

# COMPACT ELECTRONIC LOGBOOK SYSTEM

L. Wang<sup>†</sup>, Y. L. Zhang, P. Chu, X. Wu, F. Q. Guo, Y. C. He, P. Zhu,  
M. T. Kang, Z. Zhao, J. Liu, D.P. Jin

Institute of High Energy Physics, Chinese Academy of Sciences, Beijing, China

## Abstract

Compact Electronic Logbook System (Clog) is designed to record the events in an organized way during operation and maintenance of an accelerator facility. Clog supports functionalities such as log submission, attachment upload, easy to retrieve logged messages, RESTful API and so on, which aims to be compact enough for anyone to conveniently deploy it and anyone familiar with Java EE (Enterprise Edition) technology can easily customize the functionalities. After the development is completed, Clog can be used in accelerator facilities such as BEPC-II (Beijing Electron/Positron Collider Upgrade) and HEPS (High Energy Photon Source). This paper presents the design, implementation and development status of Clog.

## INTRODUCTION

In China, several accelerator facilities are being constructed right now or will be constructed in the near future, so there are strong needs for an electronic logbook system adopting new technologies and architecture to meet the requirements of modern accelerator controls and operations.

Clog is a web-based electronic logbook system, which is developed using Java EE [1] framework, PrimeFaces [2] component library, MySQL relational database and RESTful web service technologies. It forks from the source code of Olog [3] and adopts the display style of Elog [4]. Clog provides log submission, log query, log update and log deletion functionalities via the PrimeFaces UI (User Interface), it also provides RESTful API to communicate with CS-Studio and mobile web UI using HTTP (Hypertext Transfer Protocol) protocol.

## SYSTEM OVERVIEW

Clog is developed using Java EE framework and deployed as a web application in GlassFish container. It implements a PrimeFaces UI to be accessed directly by any popular Web browsers and RESTful API to interface with CS-Studio and mobile web UI. The system architecture is shown in Fig. 1.

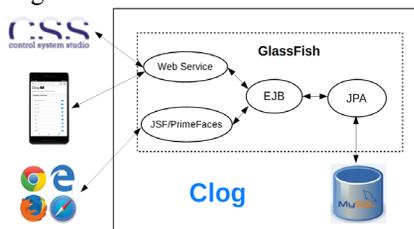


Figure 1: Architecture of Clog.

Clog is developed based on the source code of Olog and has made the following modifications to simplify the implementation:

- Replaced the HTML/Bootstrap/JQuery web UI with JSF/PrimeFaces in order to simplify the UI development, deployment and maintenance.
- Removed BitemporalLog data type, which is very complicated and unnecessary for a logbook system.
- Replaced the Jackrabbit attachment storage mechanism with plain file system, which is straightforward and will facilitate data backup in the future.
- Replaced the JPA (Java Persistence API) Criteria API with JPQL (Java Persistence Query Language) in order to improve the readability of the source code and easier to program database queries.
- Added “subject” field into log entry, which is useful to summarize the log entry.
- Removed “property” table from the database, which is not commonly used.
- Removed Flyway database migration tool, which is not necessary for a logbook system.

## BACKEND IMPLEMENTATION

### Database Design

Clog has five tables in the database as shown in Fig. 2, the description for the tables are as follows:

- The “entry” table corresponds to log entries.
- The “log” table corresponds to different log versions of a log entry.
- The “logbook” table is shared by both logbook and tag, and they are distinguished by the “is\_tag” field.
- The “sysuser” table saves user information.
- The mapping between entry and log are one-to-many, which means one entry includes multiple logs and one log belongs to one entry.
- The mapping between log and logbook is many-to-many, which means one log can belong to multiple logbooks and one logbook can include multiple logs.

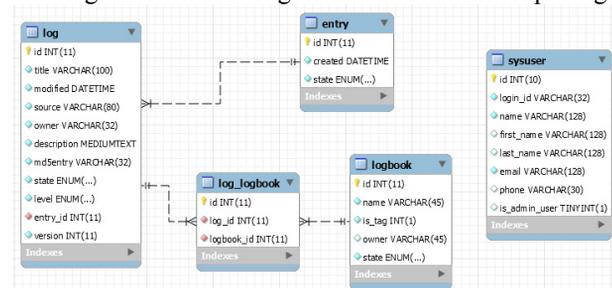


Figure 2: Clog database schema.

<sup>†</sup> wanglin@ihep.ac.cn

Content from this work may be used under the terms of the CC BY 3.0 licence (© 2019). Any distribution of this work must maintain attribution to the author(s), title of the work, publisher, and DOI.

## Core Logic Implementation

The core logic of Clog which is the most important part in backend provides the following functions:

- Log submission: Submit a log entry with logbook, tag, if any, level, subject and description. In addition, upload attached files if any.
- Log query: Search a group of log entries with specified keywords combination.
- Logbook management: Add or remove logbooks.
- Tag management: Add or remove tags.
- User management: Add or remove user information.

The core logic is implemented as multiple stateless EJBs (Enterprise Java Bean), which implement CRUD (Create, Read, Update and Delete) operations to the database tables via JPA entity beans and can be invoked by JSF (JavaServer Faces) backing beans and RESTful web service. The EJBs implemented are listed below,

- logEJB
- logbookEJB
- tagEJB
- attachmentEJB
- userEJB

## Attachment Storage

Attachments are stored in the file system, the hierarchy is shown in Fig. 3. The “Root” directory is the Clog attachment repository; the “Attachment” directory stores the original attachment files; the “Thumbnail” directory stores the corresponding thumbnails for image files which are generated during the attachment submission; the “Year”, “Month” and “Day” directories are the timestamps of the log entries; and the “Log ID” directory is the generated ID when a log entry is created.

When a log entry is submitted with attachments, the attached files will be stored in the directory hierarchy comprised of year, month, day and logId. Meanwhile, for each image file, an additional thumbnail with size 80\*80 will be stored in the same hierarchy in “Thumbnail” directory.

Similarly, when a log entry with attachments is retrieved, the attached files can be located with the log’s timestamp and ID.

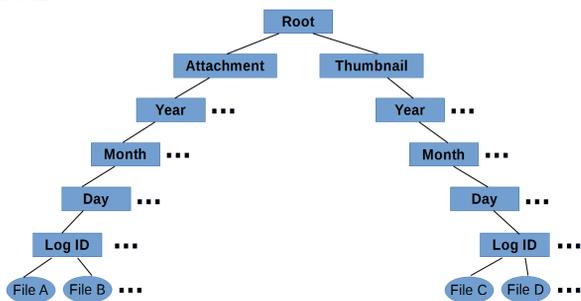


Figure 3: Attachment directory hierarchy.

## FRONTEND IMPLEMENTATION

### Application Management Page

The logbook, tag and user information can be created and edited by Clog administrators.

The user management page is shown in Fig 4, and the logbook and tag management pages are similar.

User ID	Name	Email	Admin
admin1	admin1	admin1@hep.ac.cn	✓
admin2	admin2	admin2@hep.ac.cn	✓
user1	user1	user1@hep.ac.cn	✗
user2	user2	user2@hep.ac.cn	✗
user3	user3	user3@hep.ac.cn	✗
wangjia	wangjia	wangjia@hep.ac.cn	✓

Figure 4: User management page.

### Home Page

The Clog home page, as shown in Fig. 5, lists all the available logbooks, including the logbook name, number of logs in the logbook as well as the last submission time and the user. The logbook name is clickable to navigate to the log overview page for the logbook.

Name	Entries	Last submitted at	Last submitted by
CCF-Overseas-1	3	2019-08-10 14:10	admin1
Control	62	2019-08-09 09:10	王琳
Control2	13	2019-08-09 09:10	王琳
Control3	6	2019-02-24 10:25	王琳
new	0		
test	9	2019-02-24 10:25	王琳
test 6	2	2019-02-14 14:42	王琳
test 7	0		
test 8	0		
test 9	0		
test4	1	2019-08-07 11:27	王琳
test5	0		
test6	1	2019-02-10 10:23	王琳
test7	3	2019-08-23 11:25	王琳

Figure 5: Clog home page.

### Log Overview Page

The log overview page, as shown in Fig 6, is composed of function buttons, search buttons and a log summary table. The function buttons can create, edit, delete and view logs; the search buttons support keyword query of logs with the combination of start time, end time, tag, level, author, subject and description; the log summary table displays log information line by line which is clickable to navigate to the log detail page.

Index	Date	Author	Author email	Category	Tag	Subject	Text
1	2019-08-09 09:10	王琳	wangjia@hep.ac.cn	Info	log	co-studio client test	Create a log entry from co-studio
2	2019-09-09 09:09	王琳	wangjia@hep.ac.cn	Info	log	co-studio attachment test	test test
3	2019-09-09 09:09	王琳	wangjia@hep.ac.cn	Info	server	co-studio attachment test	test test test
4	2019-08-07 09:08	王琳	wangjia@hep.ac.cn	Info	INFO	co-studio attachment test	test test test
5	2019-08-07 09:07	王琳	wangjia@hep.ac.cn	Info	subject	co-studio attachment test	test test test
6	2019-08-07 09:07	王琳	wangjia@hep.ac.cn	Info	subject	co-studio attachment test	test test test
7	2019-08-07 09:07	王琳	wangjia@hep.ac.cn	Info	subject	co-studio attachment test	test test test
8	2019-08-17 10:10	王琳	wangjia@hep.ac.cn	Info	subject	co-studio attachment test	test test test
9	2019-08-17 10:10	王琳	wangjia@hep.ac.cn	Info	subject	co-studio attachment test	test test test
10	2019-08-17 10:10	王琳	wangjia@hep.ac.cn	Info	subject	co-studio attachment test	test test test
11	2019-08-17 10:10	王琳	wangjia@hep.ac.cn	Info	subject	co-studio attachment test	test test test
12	2019-08-17 10:10	王琳	wangjia@hep.ac.cn	Info	subject	co-studio attachment test	test test test
13	2019-08-17 10:10	王琳	wangjia@hep.ac.cn	Info	subject	co-studio attachment test	test test test
14	2019-08-15 10:05	王琳	wangjia@hep.ac.cn	Info	subject	co-studio attachment test	test test test
15	2019-08-14 10:00	王琳	wangjia@hep.ac.cn	Info	subject	co-studio attachment test	test test test
16	2019-08-14 09:25	王琳	wangjia@hep.ac.cn	Info	subject	co-studio attachment test	test test test
17	2019-08-14 09:10	王琳	wangjia@hep.ac.cn	Info	subject	co-studio attachment test	test test test

Figure 6: Log overview page.

### Log Submission Page

The log submission page, as shown in Fig. 7, is used for creating a new log entry in the current logbook. The log information that needs to be filled out includes tag, level, subject, description and attachment.

New Log Entry

Logbook: Control

Tag: None

Level: Info

Subject: [Text Field]

Attachments: [Choose] [Upload] [Cancel]

[Create] [Cancel]

Figure 7: Log submission page.

## Log Detail Page

The log detail page, as shown in Fig. 8, displays all the information of a log entry, which includes creation time, author name, author email, level, tag, subject, description and attachment overview.

The attachment overview icons are shown as the image thumbnails or other document icons according to the file name extension, and they are clickable to access the original file, which is served by the attachment module in the RESTful web service.



Figure 8: Log detail page.

## WEB SERVICE IMPLEMENTATION

The Clog web service provides RESTful API to communicate with other frontend client such as CS-Studio and mobile web UI via HTTP protocol.

The following data types are supported:

- XML: Implemented with JAXB (Java Architecture for XML Binding) package to communicate with the desktop application client like CS-Studio.
- JSON: Implemented with FasterXML/Jackson package to communicate with web application client like mobile web UI.

The available web service modules and the corresponding URL (Uniform Resource Locator) are listed below:

- Log: /resources/logs
- Logbook: /resources/logbooks
- Tag: /resources/tags
- Attachment: /resources/attachments

## CS-STUDIO INTERFACE

CS-Studio is widely used as OPI (Operator Interface) with EPICS control systems, so it will be very convenient for users to create log entries within CS-Studio.

The SNS fork of CS-Studio [5] is used to interface with Clog. However, in CS-Studio version 4, there is no subject field in the log entry. Therefore, the CS-Studio source code need to be refactored to support the subject field.

The following work has been done to achieve this goal:

- Add subject field support in “edu.msu.nslc.olog.api” plugin in “cs-studio-thirdparty” module.
- Add subject field support in “org.csstudio.logbook” plugin in “cs-studio” module.
- Add subject field support in “org.csstudio.logbook.olog” plugin in “cs-studio” module.
- Add subject label and subject text widgets in the SWT (Standard Widget Toolkit) FormLayout in “org.csstudio.logbook.ui” plugin in “cs-studio” module.

After refactoring, the subject input widget is provided in the log submission UI, the subject text is wrapped in the XML data to be sent to Clog web service, and a new log with subject field can be created via CS-Studio successfully. The CS-Studio log submission UI is shown in Fig. 9.

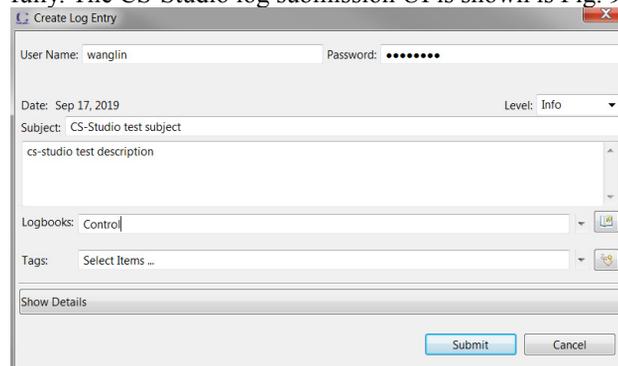


Figure 9: CS-Studio log submission UI.

## MOBILE WEB UI

In addition to the JSF web UI, Clog also provides mobile web UI with read-only permission for users to conveniently view logs on cell phones or other mobile devices. The mobile UI is developed with Vue.js JavaScript framework and Bootstrap responsive CSS library.

Clog mobile home page is shown in Fig. 10 (a), log overview page is shown in Fig. 10 (b), and log detail page is shown in Fig. 10 (c).

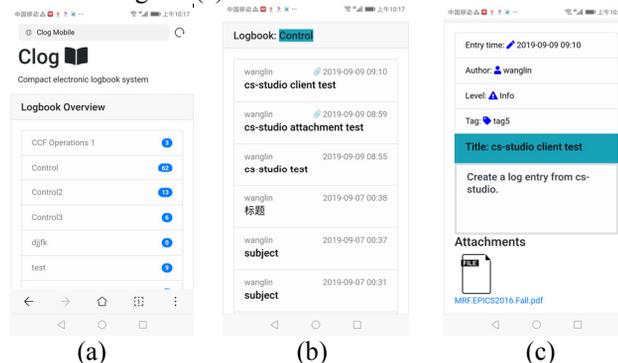


Figure 10: Mobile web page: (a) Home page. (b) Log overview page. (c) Log detail page.

## CONCLUSION

By now, the development of Clog, including database, backend core logic, web service, JSF web UI, mobile web UI and CS-Studio interface, has been finished. It implements the complete functionalities of electronic logbook and provides multiple user interfaces to improve the user experience, which could meet the requirement of new accelerator facilities.

## REFERENCES

- [1] <https://docs.oracle.com/javaee/7/tutorial/>
- [2] <https://www.primefaces.org/>
- [3] <https://github.com/Olog/olog-service>
- [4] <https://elog.psi.ch/elog/>
- [5] <https://github.com/ControlSystemStudio/org.csstudio.sns>