Putting the pieces together: Integrated Research Data Management Using the LabKey Server

Jeanne Holden-Wiltse, MPH
Co-Director of Data Management for RPRC, NYICE, CBIM, CIBEM

Andy Straw, MS
BLIS Leader Developer/Administrator

Center for Integrative Bioinformatics and Experimental Mathematics
University of Rochester Medical Center
Agenda

• Background
• Motivating Problems
• Bio-Lab Informatics Server (BLIS)
• Integration – Value added
• LabKey Extension Mechanisms
• Considerations
Background

- Center for Integrative Bioinformatics & Experimental Mathematics, Dept of Biostatistics & Computational Biology, URMC (Hulin Wu - PI)
  - Provide data management and integration for infectious disease studies
  - Perform statistical and bioinformatics analyses
  - Develop novel statistical methodology
  - Mathematical modeling of host responses to influenza incorporating systems biology approaches
Motivation

- 2005 CBIM
- 2007 NYICE
- 2007 URMC D-CFAR, 2012 CFAR
- 2011 RRPC
- Several R01’s
**Bio-Lab Informatics Server (BLIS)**

- Customized LabKey Server (2010)
- Data Management Operations
  - Developers
  - Statisticians/Analysts
- 150 Users
- 8 Projects
- 27 studies (mostly longitudinal)
  - 3885 subjects, 14803 visits, 345 datasets, 129817 specimen vials, 48323 .fcs files
Integration/Management Goals

- Manage information/data from multiple sources, formats
- Timely, complete and clean
- Feedback
- Data Sharing
- Flexible, Adaptable, Responsive to changing research needs
- BLIS as active research member
Integration Requirements

- Subject - Visit
- Specimen
- Plate Assays
- Information
- ‘Omics

BLIS

Data Sharing

Reporting

HPC Storage
Integration - Information

• Study/Lab Portals
  – Protocol Docs, SOPs
  – Sharepoint
  – iCalendar
  – Google Maps

• Collaborative workspace
  – Manuscript development
Integration – Subject/Visit

- Human, Animal
- BLIS eCRFs
  - validation controls
- txt, csv, xls from
  - REDCap
  - CROs
  - Labs
Integration - Specimen

• Biospecimen Inventory
  – Requests
  – Tracking/Reporting
  – Contract closeout
• Unique specimen ID
• LDMS (LIMS)
• xls (small studies)
Integration – Assays

- ELISA
- ELIspot
- HAI, MN
  - Preload specimen IDs
- qtPCR
- Specimen Processing/extraction
- Frequent plate design changes
Integration - Omics

- Flow cytometry
- Microarray
- RNAseq
- 16S rRNA
- Imaging (coming)
- Files pipeline (raw and processed)
  - HPC
  - Large file archive
Integration - Reporting

- Monitoring/ Visit Check Lists
- QC checking
- NIAID reports
- Visualization - Charting
- Messaging
- Issue Tracker for data discrepancies
Integration – Data Sharing

• Datasets for analysis
  – SAS, R, Excel

• NIH data repositories
  – Specific formats

• Security –
  – Fine grain data sharing
  – FISMA requirements
LabKey Extension
Mechanisms We Use

• LabKey SQL
• LabKey import/export features
  – To/from Excel, TSV, CSV
  – Archive formats (Specimen; Flow Analysis; etc.)
• LabKey client APIs
  – Java (and HTTP) for external apps (used by developers)
  – JavaScript on wiki pages (used by end users)
• LabKey module mechanism
  – File-based
  – Java-based (use LabKey server APIs)
What We Have Implemented Using LabKey Extension Mechanisms

1. Framework for electronic case report forms
2. Automated import of specimen data from LDMS
3. Excel templates to minimize specimen data entry
4. QC dashboard for flow cytometry file metadata
5. Import of custom flow cytometry analysis file formats
6. Java-based custom assay modules
7. Extracting, formatting data for publishing to public databases
## Where We Use Extension Mechanisms

<table>
<thead>
<tr>
<th>Solution</th>
<th>Extension Mechanisms</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SQL</td>
</tr>
<tr>
<td>Subject / Visit</td>
<td></td>
</tr>
<tr>
<td>1. eCRF Framework</td>
<td>X</td>
</tr>
<tr>
<td>Specimen</td>
<td></td>
</tr>
<tr>
<td>2. Import from LDMS</td>
<td></td>
</tr>
<tr>
<td>3. Excel Templates</td>
<td>X</td>
</tr>
<tr>
<td>Assay / Omics</td>
<td></td>
</tr>
<tr>
<td>4. QC dashboard for flow file metadata</td>
<td>X</td>
</tr>
<tr>
<td>5. Import custom flow analysis file formats</td>
<td></td>
</tr>
<tr>
<td>6. Java-based custom assay modules</td>
<td>X</td>
</tr>
<tr>
<td>Data Sharing</td>
<td></td>
</tr>
<tr>
<td>7. Extract, format, publish to public DBs</td>
<td>X</td>
</tr>
</tbody>
</table>
1. Framework for Electronic Case Report Forms

Problem: Collect large amount of data from clinic/hospital visits

• Data is “similar but different” across multiple visits, multiple cohorts (need to re-use datasets, pieces of forms)
• Must be easy for nurses to collect data “live” via iPads
• Data validation is complex, but critical
  – Can’t “go back” to subjects later if data are missing or incorrect
  – eCRF data is source of record – no paper forms
• LabKey’s built-in “Insert New” and “Edit” forms are not enough
1. Framework for Electronic Case Report Forms

Solution: eCRF JavaScript framework that supports:
• Building sets of forms composed of reusable sections
• Dynamic showing/hiding based on visit and/or cohort
• Intuitive navigation
• Complex validation

Approach:
• LabKey DataSets – one per “section”
• LabKey JavaScript client API + wiki pages
• 3rd-party JavaScript libraries (e.g., Backbone for MVC)
• Framework of base “classes” => new forms require minimal code
1. Framework for Electronic Case Report Forms

Study Visit

Vital Signs

- Systolic Pressure: 43 mm Hg
- Diastolic Pressure: 52 mm Hg
- Heart Rate: 80 beats/min
- Respiration: 55 breaths/min
- Height: 53 inches
- Weight: 140 pounds
- Temperature: 95°F
1. Framework for Electronic Case Report Forms
2. Automated Import of Specimen Data from LDMS

LDMS:
• Sophisticated specimen inventory management system
• Used by several of the studies we support

Problem: LabKey studies use internal specimen repository
• Need to export data from LDMS, import to LabKey (multiple studies)
• Data in LabKey must be “reasonably” current
2. Automated Import of Specimen Data from LDMS

Solution: Combination of small manual steps, plus custom application to automate export/import process

Approach:
- Use LDMS reporting and export features to export data (manual)
- Use custom Java application to:
  - Transform exported data into LabKey specimen archive format
  - Split data into one archive per study
  - Move archive files to LabKey using WebDAV Java API
  - Start pipeline import job for each study using LabKey HTTP interface
- Done daily by developer (5 minutes)
3. Excel Templates to Minimize Specimen Data Entry

Background:

- Some studies we support do NOT use LDMS, but still require use of LabKey specimen repository.
- Sometimes: specimen IDs encode info

Problem: Tedious and error-prone for lab personnel to properly enter all required specimen info, in proper format.
3. Excel Templates to Minimize Specimen Data Entry

Solution:

• Agree with labs on how many of which type of specimens will be generated for each subject-visit combination

• Use custom Java application to generate partially-filled-in Excel templates, including specimen id, and other known info - one row per expected vial

• Lab personnel manually fill in rest (mainly: volume and location)
  – Lab personnel delete rows for any vials not actually generated

• Use custom Java application to transform filled-in Excel files into LabKey specimen archive format

• Use LabKey specimen archive format to import data into LabKey
3. Excel Templates to Minimize Specimen Data Entry

![Excel Template](image)
4. QC Dashboard for Flow Cytometry File Metadata

Background:
• Flow (FCS) files contain metadata in header keywords
  – Useful for storing specimen Id, experimental conditions (panel, stimulation, etc.)
• LabKey Flow module imports FCS files, reads and stores keyword metadata
  – One row per file, one column per keyword
• Labkey Flow module can use keyword value as specimen Id
  – LabKey automatically links to specimen repository of target study

Problem: Some metadata values are entered manually – errors result that must be detected and corrected
4. QC Dashboard for Flow Cytometry File Metadata

Solution: Dashboard page showing all detected errors

Approach:
• LabKey SQL queries (or views) to detect metadata errors
• JavaScript API to invoke queries, display errors on wiki page
• Shared, unambiguous communication of errors (no files, emails)

Downside:
• Metadata errors not detected until files imported =>
  Files must be re-imported after errors are corrected
4. QC Dashboard for Flow Cytometry File Metadata
5. Import of Custom Flow Cytometry Analysis File Formats

Background:

• LabKey Flow module supports multiple formats for importing analysis results (primarily: \#cells of various types, per sample)
  – FlowJo workspace (exported to XML)
  – FCS Express
  – A LabKey flow analysis archive (zip file) format
• We have investigators using other tools to analyze flow data
  – FLOCK (https://immpact.niaid.nih.gov)
  – SWIFT (http://www.ece.rochester.edu/projects/siplab/Software/SWIFT.html)
  – FlowJo, stats exported to Excel, aggregated from multiple workspaces

Problem: How to import these other analyses?
5. Import of Custom Flow Cytometry Analysis File Formats

Solution: Small amount of reusable custom Java code to transform other formats into LabKey’s archive format.

- Store analyses in native format in LabKey “Files” web part, for reference
- Transform data from native format to LabKey format via external Java program
- Create analysis archive (zip file), import via LabKey Flow module
6. Java-Based Custom Assay Modules

Background: LabKey supports import of experiment (assay) data

- Concepts of Batch, Run, Results
- General Purpose Assay supports:
  - Tabular data file (Excel, TSV, CSV)
  - One run per file
  - One row per result (i.e., experiment data per sample)
- Other LabKey assays support certain specific file formats

Problem: We need support for:

- Other file formats (e.g., XML)
- Files containing batch and run properties, in addition to results
- Ability to include calculated properties, not just those entered by user or found in file
6. Java-Based Custom Assay Modules

Solution: Use LabKey Module concept to add custom assay types
- File-based: less/no code, less flexible
- Java-based: mostly code, more flexible

Approach:
- Primarily Java-based modules
- Built set of our own base classes that use, extend LabKey server classes; Added some utility classes
- Development of new assay module requires minimal additional code (e.g., developed one in ½ day, < 200 lines of code – mostly config data)
- Caveat: Assays must fit within certain restrictions (though we have been progressively removing restrictions)
7. Extracting, Formatting Data for Publishing to Public Databases

Problem: Grants/Contracts require some study data be published to external databases in very specific formats

Solution: Custom external Java application

Approach:

• Use Java Client API to extract data from LabKey
• Use 3rd-party Java APIs to fill in Excel templates provided by external database
• Zip files, then upload via external database web site
Considerations

Lots of hooks provided by LabKey
• Significant learning curve: What they all are, what each is good for
• Significant power and flexibility
• Choice of hook may depend on:
  – The expertise your team members have
  – Where you want to touch the data (outside or inside LabKey)
    • E.g., prefer to fix assay metadata errors at source, but easier to detect them after import into LabKey
• Other hooks we have NOT fully explored:
  – Transformation scripts
  – File-based modules
  – UI customizations (e.g., adding buttons)
  – Study/Folder templates
Considerations

Required amount of software development expertise varies

• **Very little:**
  – Transforming flow analysis data from one TSV format to another (Java, Perl, Python, R – any will do)
  – QC dashboard for flow metadata (basic SQL + basic JavaScript/HTML)

• **A bit more:**
  – Automated import of specimen data from LDMS (Java, Swing, LabKey Java client API, 3rd-party Java APIs)

• **Even more + LabKey source code + help from LabKey personnel:**
  – Java-based custom assays (documentation of server APIs is minimal) (Each developer has own LabKey server, built from source)
Acknowledgements

Director: Hulin Wu, PhD
Co-Director: Jeanne Holden-Wiltse, MPH
Database & Software Developers
Andy Straw, MS (Lead, Admin)
Anthony Corbett
Jennifer Dutra
Jeffrey Williams
Alicia Tyrell

Biostatisticians/Bioinformaticians
Hongmei Yang, PhD
Joe Guido, MS
Lu Wang, MS
Sanjukta Bandyopadhyay, MS
Shuang Wu, PhD
Hongyu Miao, PhD
Xing Qiu, PhD
Le Zhang, PhD

University Support
Center Integrated Res Computing
Primary Data Center

Consultant
LabKey Software, Seattle, WA

ALL BLIS USERS

Funding

URMC CFAR