

ELOG from JLAB as implemented at SLAC Q&A sheet for Electronic Logbook Architecture

1. In what language(s) is the logbook written?

database is Oracle
user interfaces use PHP with HTML & JavaScript
PERL for xml file processing/creation & getting form info into the db
tkl used for one interface (view and entry – interesting feature: can make entries while oracle is down which are swept in when oracle comes back)

2. What database is used?

Oracle

3. Can the database be changed or is it vendor specific?

Switching to mySQL might be an option but is a lot of work

4. What is stored in the database?

A blurb here about how data gets to database:
from web page in php+perl script, the data is entered (log book, title, body text, username, etc) and then stored in an xml file that has the info between appropriate tags. Then a periodic perl script (every 2-3 min) comes through and eats of the new xml files and puts the data they contain into the right fields in the database. How the tk interface and auto-logging facilities work is to create an xml file in the right directory so it gets swept up with the rest.

What is actually stored?

The pieces of text data from the web form or xml file and pointers to the graphics attachments.

5. In what format are the entries stored?

text in database
pointers to graphics PNG files – all graphics attachments are converted to png (via the print queue mechanism presumably)
pdf, xls, doc are stored in their original format (Janice's guess on this last point)

6. What security mechanism is used?

A list of allowed users is maintained and log entries require AFS authentication
There are only three Oracle users: one read-only, one read/write, and one just for the software developer who is all-powerful.

7. How is it deployed?
 - a. As source code
 - b. As a (web) application
 - c. As a service
 - d. from a webpage (download)
 - e. other

user side is web application & Tk/xwin
Get source code by calling Theo at JLAB

8. How mature is it? How many years has it been in use?

2.5 yrs since first install at SLAC (when NLCTA started using it)
6 months for Main Control
we don't know for how long at JLAB

9. As an indication of stability, how many releases were made during the last year, and when was the last release?

All the releases in the past year have been adding new features, no bug fixes.
6 releases in past year. Last release was about a month ago.

10. Is it configurable and how does one configure the logbook?

changing the web & tk interfaces are easy
Changing the oracle db structure isn't that hard either, though requires changes to perl scripts too.

11. Does it have a backup utility for the entries?

Oracle db backed up regularly. db gets about 4 patches/yr. Each patch takes the system down from 0.25-1.5 hours.
Also db is on a different machine than the web server (which has the web interface code on it) They mentioned this. I'm not sure of its relevance.

12. Is it open source?

freely distributed through owner

13. Is there a license, is there a cost?

free

14. Is it platform independent?

yes for the source code (perl, tk, php, &c), but the configuration is platform

specific

15. How is content indexed and searched?

From the Oracle db Admin, yes, the text is indexed. The list of indexed fields are at the end of this document (long list).

16. Does it have programmatic access?

See note in question 4 – any program can create an appropriately formatted xml file that a perl "elog-loader" will find and put the data into the db.

I don't know of any programs that make direct entry into the db besides the perl script.

17. SWE opinions about maintainability, design, annoyances, how could it be improved?

CONS:

-install was a pain in the butt. Took two SWE 3-4 weeks followed by one solid week of the owner at SLAC to get it going. Hardest part was setting up the print queues. 2nd hardest part was getting the non-standard perl module amassed. They estimate two weeks for someone who knew what they were doing to get it installed.

- perl code is a bit messed up and have to change the same thing in multiple files
- source code has minimal comments, headers, general help. Also sounded like the installation guide was less helpful than needed.

PROS:

- once installed very easy to maintain, very robust – no bug fixes in the past yr of its 2.5 yr tenure at SLAC
- very easy to modify user interface
- once you have the 6-step procedure, adding a new logbook is super easy and takes < 1hr

18. Their answers to my miscellaneous questions:

We're now up to 15 logbooks, 2 of these get >100 entries/day. Now have > 90000 entries.

They also pointed out that they have a separate Oracle database admin person. He has dealt with a few odd problems with oracle since the install at SLAC.

From the Oracle db Admin:
Here are the indexed fields

TABLE_NAME	COLUMN_NAME
AREAS	AREA_ID
AREAS	SUBAREA_OF
ATTACHMENTS	ATTACHMENT_ID
ATTACHMENTS	ELOG_ID
BINARY_ATTACHMENTS	ATTACHMENT_ID
COMPONENTS	COMPONENT_ID
COMPONENTS	PROBLEM_CATEGORY_ID
COMPONENTS	SUBCOMPONENT_OF
COMPONENT_SUPPORT	COMPONENT_ID
COMPONENT_SUPPORT	GROUP_ID
DOWNTIME	ELOG_ID
DOWNTIME	TIME_RESTORED
DR\$ELOG_TEXT_INDEX\$I	TOKEN_COUNT
DR\$ELOG_TEXT_INDEX\$I	TOKEN_FIRST
DR\$ELOG_TEXT_INDEX\$I	TOKEN_LAST
DR\$ELOG_TEXT_INDEX\$I	TOKEN_TEXT
DR\$ELOG_TEXT_INDEX\$I	TOKEN_TYPE
DR\$ELOG_TEXT_INDEX\$K	TEXTKEY
DR\$ELOG_TEXT_INDEX\$N	NLT_DOCID
DR\$ELOG_TITLE_INDEX\$I	TOKEN_COUNT
DR\$ELOG_TITLE_INDEX\$I	TOKEN_FIRST
DR\$ELOG_TITLE_INDEX\$I	TOKEN_LAST
DR\$ELOG_TITLE_INDEX\$I	TOKEN_TEXT
DR\$ELOG_TITLE_INDEX\$I	TOKEN_TYPE
DR\$ELOG_TITLE_INDEX\$K	TEXTKEY
DR\$ELOG_TITLE_INDEX\$N	NLT_DOCID
ELOG	ELOG_ID
ELOG	PROGRAM_ID
ELOG	TEXT
ELOG	TITLE
ELOG_ERRORS	ELOG_ERROR_ID
ELOG_ERRORS	ELOG_ID
ELOG_NEWTS	ELOG_ID
ELOG_NEWTS	NEWTS_ID
ELOG_NOTIFICATIONS	ELOG_ID
ELOG_NOTIFICATIONS	EMAIL_ADDR
ELOG_SEGMENTS	ELOG_ID
ELOG_SEGMENTS	SEGMENT_ID
ELOG_USERS	USERNAME
ELOG_USERS	USER_ID
ENTRY_LOGBOOKS	ENTRY_ID
ENTRY_LOGBOOKS	LOGBOOK_ID
ENTRY_TYPES	ENTRY_TYPE_ID
GROUPS	GROUP_ID
GROUPS	SUBGROUP_OF
GROUP_LEADERS	GROUP_ID
GROUP_LEADERS	USER_ID
GROUP_MEMBERS	GROUP_ID
GROUP_MEMBERS	USER_ID
LOGBOOKS	LOGBOOK_ID
LOGBOOK_USERS	LOGBOOK_ID
LOGBOOK_USERS	USER_ID
LOGBOOK_WEBLINKS	LOGBOOK_ID
LOGBOOK_WEBLINKS	WEBLINK_ID
LOGGING_USERS	ELOG_ID

LOGGING_USERS	USER_ID
LOG_REFERENCES	ENTRY_ID
LOG_REFERENCES	REFERENCES
LOG_SUPERCEDE	ENTRY_ID
LOG_SUPERCEDE	SUPERCEDES
NEWTS	ACTION_BY
NEWTS	ANCESTOR_OF
NEWTS	AREA_ID
NEWTS	ASSIGNEE
NEWTS	COMPONENT_ID
NEWTS	NEWTS_ID
NEWTS	PROBLEM_ID
NEWTS	TIMESTAMP
NEWTS_USERS	USER_ID
NLCTA_CONDITIONS	ABBREVIATION
NLCTA_CONDITIONS	CONDITION_ID
NLCTA_CONDITIONS	NAME
NLCTA_CONDITIONS_TO_ITEMS	CONDITION_ID
NLCTA_CONDITIONS_TO_ITEMS	CONDITION_ITEM_ID
NLCTA_CONDITION_ITEMS	ABBREVIATION
NLCTA_CONDITION_ITEMS	CONDITION_ITEM_ID
NLCTA_CONDITION_ITEMS	NAME
NLCTA_ITEMS_TO_ENTRIES	CONDITION_ITEM_ID
NLCTA_ITEMS_TO_ENTRIES	ELOG_ID
PROBLEMS	PROBLEM_CATEGORY_ID
PROBLEMS	PROBLEM_ID
PROBLEM_CATEGORIES	PROBLEM_CATEGORY_ID
PROGRAMS	PROGRAM_ID
SEGMENTS	ABBREVIATION
SEGMENTS	NAME
SEGMENTS	SEGMENT_ID
SEGMENT_LOGBOOKS	LOGBOOK_ID
SEGMENT_LOGBOOKS	SEGMENT_ID
TEXT_ATTACHMENTS	ATTACHMENT_ID
WEBLINKS	WEBLINK_ID