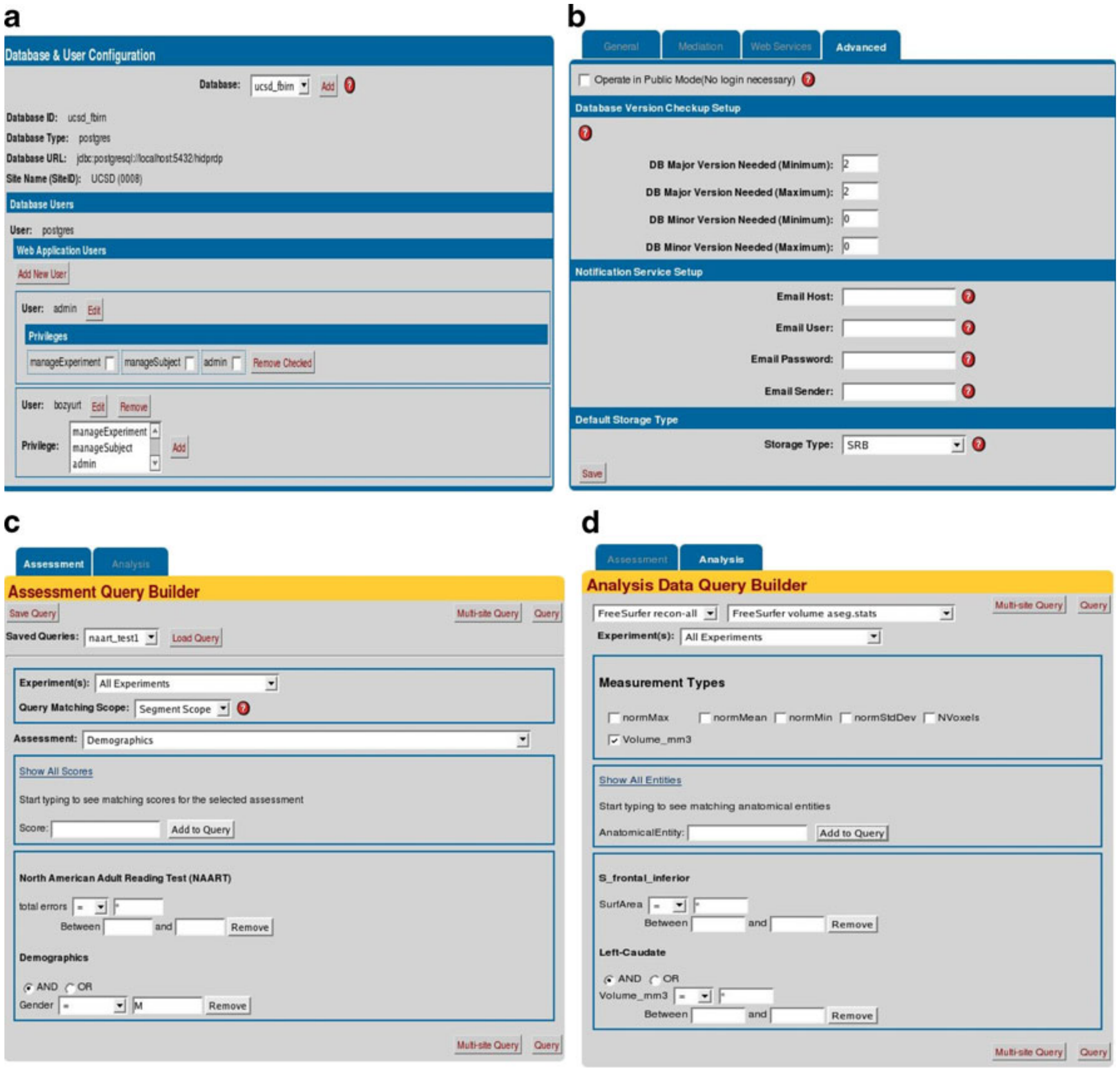
A data entry person can use the subject/visit management screens to enter clinical assessments as generated by CALM (Fig. 9d). For increased validation and reliability, clinical assessments are double entered and discrepancies are captured and reconciled by the system. Multiple imaging projects (experiments) can be managed via the Experiment/Subject management screens and subjects can be shared between projects.

Deploying HID Application

The HID application is an open source application freely available under BSD/BIRN license. The system is hosted at the Neuroimaging Informatics Tools and Resource Clearinghouse (NITRC) site[19] and includes technical documentation, end user tutorials, setup guides, wiki and mailing lists.

[18] http://jamonapi.sourceforge.net/.

[19] http://www.nitrc.org/projects/hid/.



Database & User Configuration

Database: ucsd_fbirn

Database ID: ucsd_fbirn

Database Type: postgres

Database URL: jdbc:postgresql://localhost:5432/hidprdp

Site Name (SiteID): UCSD (0008)

Database Users

User: postgres

Web Application Users

User: admin

Privileges

manageExperiment □ manageSubject □ admin □

User: bozyurt

Privilege: manageExperiment manageSubject admin

□ Operate in Public Mode(No login necessary)

Database Version Checkup Setup

DB Major Version Needed (Minimum): 2

DB Major Version Needed (Maximum): 2

DB Minor Version Needed (Minimum): 0

DB Minor Version Needed (Maximum): 0

Notification Service Setup

Email Host:

Email User:

Email Password:

Email Sender:

Default Storage Type

Storage Type: SRB

Assessment Query Builder

Saved Queries: naart_test1

Experiment(s): All Experiments

Query Matching Scope: Segment Scope

Assessment: Demographics

Start typing to see matching scores for the selected assessment

Score:

North American Adult Reading Test (NAART)

total errors = *

Between and

Demographics

● AND ○ OR

Gender = M

Analysis Data Query Builder

FreeSurfer recon-all FreeSurfer volume aseg.stats

Experiment(s): All Experiments

Measurement Types

□ normMax □ normMean □ normMin □ normStdDev □ NVoxels

☑ Volume_mm3

Start typing to see matching anatomical entities

AnatomicalEntity:

S_frontal_inferior

SurfArea = *

Between and

Left-Caudate

● AND ○ OR

Volume_mm3 = *

Between and

Fig. 8 HID application web interface screens for user/database/application configuration and end user query building

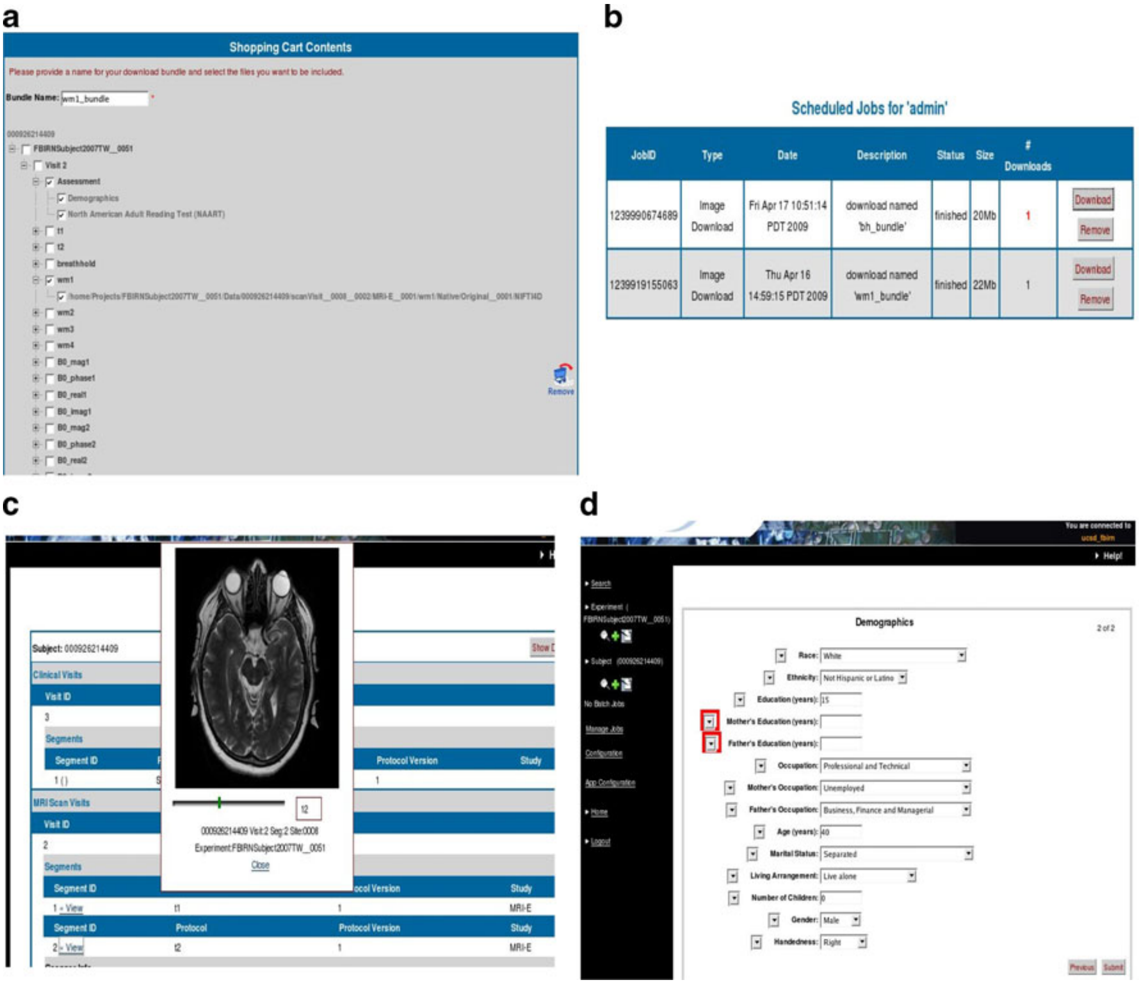
The HID application is designed to be setup with ease requiring no programmatic tweaking for an immediately usable system. The system as distributed can be installed following the quick start guides within minutes. It comes with support for twenty clinical assessment types which need to be registered using CALM with the site's database before usage. The configuration console of the HID web application handles all user/database/application configurations, which include joining the federation of HIDs for a multisite study where each site maintains its data autonomously within its HID.

While the meta-data driven design of the system minimizes the need for programmatic extension of the system, the session façade layer of the system acts as an application programming interface (API) for extension and is scriptable using Groovy,[20] a dynamic scripting language. Examples of scripts for building supporting/administration tools using the core HID application are provided with the source distribution. The HID application web interface can be extended with new screens and functionality by creating a Struts Action, a Form object and one or more JSPs while only interacting with the session façade layer.

Results and Discussion

To determine the usability of such a system in a large distributed setting, the system was tested on two distinct populations collected by the FBIRN consortium. Each population brings with it unique requirements and challenges to the data management system. The first population (Phase 1) was used to assess the sources of variability among consortium sites by an escalating series of phantom studies, small-scale human phantom studies, and larger scale healthy control human studies using the consortium suite of cognitive

[20] http://groovy.codehaus.org/.



Shopping Cart Contents

Please provide a name for your download bundle and select the files you want to be included.

Bundle Name: wm1_bundle *

```
000926214409
⊟ □ FBIRNSubject2007TW__0051
   ⊟ □ Visit 2
      ⊟ ☑ Assessment
            ☑ Demographics
            ☑ North American Adult Reading Test (NAART)
      ⊞ □ t1
      ⊞ □ t2
      ⊞ □ breathhold
      ⊟ ☑ wm1
            ☑ /home/Projects/FBIRNSubject2007TW__0051/Data/000926214409/scanVisit__0008__0002/MRI-E__0001/wm1/Native/Original__0001/NIFTI4D
      ⊞ □ wm2
      ⊞ □ wm3
      ⊞ □ wm4
      ⊞ □ B0_mag1
      ⊞ □ B0_phase1
      ⊞ □ B0_real1
      ⊞ □ B0_imag1
      ⊞ □ B0_mag2
      ⊞ □ B0_phase2
      ⊞ □ B0_real2
```

Scheduled Jobs for 'admin'

JobID	Type	Date	Description	Status	Size	# Downloads	
1239990674689	Image Download	Fri Apr 17 10:51:14 PDT 2009	download named 'bh_bundle'	finished	20Mb	1	Download Remove
1239919155063	Image Download	Thu Apr 16 14:59:15 PDT 2009	download named 'wm1_bundle'	finished	22Mb	1	Download Remove

Subject: 000926214409

Clinical Visits

Visit ID			
3			

Segments

Segment ID	P	Protocol Version	Study
1 ()	S	1	

MRI Scan Visits

Visit ID			
2			

Segments

Segment ID		ocol Version	Study
1 - View	t1	1	MRI-E

Segment ID	Protocol	Protocol Version	Study
2 - View	t2	1	MRI-E

t2
000926214409 Visit:2 Seg:2 Site:0008
Experiment:FBIRNSubject2007TW__0051
Close

You are connected to ucsd_fbirn

▸ Help!

▸ Search
▸ Experiment (FBIRNSubject2007TW__0051)
▸ Subject (000926214409)
No Batch Jobs
Manage Jobs
Configuration
App Configuration
▸ Home
▸ Logout

Demographics 2 of 2

Race:	White
Ethnicity:	Not Hispanic or Latino
Education (years):	15
Mother's Education (years):	
Father's Education (years):	
Occupation:	Professional and Technical
Mother's Occupation:	Unemployed
Father's Occupation:	Business, Finance and Managerial
Age (years):	40
Marital Status:	Separated
Living Arrangement:	Live alone
Number of Children:	0
Gender:	Male
Handedness:	Right

Previous Submit

Fig. 9 HID application web interface example screens. **a**) image/clinical data download shopping cart **b**) download job monitoring, **c**) subject data viewing **d**) assessment data entry

challenge tasks. The Phase 1 healthy control subject population consisted of five subjects scanned twice at each of nine clinical imaging sites. Both functional and structural imaging techniques were used along with seven clinical assessments were collected electronically on each subject using data entry forms designed in CALM. The second population (Phase 2) tested the usefulness of a federated database in studying functional brain imaging in a large clinical population. The Phase 2 population consisted of 122 schizophrenic and 128 healthy control subjects. Nineteen clinical assessments were collected electronically on each subject using data entry forms designed in CALM resulting in over 3,000 online assessment entries. Clinical assessments ranged from simple demographics forms to more complex multi-page assessments such as the Scale for the Assessment of Negative Symptoms (SANS), Scale for the Assessment of Positive Symptoms (SAPS), and the Structured Clinical Interview for DSM-IV (SCID). All data were double entered and validated using the HID application. Finally, performance tests were run over the federated database infrastructure to determine query response time and feasibility of using such a system in larger clinical trials.

FBIRN Processing Workflow Using HID Application

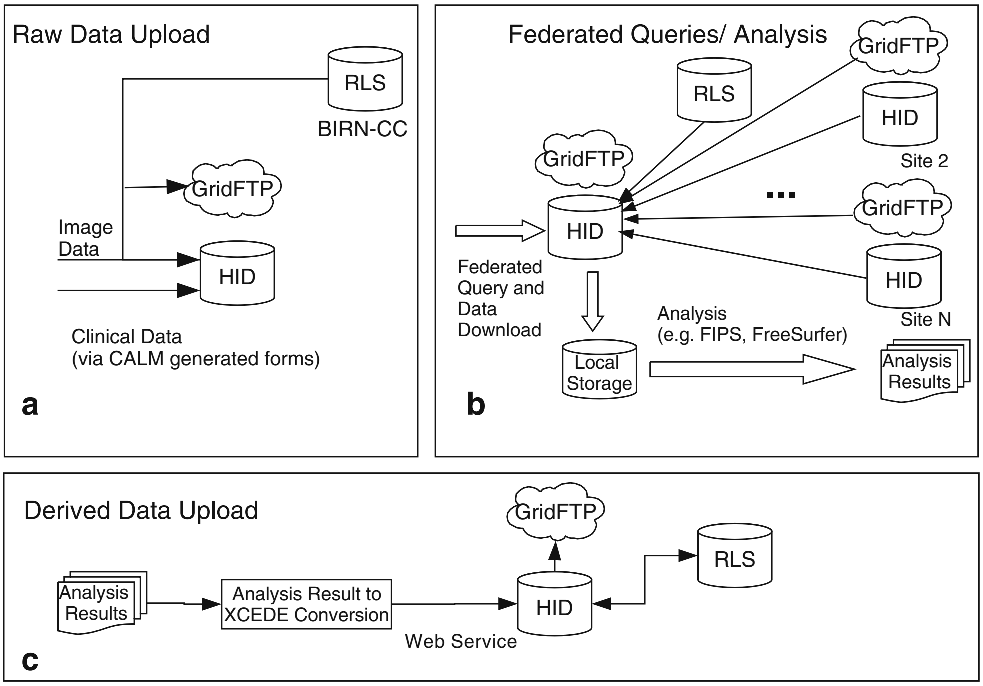
A typical workflow for managing (multi-site) imaging study using the HID application is depicted in Fig. 10. Imaging data acquired by collection devices such as MRI or PET scanners are entered to the system (HID and distributed storage system Storage Resource Broker (SRB) or Globus GridFTP) via the upload script developed by FBIRN. The upload script is also available from NITRC[21] and supports many native MRI scanner file formats from major vendors including GE, Siemens and Phillips along with standard formats such as DICOM[22] and NIFTI.[23] It also provides image de-identification via a rule-based

[21] http://www.nitrc.org/projects/fbirn (svn co https://www.nitrc.org/svn/fbirn/trunk/DataUpload).

[22] http://medical.nema.org/.

[23] http://nifti.nimh.nih.gov/.

Fig. 10 FBIRN processing workflow using HID application



DICOM header anonymizer and defacing capability for anatomical images based on Freesurfer defacing software (Bischoff-Grethe et al. 2007). Native formats are converted to NIFTI during the upload process to facilitate image use in the largest number of image analysis tools. A well-defined directory hierarchy in support of automatic analysis workflows and data sharing and maintenance is used in the distributed (or local) storage system. The logical locations of image resources are also registered with the HID by the upload script. For location independent access to GridFTP resources, the logical locations are also registered with the Replica Location Service (RLS) provided by BIRN. Two levels of quality assurance are applied to the image data. First, upload script checks image headers for errors against acquisition specific rules. Second, a quality assurance script, also developed by FBIRN, checks the image series for many quality measures including spikes and signal-to-noise ratio. Next, a human expert checks these results and makes a go/no-go decision. Clinical/genetic data associated with the scan sessions are entered into the HID application via CALM generated online assessment forms in double entry fashion or Tablet PC based mobile assessment entry system currently being tested by the FBIRN consortium (See Fig. 10a).

A researcher can then query the system using its web interface to select a portion of data matching specific criteria and download a bundle of image and assessment data for analysis. Analysis, visualization, and workflow tools, including 3DSlicer, Freesurfer or FIPS, can then be used on the custom data set retrieved from the HID data management system. These interactions are depicted on Fig. 10b. The resulting derived data including segmentation/parcellation

Fig. 11 Execution times for local single assessment queries

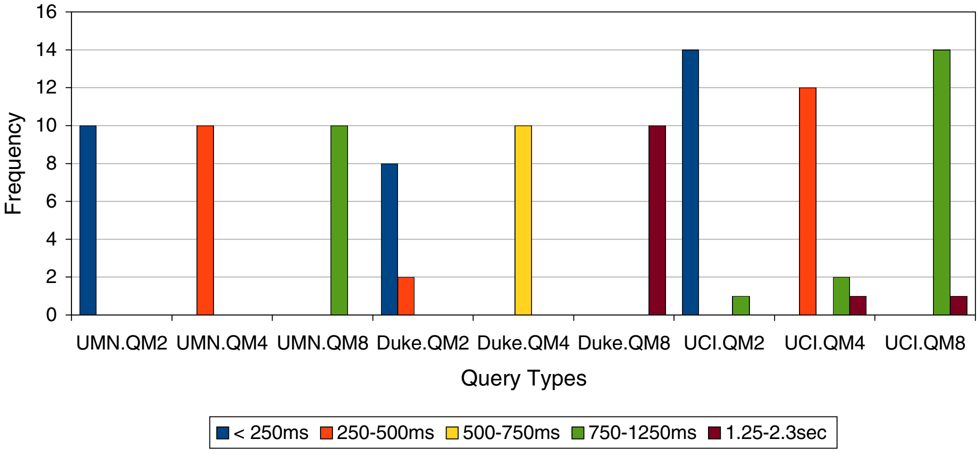
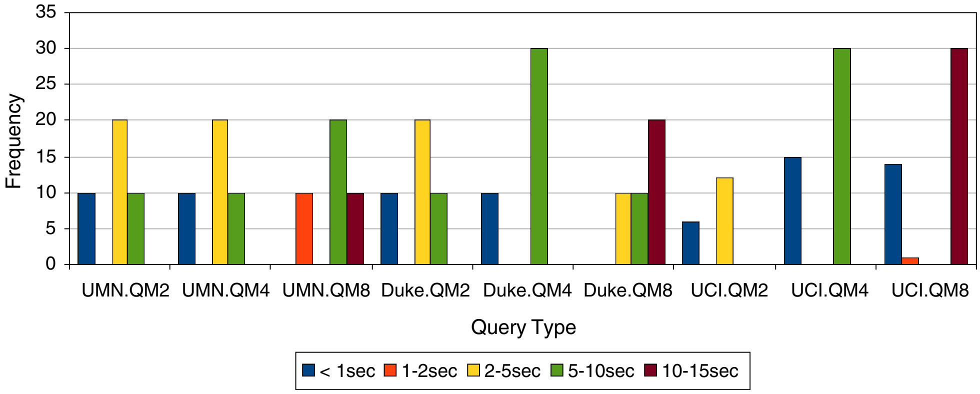


Fig. 12 Execution times for multi-site single assessment queries initiated from the local site



results and volumetric measures are then imported into the system using XCEDE2 formatted files via the HID web service interface (Keator et al. 2006) as shown in Fig. 10c.

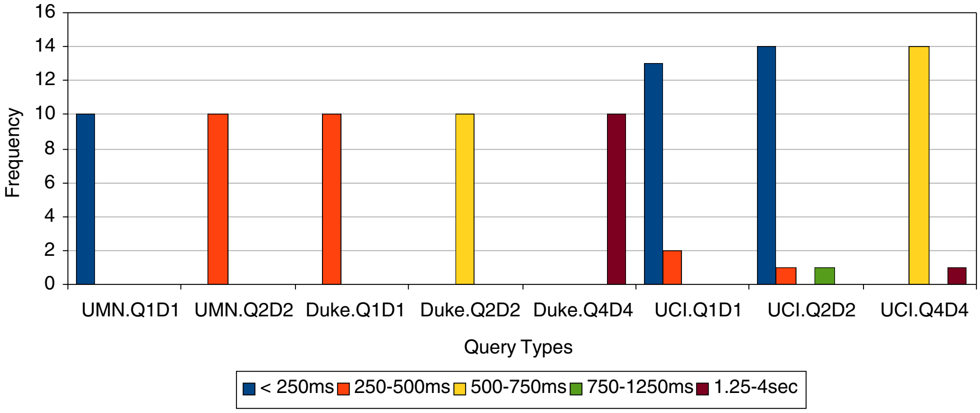
Federated Query Performance Analysis

The query performance tests are summarized in Figs. 11, 12, 13 and 14. It is evident that the multi site queries show multiple modes, while the local queries are unimodal. The query time distribution for multi-site queries shows at least two distinct modes, one cluster for local site and the other for the remote site queries. These differences can be attributed mostly to network latency and other connection related issues, as the local queries of the same type takes much less time irrespective of the local site. The results for the two types of queries with same overall number of scores but differ only in whether the scores come from one or two assessments gives insight into the efficiency of the overall SQL query. Using a two-sample *t*-test under the assumption that the query time distribution is Gaussian and the variances are not equal, pairs of local query times for equal number of scores (e.g. UMN.QM2 vs. UMN.Q1D1) are tested for the null hypothesis that there is no difference in average query times. For UMN and Duke data, at significance level α=0.005, for every pair the null hypothesis is rejected (largest p=0.00127). For UCI's data, due to the one outlier (startup caching effect) the variance is very high and, hence the null hypothesis was not rejected. The data from UMN and Duke suggests that the portion of the query which combines results from multiple assessments scales better than the inner query responsible for filtering the records belonging to a single assessment with increasing number of scores.

Another result from these data is that the queries scale linearly with increasing number of scores (query complexity). To provide evidence in support of this claim, the query performance data is transformed to average query time ratios and simple linear regression is applied and the adjusted R^2 values and line slopes noted. The adjusted R^2 ranges between 0.9441 and 0.9982 indicating that a linear model explains most of the relation between number of variables and query time ratio. The slope of the line for local queries range between 0.88 and 1.33 ($x = 1.13$ and s^2=0.025). The slope of the line for multi-site queries range between 0.13 and 0.50 ($x = 0.24$ and s^2=0.021).

Multisite queries on the average seem to perform similarly independent of the site of the origin. However, no hypothesis testing can be done, due to the multimodal nature of the multi-site data; Gaussian distribution assumption is definitely invalid. Sign and Ranking tests, which do not make normal distribution assumptions, cannot be used because of the lack of individual data point information from the JAMon profiling tool reports.

Fig. 13 Execution times for local multi assessment queries



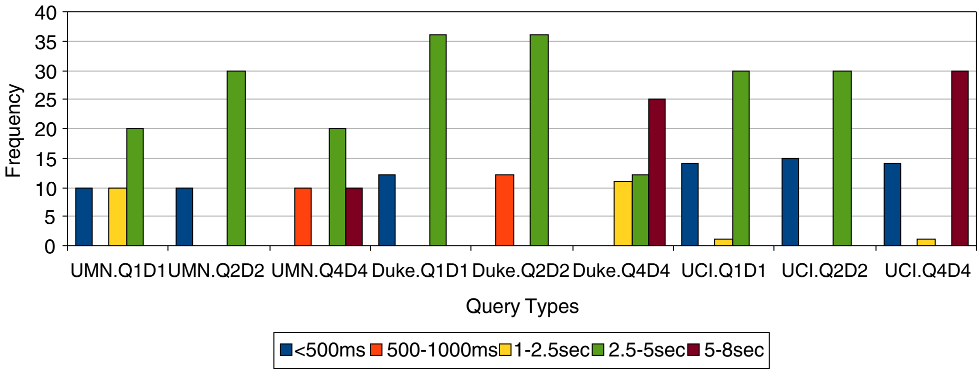


Fig. 14 Execution times for multi-site multi assessment queries initiated from the local site

Discussion

The HID application is designed with the following goals in mind; 1) the system should be extensible by nonprogrammers; 2) it should be easy to install and use; 3) it should be able work in multiple operational modes including a central repository, loosely coupled federated multisite mode, and for single site usage only; and 4) it should perform well even for large data sets such as clinical FMRI data.

Most of these goals are conflicting with each other in practice, hence a careful consideration of tradeoffs and optimizations, which are not necessary for more traditional data management systems, were required during the design and implementation of the system. Generality/extensibility at the end user level required a layered architecture as robust to changes at every system level as possible, with end user level tools not only providing intuitive user interfaces but also heavy and sometimes proactive validation. A metadata driven database schema with abstract data types allowed extensibility without schema modification. Schema changes were only necessary if an unforeseen abstract data type and/or extension to an existing one is needed and occurred much less frequently than metadata based extensions. The design of the system and its code generator tool handles the use-cases well.

The system has been deployed at eleven FBIRN sites for many years with many sites having multiple installations for other collaboratory projects. The FBIRN testbed provided valuable experience in usability and deployment, highlighting the importance of automated testing and continuous integration. A testing framework with idempotent system/database tests have been developed and are distributed with the system, providing self test capabilities. For continuous integration and automatically running system/database tests providing a more robust codebase, CruiseControl[24] is used.[25] Special emphasis is put in the optimization of the system performance using best practices for mission critical enterprise level system development, minimizing costly remote calls in federated operation mode and using caching and cache synchronization in a sensible manner.

While the focus and main development goal of HID application and its associated toolkit was to serve and manage neuroscience imaging studies, the underlying schema and the HID framework can be used and built upon in other scientific data management areas. Existing use of the HID across various locations includes the incorporation and integration of data from assessments of any type (clinical psychology, survey results, etc.), derived data values (any numeric values, for example), and association with data files of any size and location.

The RDBMS agnostic approach taken by the HID application allows for incorporation/interchanging of new database technologies such as vertical storage systems with hardware cache-conscious algorithms, for example MonetDB (Boncz et al. 2008), which have been demonstrated to perform better than traditional RDBMS systems like Oracle and PostgreSQL, especially for data mining/scientific applications. The HID application has performed well in the FBIRN collaboratory and has become a vital tool in supporting the multi-site FMRI studies.

Information Sharing Statement

For further information or to download the HID toolkit and HID schema please visit the Neuroimaging Informatics Tools and Resource Clearinghouse (NITRC) site: www.nitrc.org/projects/hid/. The latest release is 2.2 for HID schema and 2.2beta for HID toolkit. They can be downloaded from the source control by the following commands

```
svn co https://www.nitrc.org/svn/hid/clinical/tags/release-2.2beta
svn co https://www.nitrc.org/svn/hid/schema/tags/release-2.2
```

[24] http://cruisecontrol.sourceforge.net/.

[25] https://loci.ucsd.edu/cruisecontrol/.

Acknowledgements This research was supported by 1 U24 RR021992 to the Function Biomedical Informatics Research Network (BIRN, http://www.nbirn.net) that is funded by the National Center for Research Resources (NCRR) at the National Institutes of Health (NIH).

The authors thank Syam Gadde at the Duke University for XCEDE Schema and HID application testing, Raul de La Garza at the MIND Institute, New Mexico for HID testing, Gregory G. Brown at the University of California, San Diego and Steven Potkin at the University of California, Irvine for their support on the work presented here, and the FBIRN consortium members for the data collection and support of the scientists and engineers involved in the project.

References

Alur, D., Crupi, J., & Marks, D. (2001). Core J2EE patterns: Best practices and design strategies. Prentice Hall.

Bischoff-Grethe, A., Ozyurt, I. B., Busa, E., Quinn, B. T., Fennema-Notestine, C., Clark, C. P., et al. (2007). A technique for the deidentification of structural brain MR images. *Human Brain Mapping, 28*(3), 892–903.

Boncz, P. A., Kersten, M. L., & Manegold, S. (2008). Breaking the memory wall in MonetDB. *Communications of the ACM, 51*(12), 77–85.

Brown, G. G., Pieper, S., Martone, M., Aucoin, N., Joyner, A., Bischoff-Grethe, A., et al. (2004). The query atlas: A brain referenced knowledge discovery tool. Annual Neuroscience meeting.

Brown, G. G., McCarthy, G., Bischoff-Grethe, A., Ozyurt, B., Greve, D., Potkin, S. G., et al. (2009). Brain-performance correlates of working memory retrieval in schizophrenia: a cognitive modeling approach. *Schizophrenia Bulletin, 35*(1), 32–46.

Ford, J. M., Roach, B. J., Jorgensen, K. W., Turner, J. A., Brown, G. G., Notestine, R., et al. (2009). Tuning in to the voices: a multisite FMRI study of auditory hallucinations. *Schizophrenia Bulletin, 35*(1), 58–66.

Friedman, L., & Glover, G. H. (2006). Reducing interscanner variability of activation in a multicenter fMRI study: controlling for signal-to-fluctuation-noise-ratio (SFNR) differences. *Neuroimage, 33*(2), 471–481.

Friedman, L., Stern, H., Brown, G. G., Mathalon, D. H., Turner, J., Glover, G. H., et al. (2008). Test-retest and between-site reliability in a multicenter fMRI study. *Human Brain Mapping, 29*(8), 958–972.

Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (1994). *Design patterns—elements of reusable object-oriented software*. Reading: Addison Wesley.

Hamilton, G. (Ed.) (1997). JavaBeans Specification Version 1.0.1. Sun Microsystems, Online.

Hastings, S., Oster, S., Langella, S., et al. (2005). A grid-based image-archival and analysis system. *Journal of the American Medical Informatics Association, 12*, 286–295.

Keator, D., Gadde, S., Grethe, J., Taylor, D., Potkin, S., & FIRST BIRN. (2006). A general xml schema and associated spm toolbox for storage and retrieval of neuro-imaging results and anatomical labels. *Neuroinformatics, 2*, 199–212.

Keator, D., Grethe, J. S., Marcus, D., Ozyurt, B., Gadde, S., Murphy, S., et al. (2008). A National Human Neuroimaging Collaboratory enabled by the Biomedical Informatics Research Network (BIRN). *IEEE Transactions on Information Technology in Biomedicine, 12*(2), 162–172.

Keator, D.,Wei, D., Gadde, S., Bockholt, J., Grethe, J. S., Marcus, D., et al. (2009). Derived data storage and rxchange workflow for large-scale neuroimaging analyses on the BIRN grid. Frontiers in Neuroinformatics. In press.

Kim, D. I., Manoach, D. S., Mathalon, D. H., Turner, J. A., Mannell, M., Brown, G. G., et al. (2009a). Dysregulation of working memory and default-mode networks in schizophrenia using independent component analysis, an fBIRN and MCIC study. *Hum Brain Mapp*.

Kim, D. I., Mathalon, D. H., Ford, J. M., Mannell, M., Turner, J. A., Brown, G. G., et al. (2009). Auditory oddball deficits in schizophrenia: an independent component analysis of the fMRI multisite function BIRN study. *Schizophrenia Bulletin, 35*(1), 67–81.

Magnotta, V. A., & Friedman, L. (2006). Measurement of Signal-to-Noise and Contrast-to-Noise in the fBIRN Multicenter Imaging Study. *Journal of Digital Imaging, 19*(2), 140–147.

Marcus, D. S., Olsen, T. R., Ramratnam, M., & Buckner, R. L. (2007). The extensible neuroimaging archive toolkit. An informatics platform for managing, exploring and sharing neuroimaging data. *Neuroinformatics, 3*, 11–33.

Marenco, L., Tosches, T., Crasto, C., Shepherd, G., Miller, P. L., & Nadkarni, P. M. (2003). Achieving evolvable web-database bioscience applications using the EAV/CR framework: recent advances. *Journal of the American Medical Informatics Association, 10*, 444–453.

Nadkarni, P. M., Marenco, L., Chen, R., Skoufos, E., Shephard, G., & Miller, P. (1999). Organization of heterogeneous scientific data using the EAV/CR representation. *Journal of the American Medical Informatics Association, 6*, 478–493.

Potkin, S. G., & Ford, J. M. (2009). Widespread cortical dysfunction in schizophrenia: the FBIRN imaging consortium. *Schizophrenia Bulletin, 35*(1), 15–18.

Potkin, S. G., Turner, J. A., Brown, G. G., McCarthy, G., Greve, D. N., Glover, G. H., et al. (2009). Working memory and DLPFC inefficiency in schizophrenia: the FBIRN study. *Schizophrenia Bulletin, 35*(1), 19–31.

Potkin, S. G., Turner, J. A., Guffanti, G., Lakatos, A., Fallon, J. H., Nguyen, D. D., et al. (2009). A genome-wide association study of schizophrenia using brain activation as a quantitative phenotype. *Schizophrenia Bulletin, 35*(1), 96–108.

Rajasekar, A., Wan, M., & Moore, R. (2002). Mysrb & Srb components of a data grid, Proceedings of the 11th IEEE International Symposium on High Performance Distributed Computing HPDC-11.

Wible, C. G., Lee, K., Molina, I., Hashimoto, R., Preus, A. P., Roach, B. J., et al. (2009). fMRI activity correlated with auditory hallucinations during performance of a working memory task: data from the FBIRN consortium study. *Schizophrenia Bulletin, 35*(1), 47–57.