

Unified Electronic Logbook for 12GeV

Design and Features

Motivation/Benefits

- Common URL scheme to access any logbook or entry
 - (/book/halog, /book/hdlog/rss, /entry/1234567, etc.)
- A Common interface for programs to make entries
- Consistent records-management of logbooks
- Fewer total man-hours to maintain a single system vs. multiple systems.
- Ability to search for and display entries across logbooks (e.g. when troubleshooting or doing forensics after a downtime event)
- Sharing of logbook tools and widgets between Physics and Accelerator divisions.

The Design

- An electronic logbook system built on the open-source Drupal 7 web-based Content Management System.
 - In use by thousands of sites (<http://buytaert.net/tag/drupal-sites/>)
 - Already in use at JLAB (cc.jlab.org, cebaf.jlab.org, etc.)
 - Can be administered and extended by non-programmers
- A MySQL database on the back-end.
- An API and programming libraries to ease making entries from Java, Python, Perl, PHP, and C++

Logbook Content Management

- Log entries will be stored as custom Drupal node type(s)
 - Will inherit standard node features such as versioning

The screenshot shows the Drupal administration interface for the 'LogEntry' content type. The top navigation bar includes links for Content, Structure, Appearance, People, Modules, Configuration, Reports, Advanced help, and Help. The user is logged in as 'Hello admin' with a 'Log out' link. The breadcrumb trail is 'Home > Administration > Structure > Content types > LogEntry'. Below the breadcrumb, there are tabs for 'LogEntry', 'EDIT', 'MANAGE FIELDS', 'MANAGE DISPLAY', 'COMMENT FIELDS', and 'COMMENT DISPLAY'. The 'MANAGE FIELDS' tab is active, displaying a table of fields for the 'LogEntry' content type.

LABEL	NAME	FIELD	WIDGET	OPERATIONS
Horizontal Tab Group 1	group_htabs1	Horizontal tabs group		delete
Main	group_tab_main	Horizontal tab item	htab closed required_fields yes	delete
Log Number	field_lognumber	Serial	Hidden (Automatic)	edit delete
Title	title	Node module element		
Body	body	Long text and summary	Text area with a summary	edit delete
Logbook(s)	field_logbook	Term reference	Term reference tree	edit delete
Attachments	group_tab_attach	Horizontal tab item	htab closed required_fields yes	delete
Picture	field_image	Image	Image	edit delete
Attachment	field_attach	File	File	edit delete
More	group_tab_more	Fieldset	fieldset collapsible required_fields yes	delete
Entry Makers	field_entrymakers	Text	Text field	edit delete
Tags	field_tags	Term reference	Check boxes/radio buttons	edit delete
Log Reference	field_logref	Entity Reference	Autocomplete	edit delete
URL path settings	path	Path module form elements		
Add new field	field_	- Select a field type -	- Select a widget -	

Standard Drupal Modules can reduce custom coding

JEFFERSON LAB ELECTRONIC LOGBOOK

Sticky Entries

Log Number	Posted	Author	Title
1686988	08:12	chrysost	Hall C Ion Chamber readbacks frozen

Logbook(s): ELOG
Wednesday (09-May-2012)

Log Number	Posted	Author	Title
1783841	14:52	bcade	Autolog Fopt
1783839	14:50	spraggin	Outgoing page to Broecker - EES DC Days
1783838	14:36	hofer	Autolog Fopt failure
1783837	14:34	hofer	Autolog Fopt failure
1783836	14:33	hofer	Autolog Fopt failure
1783835	14:32	hofer	Autolog Fopt failure
1783834	14:28	spraggin	Autolog LEM North Linac
1783833	14:27	spraggin	Autolog LEM North Linac - failure
1783831	14:27	mwilson	SL25 Update C100 Work
1783832	14:27	Irishard	Autolog Fudget
1783830	14:22	spraggin	FSD Mask S-Linac RF P1s
1783829	14:22	spraggin	FSD Masks have been set for beam delivery to BSY
1783828	14:20	spraggin	QuickPic - dog7 elf
1783827	14:16	spraggin	Autolog LEM North Linac
1783826	14:11	kerns	COMMENT ON RF Cavity Recovery - 05/9/12
1783824	13:57	lharris	1L11-5 Latched FSD/P1 Line (Buffer Board Lights)
1783822	13:53	kerns	Cavities in 1L11 have been updated: 1L11-5
1783809	13:42	spraggin	Page returned from Kerns - EES RF Days
1783804	13:36	spraggin	Outgoing page to Kerns - EES RF Days
1783803	13:33	arames	(really) good nennn orbit

May 2012

Su	Mo	Tu	We	Th	Fr	Sa
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

Logbook Selections

Useful Links

- [Accel SAD Calendar](#)
- [ATLIs](#)
- [CEBAF Status Information](#)
- [Crew Chief Shift Logs](#)
- [Experiment Calendar](#)

The Drupal *Views* Module was used to generate the listing above

Drupal Module

- JLAB-specific features built using the Drupal API and installed as a Drupal module

☒ **Date Views** 7.x-2.5 Views integration for date fields and date functionality.
Requires: Date API (enabled), Views (enabled), Chaos tools (enabled)
Required by: Calendar (enabled)

▼ ELECTRONIC LOGBOOK

ENABLED	NAME	VERSION	DESCRIPTION	OPERATIONS
<input checked="" type="checkbox"/>	eelog		Module implementing features of the JLAB unified electronic logbook. Requires: Field extract (enabled), Views (enabled), Chaos tools (enabled)	

▼ FEATURES

ENABLED	NAME	VERSION	DESCRIPTION	OPERATIONS
<input type="checkbox"/>	Date Migration Example	7.x-2.5	Examples of migrating with the Date module Requires: Date (enabled), Date API (enabled), Date Repeat API (disabled), Date Repeat Field (disabled), Date Migration (disabled), Migrate (missing), Features (enabled)	
<input checked="" type="checkbox"/>	Features	7.x-1.0-rc2	Provides feature management for Drupal. Required by: Date Migration Example (disabled), Features Tests (disabled), Feeds News (disabled), OG example (disabled)	? Help Permissions
<input type="checkbox"/>	Ldap Authorization SimpleTest	7.x-1.0-beta9	Ldap Authorization SimpleTest 1 Requires: LDAP Authorization (enabled), LDAP Servers (enabled), Strongarm (missing)	

API for making entries

- The Elog API consists of a set of published XML schemas that correspond to the different types of logentries that are possible.
 - <http://logbooks.jlab.org/schema/Logentry.xsd>
- Additional Schemas may be defined to extend Logentry for special-purpose log entries. For example an OPS-PR
 - <http://logbooks.jlab.org/schema/OPS-PR.xsd>
- A hall specific type of entry? Run entry?
 - ...

Client utilities

- Utility libraries will be made available so that clients do not need to deal with creating and validating xml.
- The following languages will be provided at the outset:
 - Java
 - Python
 - Perl
 - Php
 - C++
 - Tcl?
 - Shell Command

XML Example

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<Logentry>
  <title>My title</title>
  <created>2012-03-16T12:30:16-04:00</created>
  <body><![CDATA[This is the body of the entry. It could be many
paragraphs]]></body>
  <Logbooks>
    <logbook>ELOG</logbook>
    <logbook>CLOG</logbook>
  </Logbooks>
  <Entrymakers>
    <Entrymaker>
      <username>theo</username>
    </Entrymaker>
    <Entrymaker>
      <username>cuffe</username>
    </Entrymaker>
  </Entrymakers>
</Logentry>
```

Example: make an entry from a script

Client libraries simplify making entries so that programmers do not need to know the details of the XML schemas

- Three lines to make a minimal entry PHP example*

```
require_once('Logentry.inc.php');  
$myElog = new LogEntry('TLOG','My title - an entry with no other text');  
$lognumber = $myElog->submit();
```

- But can also easily add attachments, tags, etc.

```
$myElog->addAttachment('/usr/csmuser/theo/googlesnap02.png','Figure 1');  
$myElog->addTag('EndOfRun');  
$myElog->notify('user@jlab.org');  
$myElog->setSticky(true);
```

** This is just an example used during development that illustrates how to script entries. The officially released library may differ slightly in syntax.*

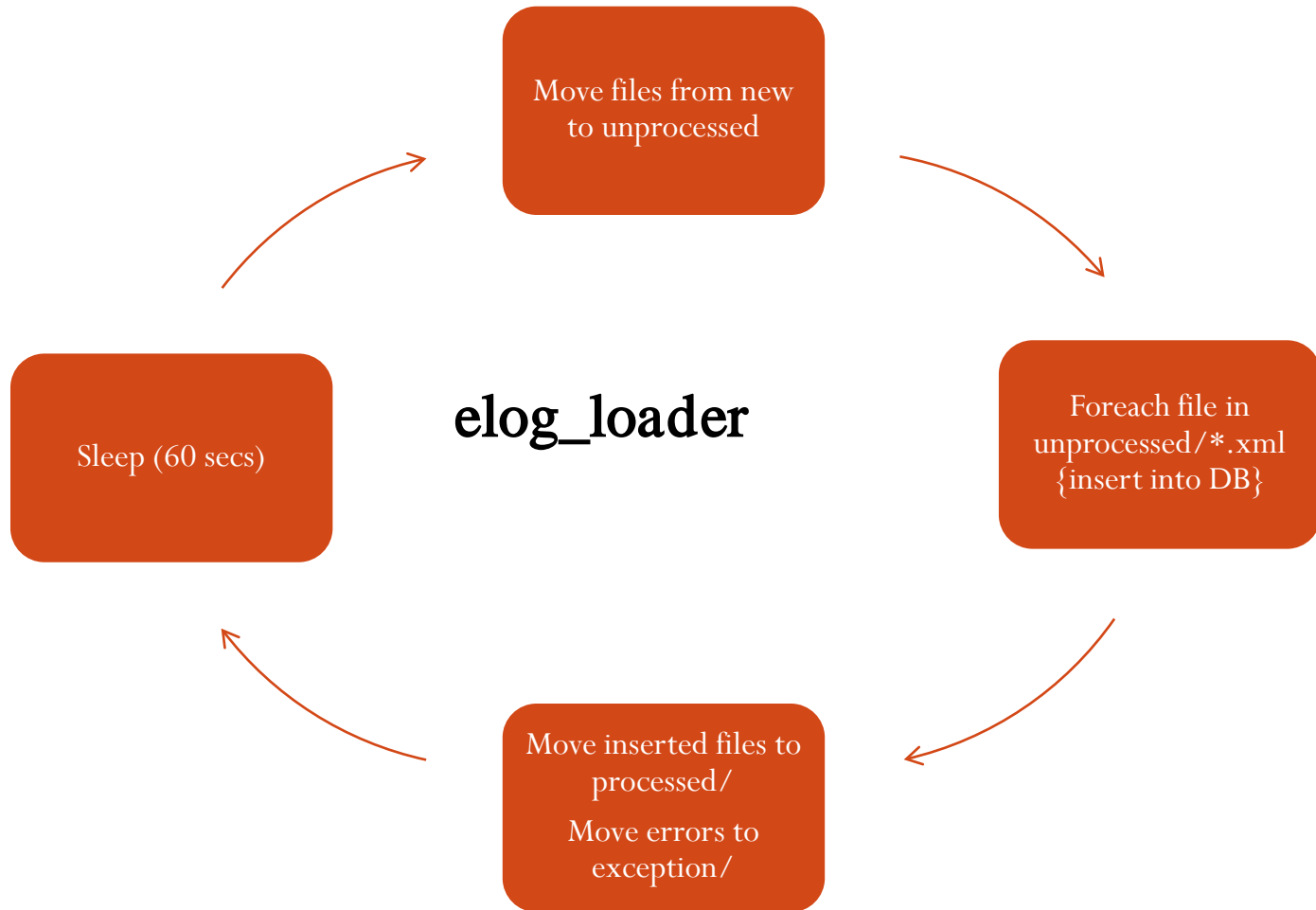
HTTP Submission

- In cases where programs need to submit an entry immediately, the xml file can be submitted for immediate processing via HTTP PUT
 - Client lib will do equivalent of `/usr/bin/curl -s -T myfile.xml http://logbooks.jlab.org/incoming/myfile.xml`
 - If successful, the lognumber assigned by the database will be returned as reference/confirmation.

```
<Response stat="ok">  
<msg>Entry saved.</msg>  
<lognumber>2047200</lognumber>  
<url>https://logbooks.jlab.org/entry/2047200</url>  
</Response>
```

- Direct Submission will require submitting authentication details of the author and use SSL.

The elog_loader queue will also be available



Feature Request

- Must be viewable on web outside firewall
 - The logbooks.jlab.org server is on the externally accessible web server network. When it is ready for production, the Computer Center will enable outside access to it.
- Onsite and offsite links to an entry must be the same (i.e. no "inside the firewall version" and an "outside the firewall version").
 - Every entry will have a persistent path of the format:
 - <https://logbooks.jlab.org/entry/1234567>

Feature Request

- Must be able to upload attachments, photos, screen dumps, PDF, Word, etc
 - This is a native capability of the drupal CMS.
 - This is supported by the XML-based API

Feature Request

- **Must be able to add entries from web, scripts and email.**
 - Drupal provides robust web forms
 - The XML API and utilities make it easy for scripts
 - Email entries can be handled by a procmail script that accepts the incoming message, parses it and writes it to logbook via the API. The Computer Center has scripts that do exactly this for CCLOG and can be adapted.
 - Entries that arrive via email are inherently unauthenticated -- concerns about forged entries?
 - Can email to logbooks arrive from outside jlab? If so, what about spam going into the logbook?

Feature Request

- **Must have web-based search facility**
 - Drupal has a built in web-based search (has limits)
 - For larger sites (>10,000 pages, i.e. the logbooks) Drupal community recommends replacing it with the open-source Solr/Lucene (see <http://drupal.org/project/apachesolr>)

Feature Request

- Should allow inline annotations of locked entries. So, add text and attachments but not modify those already there.
- Cc

The screenshot shows a web form for adding a comment. A red circle highlights the 'Add new comment' button. A red line connects this button to a larger red circle that encompasses the 'Picture' section of the form. The 'Picture' section includes a file upload area with a preview of a waveform image and a file named 'qweakx200.png (20.5 KB)'. Below this is a section for adding a new file with 'Browse' and 'Upload' buttons. The form also includes a 'Text format' dropdown set to 'Plain text' and a 'Comment' text area.

Add new comment

Your name
A_1strator

Comment *

Text format: Plain text [More information about text formats ?](#)

- No HTML tags allowed.
- Web page addresses and e-mail addresses turn into links automatically.
- Lines and paragraphs break automatically.

Picture

Include an image attachment with comment

File information

Operations

Remove

Add a new file

Browse Upload

Files must be less than 96 MB.
Allowed file types: png gif jpg jpeg.

Feature Request

- Should have history mechanism to circumvent concerns over entry modification. Revision control would be strongly preferred if modification is allowed.
- Version control is provided. When updating an entry, a new version is created with a comment describing the purpose of the revision.
- Must decide on the rules. Admins? Creator? 24 hour window?

Revision information

New revision

Revision log message

Provide an explanation of the changes you are making. This will help other authors understand your motivations.

Save Preview

Feature Request

- Must be able inline attachments with text. Example “blog” style entries where graphics are embedded inline in the text of an entry.
- Suggest something like:
 - Replace token ‘[[figure 1]]’ with ``

Feature Request

- **Must be able to lock entries.**
 - Drupal has built-in support for enabling/disabling comments. Need clarification of who/when/why
 - May have to write custom code hooks to prevent further revisions being made to the core of the document. Perhaps based on its age? Need definition of the rules for this “locking”?

Feature Request

- Log entries must have a set of mandatory fields such as: Run number, Run Type, date/time, operator name and email, entry type, topic, subject, text.
- All entry types will be required to have title, author, timestamp, and a logbook association to allow them to be displayed consistently in listings.
- Additional entry types may be defined that extend these basic fields. A Run Logentry may add Run Number, Run Type, etc. as additional required fields.
- Prefer to use Tags where possible to avoid excessive number of entry types to manage

Feature Request

- Log system must hard-code as few fields as possible, the rest should be optional.
 - The required fields a basic log entry are minimal, just a title, a logbook selection and an author really. (The system will provide datetime if necessary)
 - Other types of entries may specify additional requirements.
 - Vocabularies of tags provided to further flag log entries with various optional attributes

Feature Request

- The administrator should be able to customize the logbook webpage header and navigation bar. For example to point to the running experiment's wiki, homepage, shift signup, add hall specific branding (logo), etc.
 - Sidebar links to useful resources are logbook-specific. A link may appear on many logbook pages or just one.
 - Different blocks and sidebars can be made available for different logbook home pages.
 - Displaying a different logo depending on the url being viewed is in the realm of possibility

Feature Request

- The system must be flexible enough that future requirements can be met without modification of existing log entries.
 - *Almost an impossible request since those “future requirements” could be anything.*
 - But flexibility exists to
 - Extend current base logentry definition without invalidating existing entries.
 - Add new tags to classify entries in new ways.
 - Change the look and feel of the web pages.

Feature Request

- Should be database independent implementation, i.e. don't lock to database vendor.
 - Drupal runs on Mysql, Postgres, and even theoretically Oracle.
 - The XML format for making log entries is database agnostic.
 - The XML files are inserted using the drupal API and therefore portable to any supported database.

Feature Request

- Should have major threads based on type, topic, subtopic.
 - The same entries can/will be viewable as threaded in various ways
 - By fields (e.g. “type”)
 - By tags (e.g. “topic, subtopic”)
 - By explicit reference to other entries
 - By comments
 - By author
 - By date
 - By search
 - By faceted search

Feature Request

- Should be able to reference other entries
- User should be able to create minor thread between arbitrary entries.
- There are two ways to interpret this.
 1. Inline references such as:
 - *[[1234567]][See this entry]] becomes `See this entry` upon rendering*
 2. Explicit database references
 - Must specify a lognumber to reference when creating the entry.
 - Has the advantage of also displaying as “back links” from the referenced entry.

[Show row weights](#)

LOG REFERENCE:

+

Used to make references to an existing log entry.

Add another item

Feature Request

- Should have templates for text field based on entry type (e.g. shift summary, problem report, etc.)
 - A javascript widget can be provided to fetch a template from a url and insert it into the textarea.
 - The wysiwyg editors may provide the ability to load templates too.

Feature Request

- Should allow auto-email or page to system experts for special entry types, unresolved problems, etc.
- The Drupal triggers and actions modules can be used for this.

[Home](#) » [Administration](#) » [Structure](#) » [Triggers](#)

Triggers are events on your site, such as new content being added or a user logging in. The Trigger module associates these triggers with actions (functional tasks), such as unpublishing content containing certain keywords or e-mailing an administrator. The [Actions settings page](#) contains a list of existing actions and provides the ability to create and configure advanced actions (actions requiring configuration, such as an e-mail address or a list of banned words).

There is a tab on this page for each module that defines triggers. On this tab you can assign actions to run when triggers from the [Node module](#) happen.

TRIGGER: WHEN EITHER SAVING NEW CONTENT OR UPDATING EXISTING CONTENT

Choose an action ▼

Assign

TRIGGER: AFTER SAVING NEW CONTENT

Choose an action ▼

Assign

TRIGGER: AFTER SAVING UPDATED CONTENT

Choose an action ▼

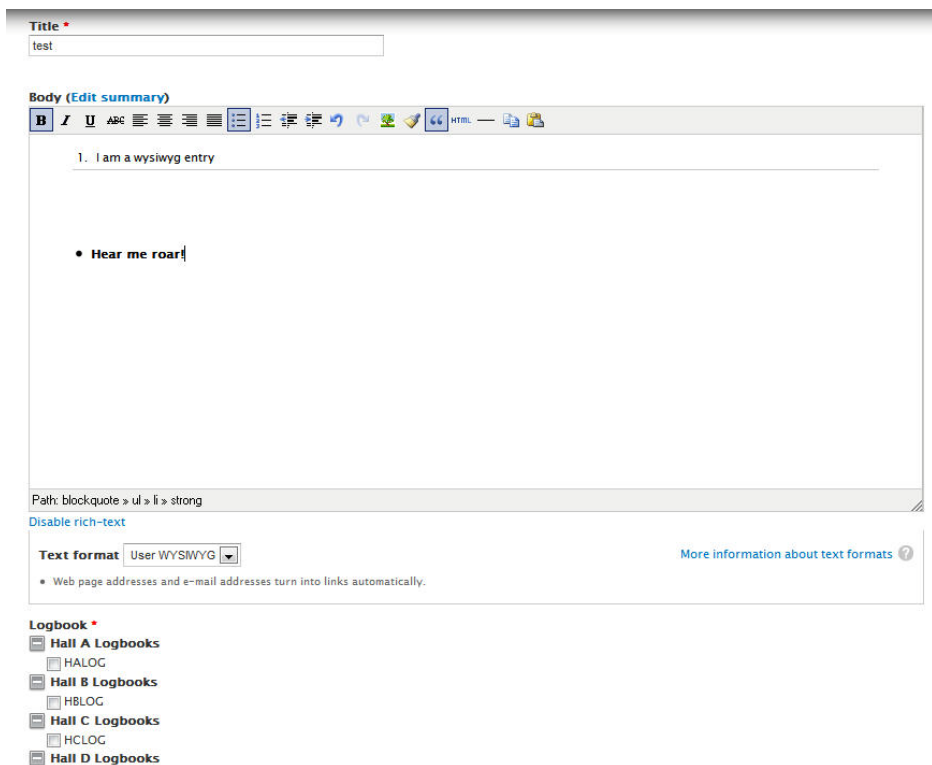
Assign

Feature Request

- Users should be able to subscribe to a logbook with email updates (RSS would be useful).
- The triggers and actions module could be used to allow users to receive email notification of new postings.
- Every logbook has an RSS feed:
 - Example: <https://logbooks.jlab.org/book/elog/rss>

Feature Request

- Would like a wiki-like or WYSIWYG editor for the main text field.
- Several are widely used as plugins for Drupal. For testing purposes, I have been experimenting with tinyMCE



Feature Request

- Would like the ability to import of old logbook into new system.
 - Possible:
 1. Read old entry
 2. Save content to XML in elog queue
 3. Rinse and repeat
 - But old logentries may be difficult to parse*:
 1. Invalid/irregular author names
 2. Inconsistent (non-semantic) HTML markup
- *Based on experience writing script to slurp sample data from HALOG and HCLOG

Feature Request

- Must be able to handle (or at least scale to) several million entries per Hall. Current logbook Hall C is at 250k entries, Hall A is $> 360k$; We expect the entry volume will rise as it becomes easier to automate or script entries to the logbook .
- Drupal is used by web sites that serve millions of visitors/day and it is capable of scaling, however it will be incumbent upon Computer Center to provide adequate hardware as the load grows. This may entail a standalone database with enough RAM to keep key tables in memory and/or a dedicated search server.
- The application will be optimized to provide the best performance to access the most recent data, usually via caching mechanisms.

Feature Request

- Would be useful if the logbook was be viewable and functional on "smart devices" (phones, tablets). This should be a relatively low bar since modern iPhone/Android browsers are quite feature complete. Just make sure the back end delivers decent HTML that does not have hard-coded assumptions about device resolution, display size, etc.
- Option 1. Use an RSS reader on the mobile device
- Option 2. Provide an optimized mobile interface using a tool such as the Drupal jquerymobile project.

Time-frame

- Phase 1 – By Sept 30, 2012
 - Client Libraries 1.0 installed for CUE, ACE
 - Basic Logentry type implemented
- Phase 2 – By October 31, 2012
 - Begin Parallel posting of OPS entries
 - Application developers may begin making test entries
- Phase 3 – By November 30, 2012
 - Search integration (Apache SOLR server)
- Phase 4 – Begin January 01, 2013
 - Wipe out test entries, Import data from old logbook systems
 - Parallel operation with existing logbooks
 - Tune performance, add features and entry types
- Fully Operational – October 1, 2013
 - Cutover and decommission old logbook systems

API – barebones usage

- Create XML file such as the one at right

```
<Logentry>
  <title>My first Logentry</title>
  <Logbooks>
    <logbook>TLOG</logbook>
  </Logbooks>
  <Entrymakers>
    <Entrymaker>
      <username>theo</username>
    </Entrymaker>
  </Entrymakers>
</Logentry>
```

- Validate it

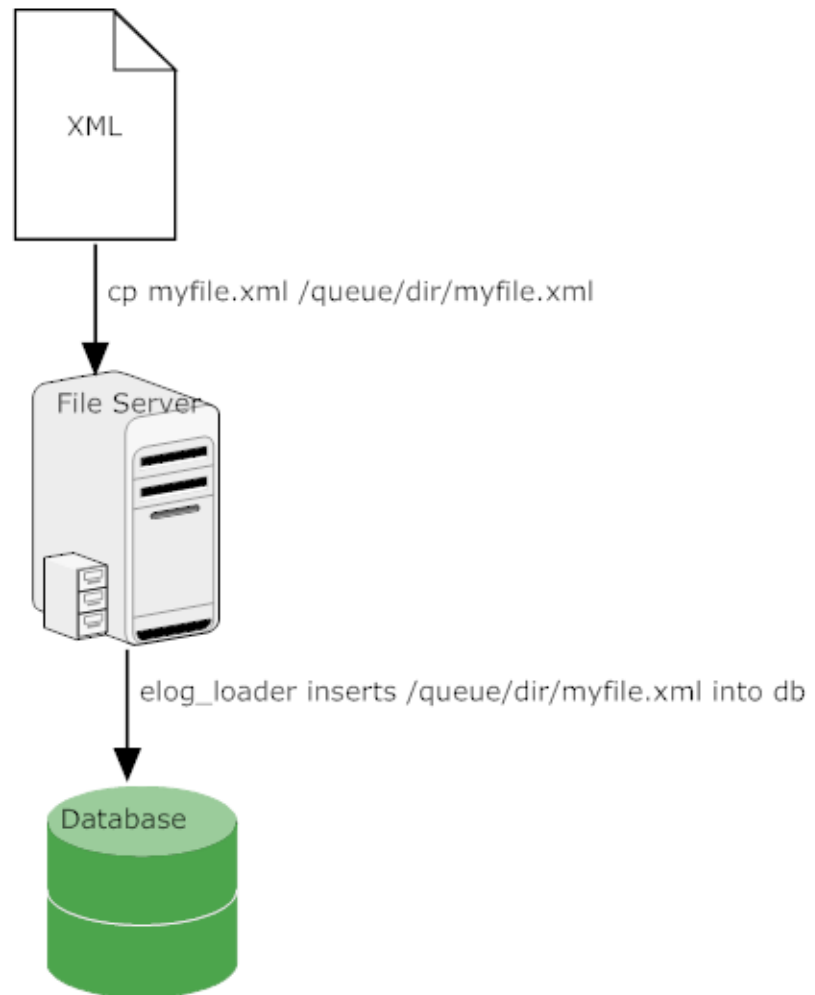
```
xmllint --schema http://logbooks.jlab.org/schema/Logentry.xsd mytest.xml
```

- Place it in the logbook input queue

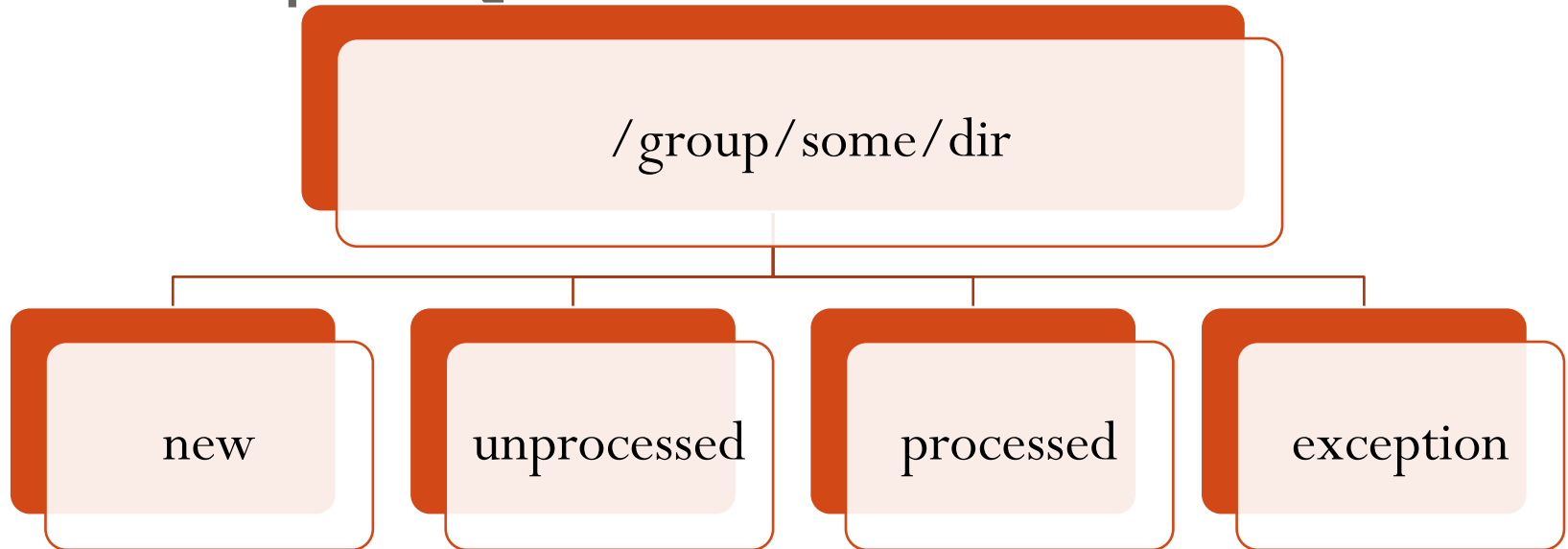
```
cp mytest.xml /group/div-acc/www/logbooks/queue/new/mytest.xml
```

Making Log Entries

- Entries placed in a filesystem queue will be processed and inserted into the database within 5 minutes, typically within 1 minute



The Input Queue



- New – Writable drop-box directory for XML files
- Unprocessed – Files moved here to be processed
- Processed – Files moved here after successful DB insertion
- Exception – Files that cannot be inserted moved here

Changes from OPS perspective

- Ability to comment on entries, add attachments with comments.
- Ability to revise an entry (old version remains in database)
 - Reorder figures, change captions
 - Add, remove tags
- New API for making entries
 - During transition, elog_loader will convert legacy XML files into the new schema and resubmit them
- New view modes
 - Headlines
 - Archive
 - Tags