

PaN-data ODI

Deliverable D7.2

D7.2: Mechanisms and tools for information representation and archiving

Grant Agreement Number	RI-283556
Project Title	PaN-data Open Data Infrastructure
Title of Deliverable	Mechanisms and tools for information representation and archiving
Deliverable Number	D7.2
Lead Beneficiary	STFC
Deliverable Dissemination Level	Public
Deliverable Nature	Report
Contractual Delivery Date	01 July 2013 (Month 21)
Actual Delivery Date	21 Oct 2013

The PaN-data ODI project is partly funded by the European Commission under the 7th Framework Programme, Information Society Technologies, Research Infrastructures.

Abstract

Structuring and archiving user experimental notebooks, using a novel web annotation service.

Keyword list

User notebook, experimental logbook

Document approval

Approved for submission to EC by all partners on 21.10.2013

Revision history

Issue	Author(s)	Date	Description
1.0	Jean-François Perrin	31 Aug 2013	Initial release
1.1	Jamie Hall	02 Sep 2013	Software development information
1.2	Franck Festivi	17 Sep 2013	Database structure
1.3	Jean-François Perrin	7 October 2013	Update and formatting
1.4	Jean-François Perrin	21 October 2013	Feedback from partners

Table of Contents

1	Getting the right information	5
1.1	Production of data, and Recording of information.....	5
1.1.1	Sample preparation details.....	5
1.1.2	Experiment logbooks	6
2	Collecting Experimental Logs and instrument user notes	6
2.1	Importance of collecting user notes	6
2.1.1	Sketching the experiment set-up.....	6
2.1.2	Highlighting important decisions.....	7
2.1.3	Recording errors	8
2.1.4	Correlating hardware logs with other information	9
2.2	Basic specifications for Electronic note-keeping	9
2.2.1	Extreme user-friendliness	9
2.2.2	Structured on the instrument control logs.....	9
2.2.3	Data access.....	9
2.2.4	System functions.....	10
2.3	Existing tools	10
2.4	A novel approach	11
2.5	Architecture.....	11
2.5.1	Collecting the instrument control logs	11
2.5.2	Parsing the logs.....	13
2.5.3	Database schema	14
2.5.4	Web application	14
2.5.5	RESTful services.....	15
2.5.6	AngularJs and Bootstrap for frontend development.....	16
2.5.7	Authentication with CAS or SAML	18
2.5.8	Real-time notifications using NodeJs.....	18
2.5.9	User access control	19
2.6	Application workflow	20
2.6.1	Normal operation.....	20
2.6.2	Failure in accessing the central log server	21
2.6.3	Resuming normal operation	22

3	Conclusion and roadmap	23
---	------------------------------	----

1 GETTING THE RIGHT INFORMATION

If we summarise what has already been achieved in terms of information preservation in the photon or neutron facilities within the PaN-data consortium which are already working on the issue, we can state that

- datasets are being archived with their technical / administrative information (sample environment, beam characteristics, etc., / proposal and experimental team ID, date and type of experiment, etc.);
- these datasets and their metadata are searchable;
- they are accessible under a well-defined data protection policy;
- they are identified in a persistent and standard manner using Digital Object Identifiers which can easily be referenced within publications.

The facilities are not all at the same point on this, but the community is clearly well on its way to elaborating a more comprehensive data preservation system. The achievements so far represent significant improvements in terms of information preservation. It is now becoming possible to re-use data long after the end of an experiment. Maximum impact will only be achieved, however, if the set of information available allows other scientists (at least those working in the same scientific field) to fully understand the technical operations performed. In the most common use case, for instance, data is reprocessed to confirm the results cited in a publication; the archive should therefore provide the answers to questions such as when and why was a specific set of the data produced, by whom, and how? And on the question of "how", we are far from advanced, mainly because much of this information is not being recorded electronically by the facilities.

1.1 PRODUCTION OF DATA, AND RECORDING OF INFORMATION

Consider a simplified 3-step experiment workflow:

1. Scientists submit a proposal for an experiment to a facility and the administrative information is recorded.
2. When the scientists receive confirmation that their proposal has been accepted, they prepare the necessary sample(s) in their home laboratories.
3. The experiment is performed at the facility, and the data and technical metadata are archived automatically. The scientists use notebooks to record processes during the experiment; in most cases these are simple informal paper notebooks.

In step 1 and step 3, the facility collects administrative and technical information; the information on sample preparation, and that in the notes taken by scientists, are not often made available and are therefore not recorded/archived by the facility.

1.1.1 Sample preparation details

Samples are almost always prepared in the scientists' labs, under their own responsibility. It is therefore very difficult for a facility to obtain a detailed description of the process. Nevertheless,

the home organizations are now beginning to request the systematic recording of the sample preparation process, in notebook or electronic form, in order to avoid scientific fraud and patent disputes. We can imagine, in a few years' time, linking the sample preparation information to the data archives. This Joint Research Activity (JRA) work package could have aimed at establishing such a link, but, given the constraints on resources and the state of the art with notebooks, we decided to postpone this and focus the available resources on capturing the notes taken by scientists during experiments; this is more likely to achieve a positive impact in a short period of time.

1.1.2 Experiment logbooks

During experiments all researchers take notes; this is extremely important for their subsequent data analysis and understanding. It is common practice to record important events, and to describe what actually happened from a human point of view (including errors); they may add sketches of the position of the sample or detectors or record the initial analyses, ... In most cases these notes are recorded in a personal paper notebook; they are not standardised, nor can they be 'publicly' archived or shared.

2 COLLECTING EXPERIMENTAL LOGS AND INSTRUMENT USER NOTES

2.1 IMPORTANCE OF COLLECTING USER NOTES

Researchers generally run experiments on their own; the notes they take are therefore essential, together with the detailed hardware logs, for understanding how the data has been produced.

We have analyzed notebooks maintained by scientists on different instruments and in different areas of science, from nuclear physics to biology. We reproduce extracts below from typical experimental notebooks, courtesy of ILL scientists Giovanna Fragneto and Paolo Mutti, with a brief description of their usage.

2.1.1 Sketching the experiment set-up

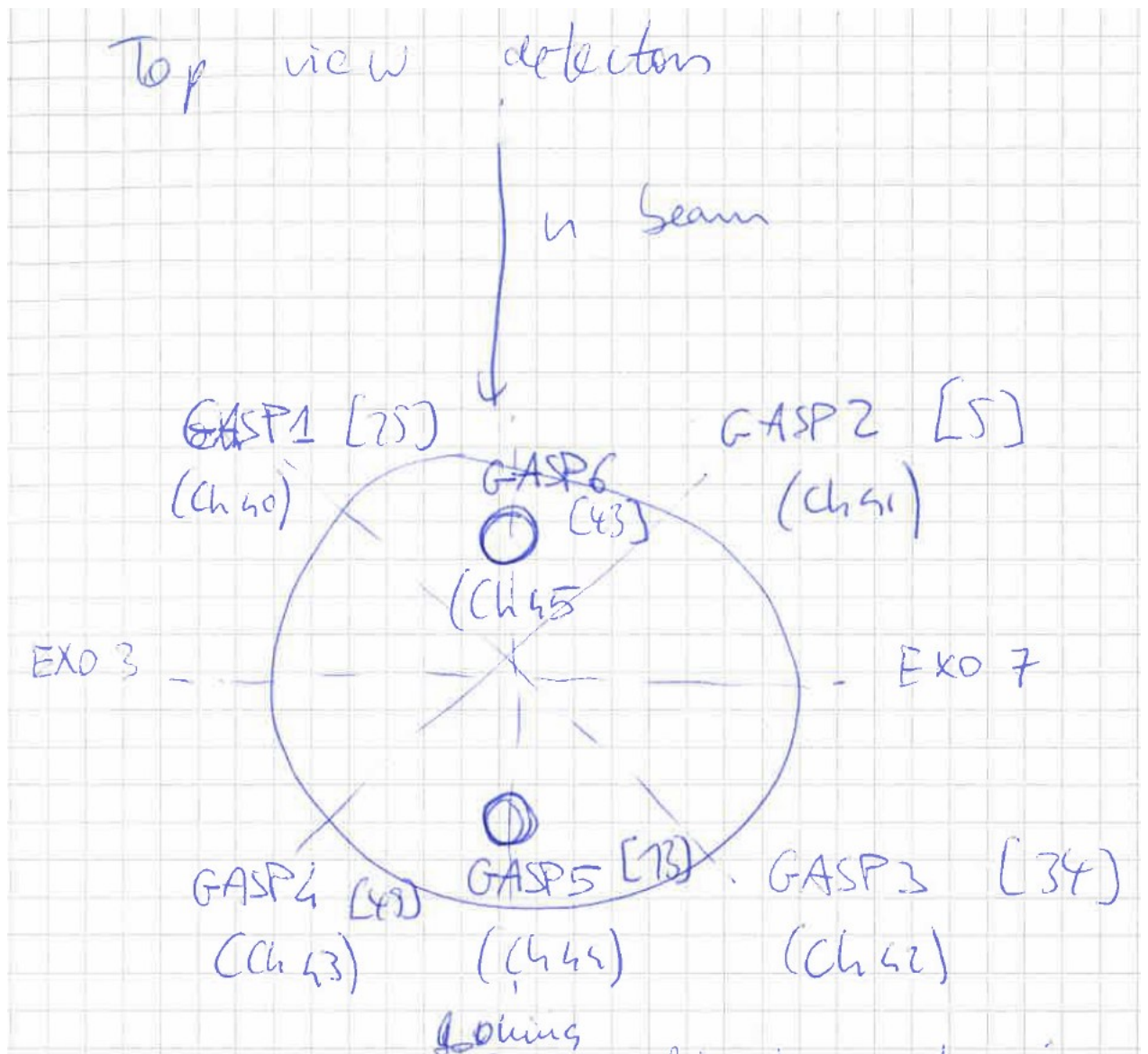


Figure 1: Exill instrument, top view of the detectors

When instrument set-ups are not standard, even rough sketches can provide useful information, on detector position for instance (and the electronic channel in this case) and more generally on the instrument layout.

2.1.2 Highlighting important decisions

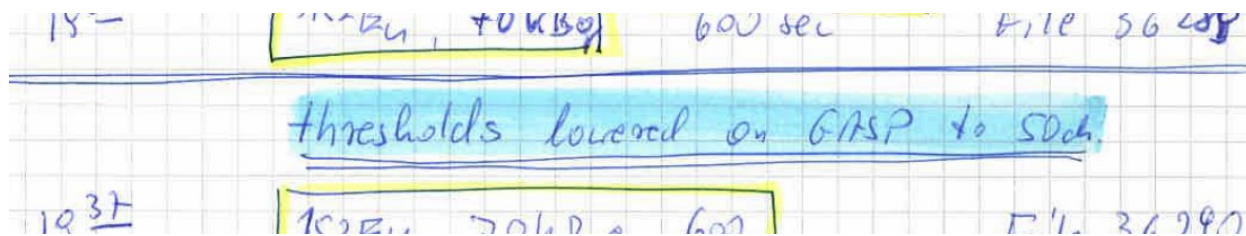


Figure 2 Exill instrument notebook

Any action taken on the instrument could be very important for this particular experiment, but it is often buried in the instrument logs.

2.1.3 Recording errors

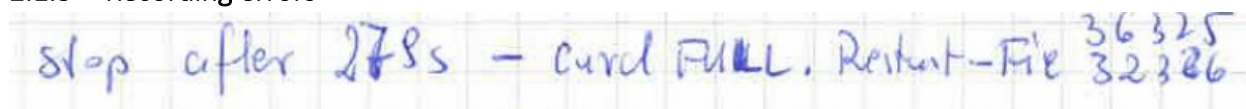


Figure 3 Exill instrument- Card Full

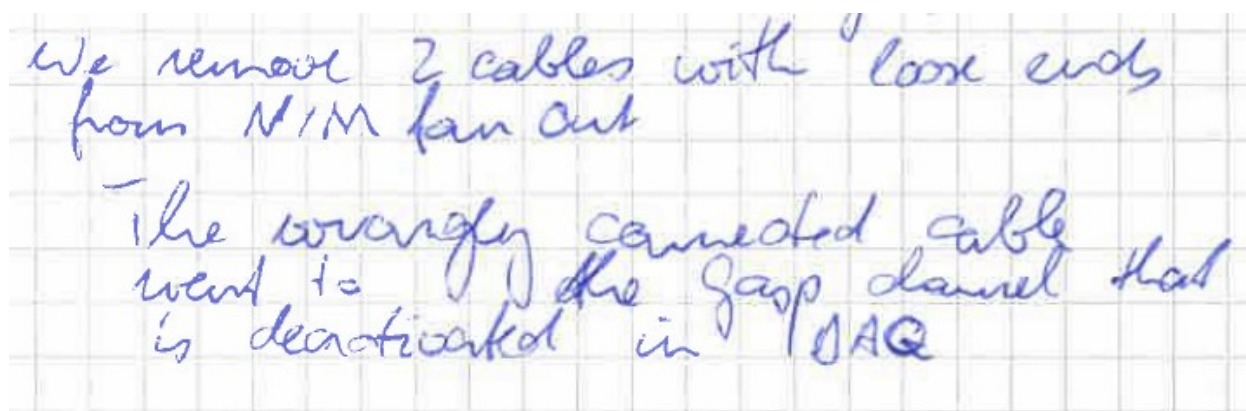


Figure 4 Exill instrument – wrongly connected cable

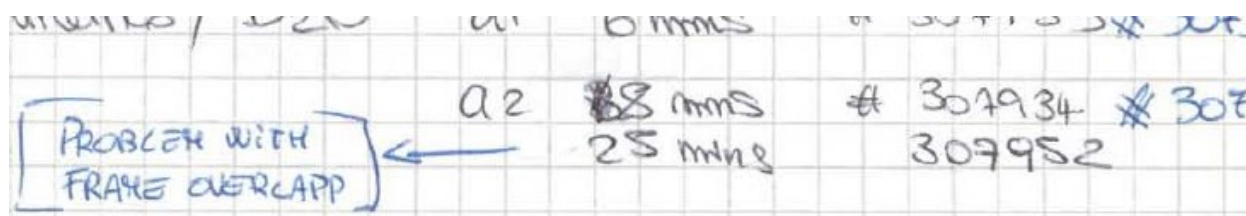


Figure 5 Figaro - frame overlap

A number of incidents could occur during an experiment, which can often only be understood by the operator. Recording this type of information is vital for understanding the data.

2.1.4 Correlating hardware logs with other information

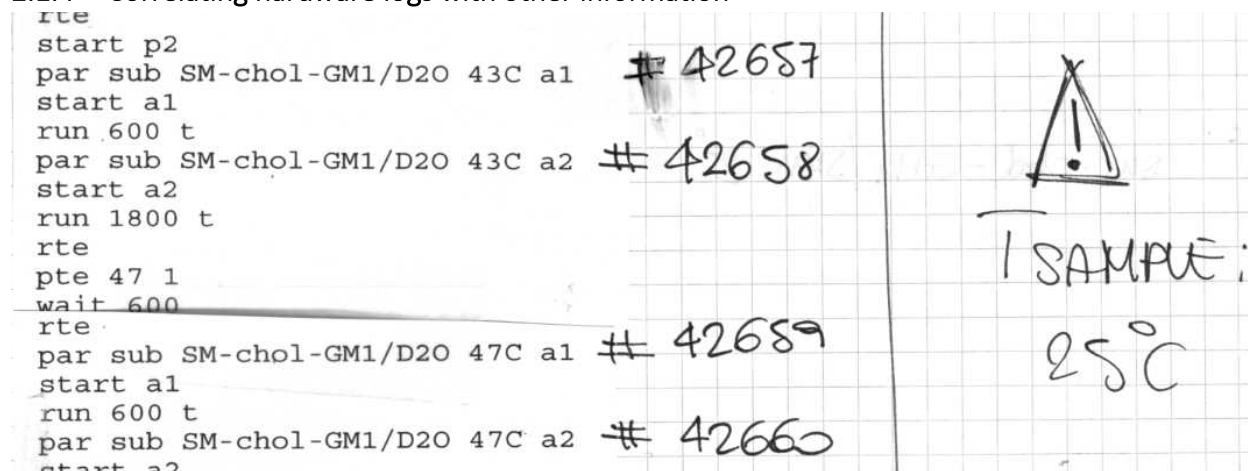


Figure 6 Example from instrument D17

Figure 6 shows the most common practice: users print the output of the instrument control log and add useful information to it by hand (in this case, the scan identifier and sample temperature).

2.2 BASIC SPECIFICATIONS FOR ELECTRONIC NOTE-KEEPING

2.2.1 Extreme user-friendliness

User-friendliness is the key element in the specification. Scientists only come to the facility once or twice a year for a few days at a time; they have neither the time nor the desire to master a complex system. The tool must be simple, self-explanatory and attractive.

2.2.2 Structured on the instrument control logs

In order to structure the scientists' notes and accommodate most common usage (2.1.4), we decided to collect and display the instrument control logs on a web-based interface and to allow experimentalists to add their notes directly to the lines of the log.

2.2.3 Data access

Access to this type of information is governed by each facility's data policy. In the photon and neutron community there is usually a non-disclosure period, ensuring that the experiment team has sufficient time to publish its work. During this period, access to the logs should be limited to the facility staff responsible for scientific support and to the experimentalists themselves. The access authorisation system should integrate easily with the facilities' own authorisation systems (quite often the electronic proposal system).

The interface should be accessible everywhere, from the instrument control computer to any internet-connected computer. It should be a web-based system.

2.2.4 System functions

In order to provide the same level of functionalities, at least, as the paper notebook, the system should allow users to:

- Add textual comments to any line of the logs
- Upload any files that could help in understanding the experiment (images of initial results of analyses, pictures of the instrument, explanatory text, ...)
- Tag individual lines of the log (important, errors, calibration, noise reduction, ...)
- Bookmark lines of the log, enabling them to structure long logs
- Sketch with the mouse and add drawings to uploaded files
- Filter the logs by type of event
- Generate a PDF of the full log, including the annotations.

Functions such as searching, reporting and event correlation should also be introduced. This however is beyond the scope of this task, where we focus on enhancing the understanding of raw data with a view to its long-term preservation. The development of search, report and data mining functions will be carried out outside this project.

2.3 EXISTING TOOLS

More than one hundred software applications already exist in the electronic logbook field (open source/closed source, freeware/commercial, etc.) They all fall into 2 basic categories:

- Applications which structure the information tightly and are often developed for a specific scientific field (biology, chemistry, etc.) or usage (shift operators, maintenance operations, etc.). These tools do not meet our general purposes and user-friendly requirements: even a few days of learning is too long before starting to enter the notes.
- Applications in which the information is not tightly structured; they are often based on blog or wiki technology. These tools are valuable as personal notebooks, but they lack the semantic and access-management functions.

We have paid particular attention to three applications used by PaN-data members:

The "j5" logbook (<http://www.sjsoft.com/>) from St James Software pertains to the first category. It is a feature-rich solution oriented towards the management of industrial processes. This tool could perfectly fit the needs of shift and maintenance operators, but it does not meet some of the requirements of this project (more specifically those of extreme user-friendliness and the specific semantics of scientific research facilities). The European Synchrotron Radiation Facility (ESRF) currently uses j5 for its control room operations^{1 2}.

¹ J5 Logbook - A Commercial e-logbook (<http://accelconf.web.cern.ch/accelconf/pc08/papers/tux04.pdf>)

² Evolution and Status of the Electronic Logbooks at the ESRF
(<http://accelconf.web.cern.ch/accelconf/e08/papers/tupp004.pdf>)

ELOG (<https://midas.psi.ch/elog/>), developed by the Paul Sherrer Institute (PSI), has been tested at the Institut Laue-Langevin (ILL) for some experiments and feedback from users has been positive. ELOG provides a simple interface which could be extended to add missing functionalities, but it is not ideal for structuring the information and access requirements of a large-scale facility. The software primarily provides a logbook for personal or team use; it is not suited for use by instrument users in a large-scale facility, where the logs access are structured by the experimental administration.

Labtrove <http://www.labtrove.org/> is mainly being developed by the University of Southampton and has been tested on a few instruments at ISIS. Labtrove is rich in functionality (sketching is possible, for instance) and probably corresponds in its conception most closely to our needs. It has been designed, however, mainly for laboratory staff, and has not proved its worth as a logbook for large-facility users.

2.4 A NOVEL APPROACH

As none of the existing solutions seems satisfactory as an experimental notebook at facility level, the decision was taken to adopt a novel approach:

- **Note-taking will be structured by the instrument control software.** As experimentalists need to keep a close watch on the logs of their instruments during experiments, we propose providing them with a purpose-built solution allowing rapid annotations to be added to the instrument control logs as they monitor them, in a simple and time-effective manner.
- **The new application will have to be seen as a new and attractive service:** if we really want to collect the notes to enhance our understanding of the data, we need to ensure that the users of the application perceive the benefits of the solution.

2.5 ARCHITECTURE

2.5.1 Collecting the instrument control logs

Logs are generated by the instrument control software, or in some cases directly by the hardware's own software. We need to collect and convey these logs safely to a central archive. It is extremely important not to lose logged events, despite problems with the network or the central archive.

This task of collecting and transporting event logs is not specific to our scenario; it is common in the IT industry and has generated a dozen well-known or emerging specialized products (see, for

example, syslog-ng³, logstash⁴, Kafka⁵, Graylog2⁶, or scribe⁷). In our case we are looking for a solution able to buffer the logs in the event of network failure. Only Fluentd⁸ and Flume⁹ seem to satisfy this requirement easily; they provide a buffer mechanism in the event of the aggregator (or next-hop) failing to handle the message.

Most of these tools only handle ASCII log messages. We shall be generating logs with ASCII text or images; for the images we will encode the binary files in base64.

We have decided on Fluentd for the prototype, mainly for its simplicity of installation; however, Flume could also satisfy, and we could switch to Flume if necessary at any point.

In the case of the ILL, only one input plugin (http) will be used to send logs to Fluentd, but there exist over a hundred input or output plugins for most of the scenarii imaginable (file-based logs, databases, etc.).

In addition to sending the logs to the central collector, we also need to send them to a local websocket, in order to display them locally in the event of a network failure.

The logs are sent from the instrument control software in JSON format. Below you will find two messages typical of this format:

A change of variable:

```
{
  "id": "a9d212ffef3ce6bcd143b67172a1b576",
  "parentid": "af1746031b5c95522c26f63092e0ed15",
  "level": "info",
  "timestamp": 1380536490,
  "event": "simple",
  "type": "variable",
  "properties": [
    {
      "name": "$j",
      "type": "double",
      "value": 156.5
    }
  ],
  "proposal": 1,
  "instrument": 20,
  "sample": 4
}
```

³ <http://www.balabit.com/network-security/syslog-ng/opensource-logging-system>

⁴ <http://logstash.net/>

⁵ <http://kafka.apache.org/>

⁶ <http://graylog2.org/>

⁷ <https://github.com/facebook/scribe>

⁸ <http://fluentd.org/>

⁹ <http://flume.apache.org>

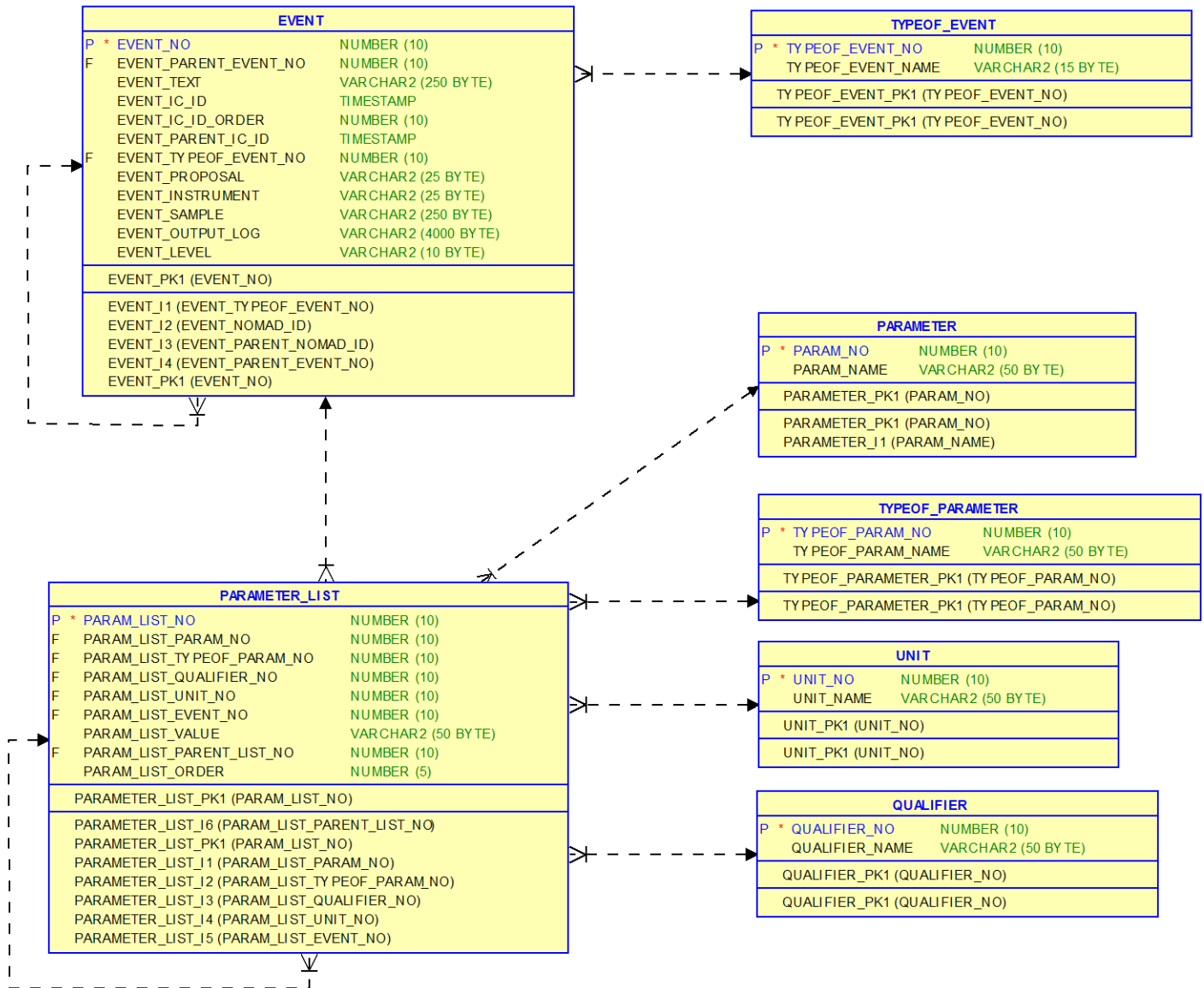
An acquisition:

```
{
  "id": "a9d212ffef3ce6bcd143b67172a1b576",
  "level": "info",
  "timestamp": 1380536490,
  "event": "acquisition",
  "type": "numor",
  "properties": [
    {
      "type": "number",
      "value": 678431
    }
  ],
  "proposal": 1,
  "instrument": 27,
  "sample": 897
}
```

2.5.2 Parsing the logs

In order to store the log messages in a structured database, they need to be parsed and understood. This is the role of the parser sitting in between the log aggregator and the database. The parser has to be adapted for each deployment. For the initial prototype the parser is a database store procedure.

2.5.3 Database schema



This Entity Relational Diagram presents the tables at the heart of the application, centred on the event and its list of parameters. More tables will be added as the log evolves (storing links to images and files, bookmarks ...) and feedback is received from the users.

2.5.4 Web application

As we have had success in the past with Symfony, an open-source web application framework, we have decided to use it for the development of this application.

Symfony provides a framework for improving and accelerating the development of applications (by structuring the development and reusing generic modules). The use of a framework facilitates long-term maintenance and scalability by complying with standard development rules. Compliance with development standards also simplifies integrating and interfacing the application with the rest of the information system.

As we want the application to be a rich user experience, most of the data coming from the database will be exposed as RESTful web services. The presentation layer will be driven by AngularJS and the Bootstrap CSS component framework provided by Twitter.

The whole application must be protected. We will be using Central Authentication Service (CAS) or Security Assertion Markup Language (SAML) to authenticate users. The web application will run behind the Apache web server.

2.5.5 RESTful services

Below is a draft list of RESTful resources. These are all protected behind the ILL's Central Authentication Service. Every response is returned in JSON with a standard response code and format.

Resource	Method	Description
/experiments	GET	Return a list of all experiments
/experiments/{id}	GET	Return details for a specified experiment
/experiments/{id}/logs	GET	Return all log messages for a chosen experiment
/logs	GET	Return a list of all logs
/logs/{id}	GET	Return an individual log message
/logs/{id}/media	GET	Return a list of all media associate to a log message
/logs/{id}/comments	GET	Return a list of all comments associate to a log message
/media	POST	Upload media to a log message
/media	GET	List all media

/media/{id}	DELETE	Delete media with a given identifier
/media/{id}	PUT	Update media with a given identifier

2.5.6 AngularJs and Bootstrap for frontend development

AngularJS is an open-source JavaScript framework, maintained by Google, which assists with running single-page applications. Its goal is to augment browser-based applications with model–view–controller (MVC) capability, in an effort to make both development and testing easier.


The library reads in HTML containing additional custom tag attributes; it then obeys the directives in those custom attributes, and binds input or output parts of the page to a model represented by standard JavaScript variables. The values of the JavaScript variables can be manually set, or retrieved from static or dynamic JSON resources.

The goal for the project is to make a single page web application that relies on models, the controller, web services, views and components to change its state.

All the data (log messages, comments etc.) will be read from the JSON web services provided by Symfony.

Bootstrap is a free collection of tools for creating websites and web applications. It contains HTML and CSS-based design templates for typography, forms, buttons, navigation and other interface components, as well as optional JavaScript extensions.

Bootstrap allows us to create responsive web pages that change their views to provide an optimal viewing experience depending on the device being used


ELogs

Home
My proposals
Account

Logs

Home / My Proposals / 123-52-444

Ferromagnetic-NSE study of the low-frequency spin-dynamical signature of the INVAR effect in Fe65Ni35

My bookmarks
Actions

Resume live updates

The logs will be updated automatically. You can pause the automatic updates by clicking on the button above

Filter by event type...
Filter by level...
Filter

Axis shift

EVALUATION ERROR

Mono1 retrying 2 of 5 actual 3.23m wanted 3.24m

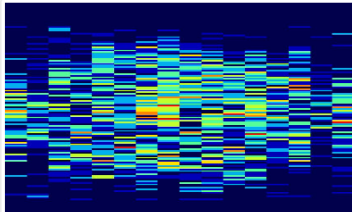
Event ID	Time	Detector	Rate det	Monitor1	Rate M1	Monitor2
4565890	2 seconds ago	6.920e+02m	2.307e+02	1.287e+05	4.289e+04	1.200e+02
4565891	2 seconds ago	6.920e+02m	2.307e+02	1.287e+05	4.289e+04	1.200e+02
4565892	2 seconds ago	6.920e+02m	2.307e+02	1.287e+05	4.289e+04	1.200e+02
4565893	2 seconds ago	6.920e+02m	2.307e+02	1.287e+05	4.289e+04	1.200e+02

2 seconds ago
2 annotations
Share
Bookmark event
Add annotation
Delete event

Image

CONTROLLER INFO

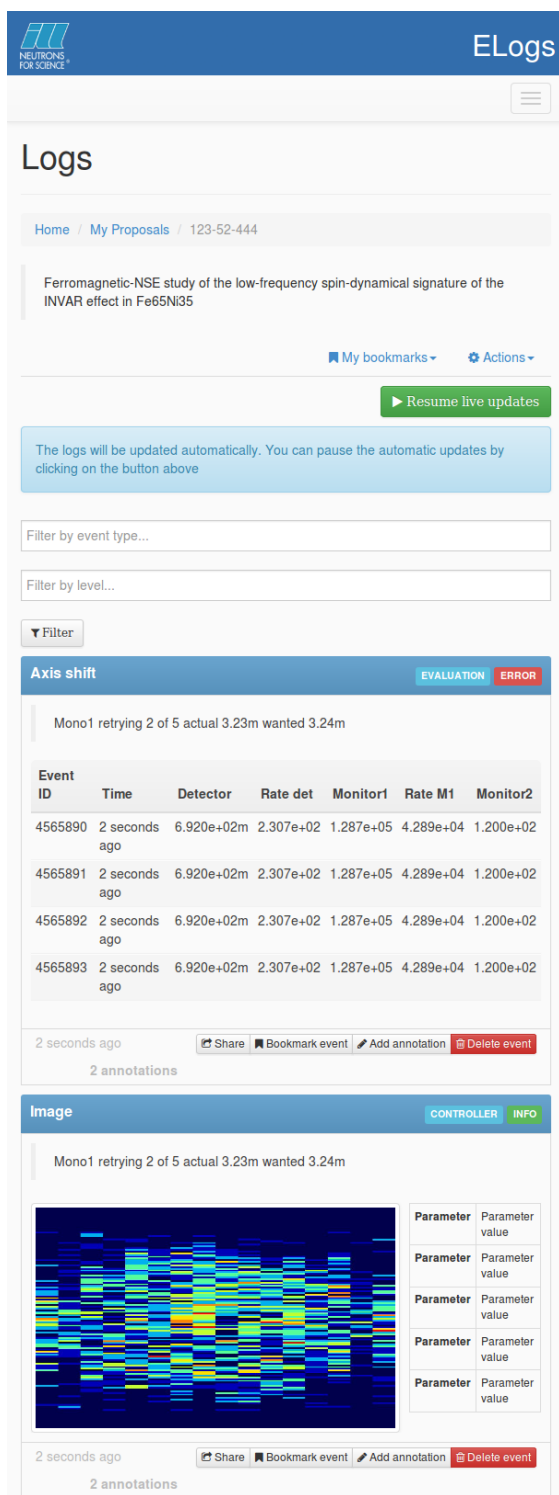
Mono1 retrying 2 of 5 actual 3.23m wanted 3.24m



Parameter	Parameter value
Parameter	Parameter value
Parameter	Parameter value
Parameter	Parameter value
Parameter	Parameter value

2 seconds ago
2 annotations
Share
Bookmark event
Add annotation
Delete event

Figure 7 Current state of the interface - Desktop view



If we combine the use of AngularJs and Bootstrap we can create a truly rich, fluid and responsive web application to enhance the users' experience.

2.5.7 Authentication with CAS or SAML

We will implement authentication through Central Authentication Service and SAML.

During the prototype phase, ILL's Central Authentication Service will be used to authenticate users' access to the web application interface.

We have decided that after a successful launch we will allow users coming from an Umbrella account to access the application if they can prove that they have a registered account at the ILL. For the test phase of the project we will only allow users with ILL credentials to access the application.

2.5.8 Real-time notifications using NodeJs

To offer a real-time experience to the user, we use websockets. Websockets provide full-duplex communication channels over a single TCP connection.

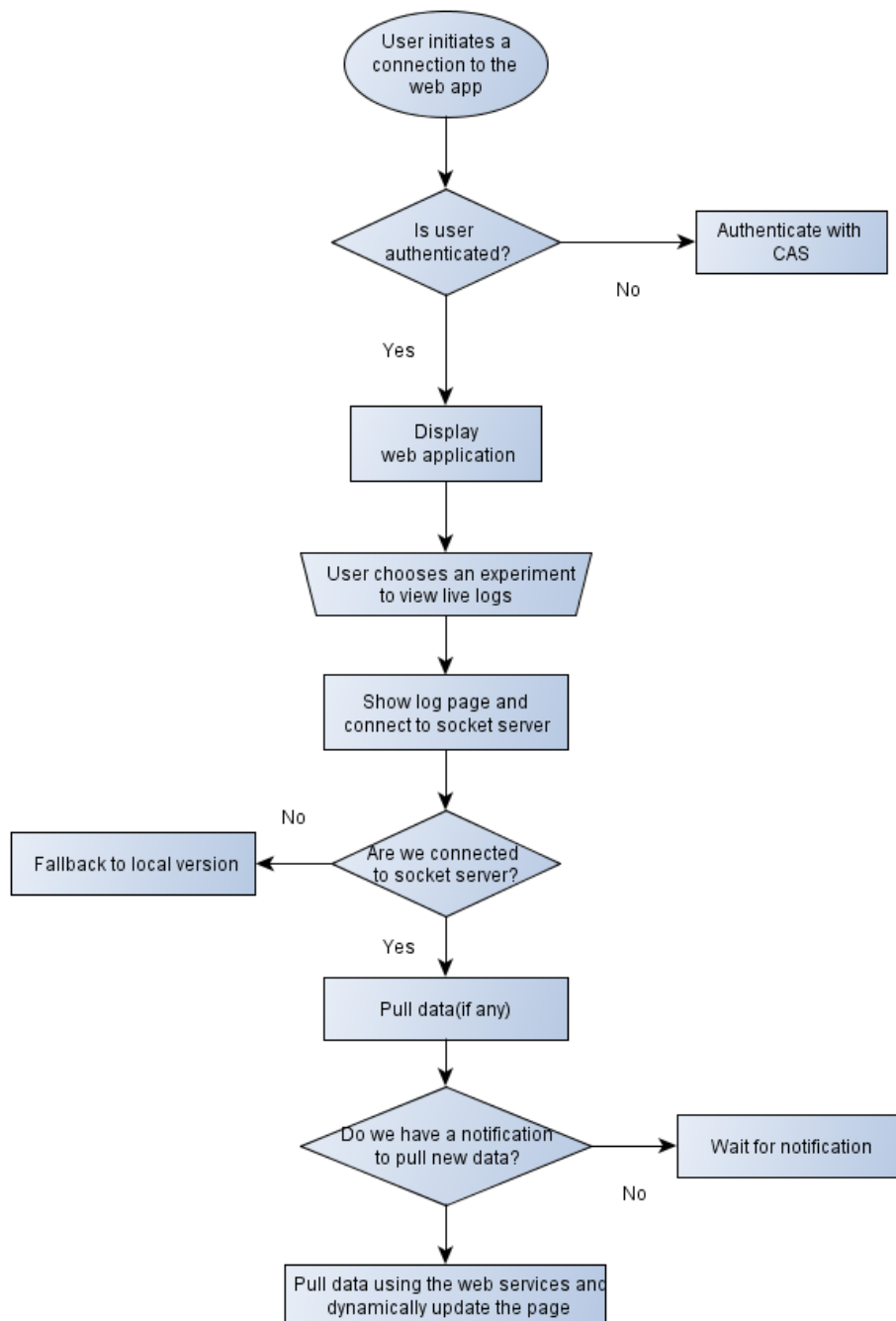
We implement a socket server using the ExpressIO framework for NodeJs, the event-driven I/O server-side JavaScript environment based on V8. We have also looked at other frameworks for implementing web sockets; the feedback from a number of users indicates, however, that NodeJs is the best choice, especially in terms of stability and maturity. The sole purpose of the socket server is to notify the user of any new data.

Figure 8 Current state of the Interface - mobile view

The idea, in essence, is simple; when the logger daemon (FluentD) receives new log data it informs the socket server; the socket server then notifies the

relevant clients of the data it has received. When a client receives notification, it updates the user's page using AJAX. We could have used AJAX to continuously poll the server every 10~ seconds, but we consider that this would unnecessarily overload the server. We want the client application to poll the server only when a notification has been received.

2.5.9 User access control



2.6 APPLICATION WORKFLOW

2.6.1 Normal operation

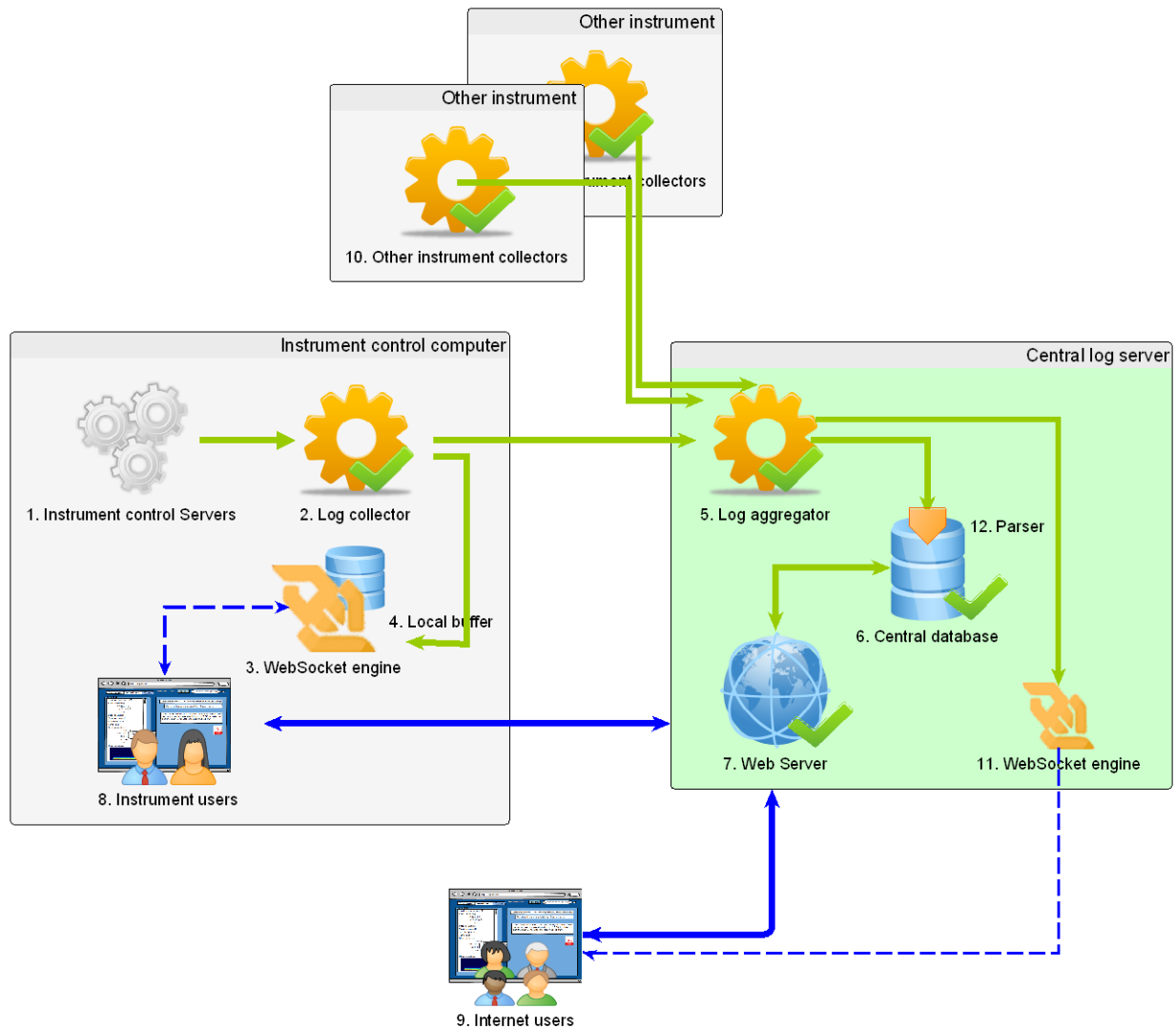


Figure 9 Normal operation architecture

During normal operation the logs are collected by the local log collector of the instrument control workstations (2); they are sent to the central aggregator (5) as well as to the local web socket engine (3) which buffers (4) them.

On the central log server, logs received from all collectors are parsed (12) and stored in the central database (6). Users (8 and 9) access the application to view and annotate the events through the central web server (7). If the experiment is live, the client application receives notification of updates from the central websocket engine.

2.6.2 Failure in accessing the central log server

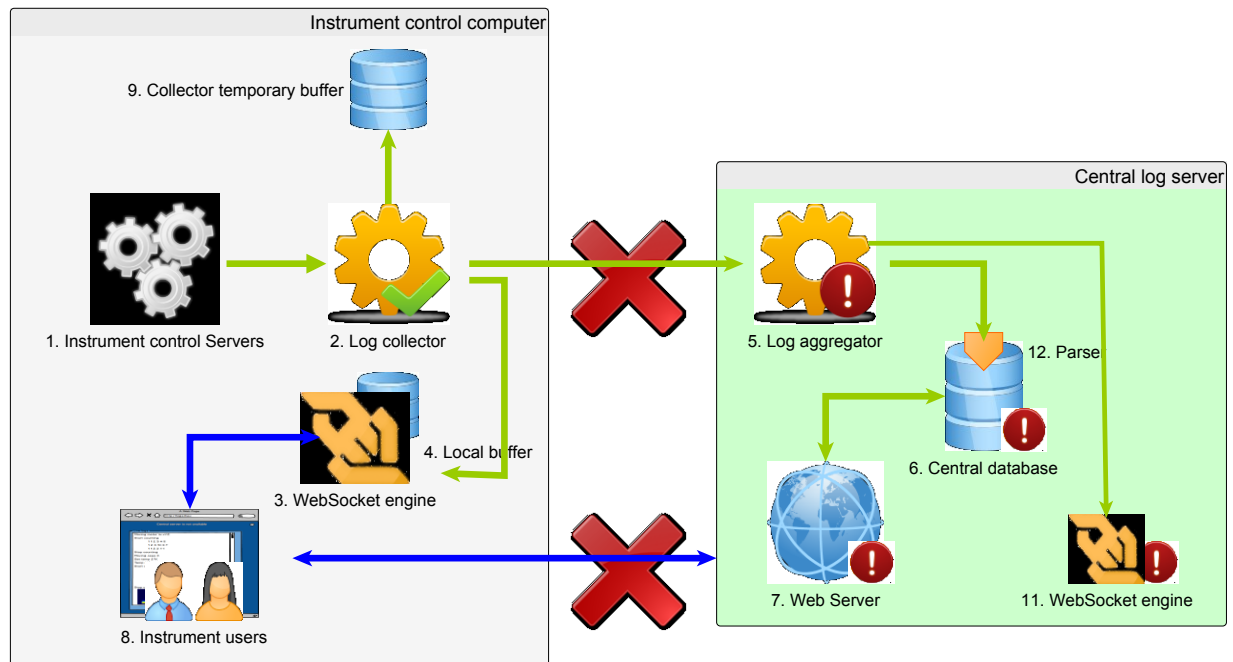
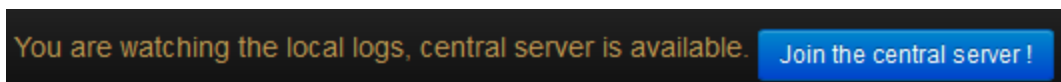


Figure 10 Failure mode

It is extremely important that experiments can continue to run and that users can monitor the logs even if the central log server is unavailable.

For such an eventuality, the client browser (8) on the instrument control workstation regularly monitors the availability of services on the log server through a polling mechanism (a JavaScript browser code polls a monitoring web service on the instrument control system through JSONP¹⁰). If the log server is down, the client code will retrieve the events (live and buffered) through the local websocket engine. When the central log server comes back up, users are invited to return to the (feature-rich) central application.



¹⁰ <http://en.wikipedia.org/wiki/JSONP>

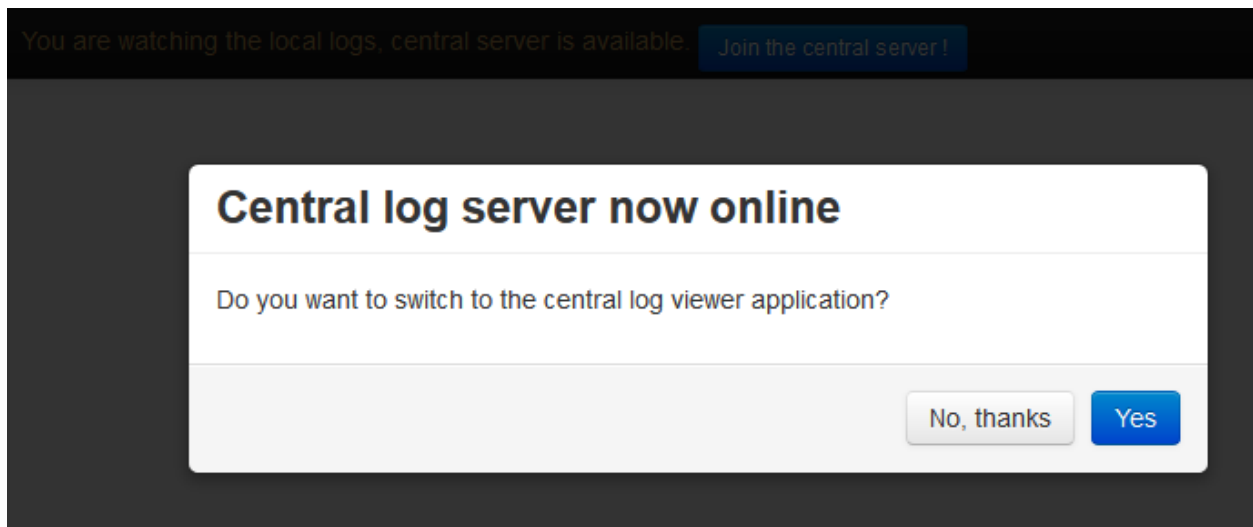


Figure 11 Central server availability monitoring

2.6.3 Resuming normal operation

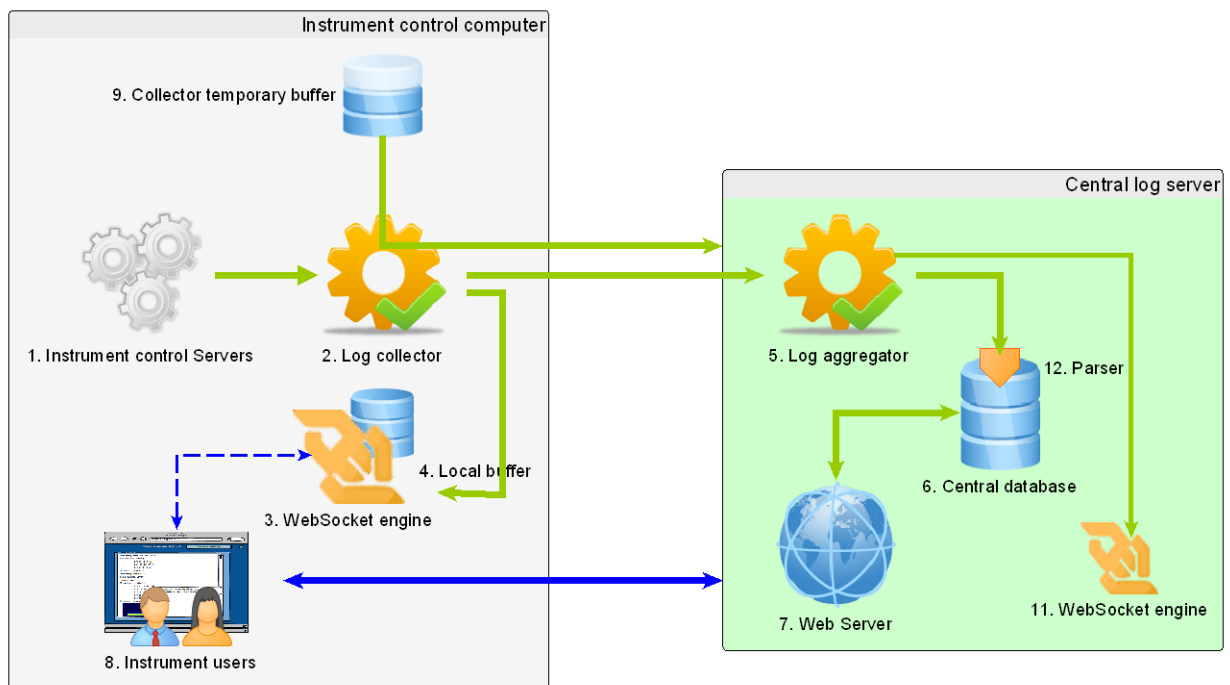


Figure 12 Back to normal operation

Once services return to normal, the log collectors empty their buffers (9) and users can resume their operations on the central server.

3 CONCLUSION AND ROADMAP

This prototype is being actively developed through the collaboration of IT and Instrument control staff at ILL. We will be performing functional tests with scientists before the end of 2013 whilst the system is deployed for testing on ILL instruments.

The development of the search functions will be postponed to the beginning of 2014. Development and testing should be complete by the end of the PaN-data ODI project.

The source code of the project will be publicly released as soon as it has reached a sufficient level of maturity (after a successful end of the test phase) under an open source licence which has still to be defined.

When the ILL reactor restarts in June 2014 the application will be deployed on all ILL and Collaborative Research Group instruments. By the end of the first 2014 experiment cycle in August 2014 we should have a clear idea of the impact such a service could have on user behaviour and the preservation of information.

If the project is successful, which we do not doubt, we will have made a major step forward in terms of data preservation. Data will be accessible and comprehensible, in the majority of cases, to the scientific community long after the end of the experiments.

This is a pilot action that will initially be deployed at full scale at the ILL. It is therefore necessary that ILL provides a report on the output of the initiative including statistics on its adoption (how many users have inserted annotations, what type of tags were used ...). This report will be produced in September 2014, right after the first period of usage. Those metrics will be valuable not only for the commission, but also for the PaN-Data consortium and the wider community of analytical facilities. This is a major element in view of the future adoption by the other facilities within the PaN-data consortium and beyond Europe.