The HTTP 'Black Box' Protocol for Control and Data Acquisition at JET

C.H.A. Hogben[1], F.S. Griph[1], P.J.L. Heesterman[1], M. Beldishevski[1], K. Kneupner[1], R.M.A. Lucock[1] and JET EFDA contributors*

JET-EFDA, Culham Science Centre, OX14 3DB, Abingdon, UK

[1]EURATOM-UKAEA Fusion Association, Culham Science Centre, Abingdon, Oxon OX14 3DB, UK
*See the Appendix of F. Romanelli et al., Fusion Energy Conference 2008 (Proc. 22nd Int. Conf. Geneva, 2008) IAEA, (2008)

ABSTRACT

The CODAS (Control and Data Acquisition System) and IT department of UKAEA Culham has developed a communication protocol for centralised simultaneous data acquisition, control and monitoring of a large number of processors. It is developed around the Hypertext Transfer Protocol (HTTP)[1] standard.

The protocol has now been in use for about 5 years. It is intended to allow parallel, collaborative developments to take place, by defining communication interfaces between systems, while allowing the internal details of each implementation to remain opaque. For this reason, it is referred to as the 'black box' protocol.

PROJECT BACKGROUND

- JET[2] is the world's largest Tokamak. It is operated since 2000 by the United Kingdom Atomic Energy Authority (UKAEA) on behalf of EFDA[3].
- JETs main role is to support the ITER[4] project - a steady addition of new equipment.
- CODAS[5] comprises the hardware and software necessary to perform controlled execution of pulsed JET experiments and retrieval of JET data.

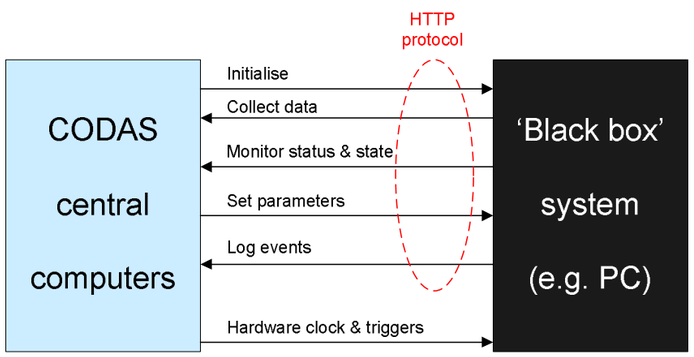

***Figure 1**. Scope of the HTTP 'black box' protocol*

SCOPE OF THE PROTOCOL

- Since 2000 a rapid increase in new JET instrumentation.
- Equipment developed by remote collaboration.
- Delivered in kind from EFDA associations.
- Requires a well defined interface to CODAS.

Collection of Experimental Data

The 'black box' protocol is intended for systems that acquire measurement data from the JET experiments; JET pulse events:

- Pre-pulse - systems are armed to acquire data
- The JET pulse - the instruments acquires data
- Post-pulse - systems may need to save data from memory to disk
- Data collection - during which the data acquired during the pulse is transferred to the central data store.

Status and State Monitoring

- The equipment operational status is narrowly defined: when it is not good there are one or more reasons why the instrument will malfunction on the next.
- Moreover JET equipment state-variables are used in the control room to reflect the equipment state.

Setting Parameters

Control room staff need to change and send settings of a system.

Logging

JET plant equipment may log information on the CODAS subsystem

REVIEW OF ALTERNATIVES

When deciding on a protocol, our primary motivation was data collection. Our goals were:

- easy interface to JET for black box implementers
- efficient transfer of data
- robust handling of error conditions

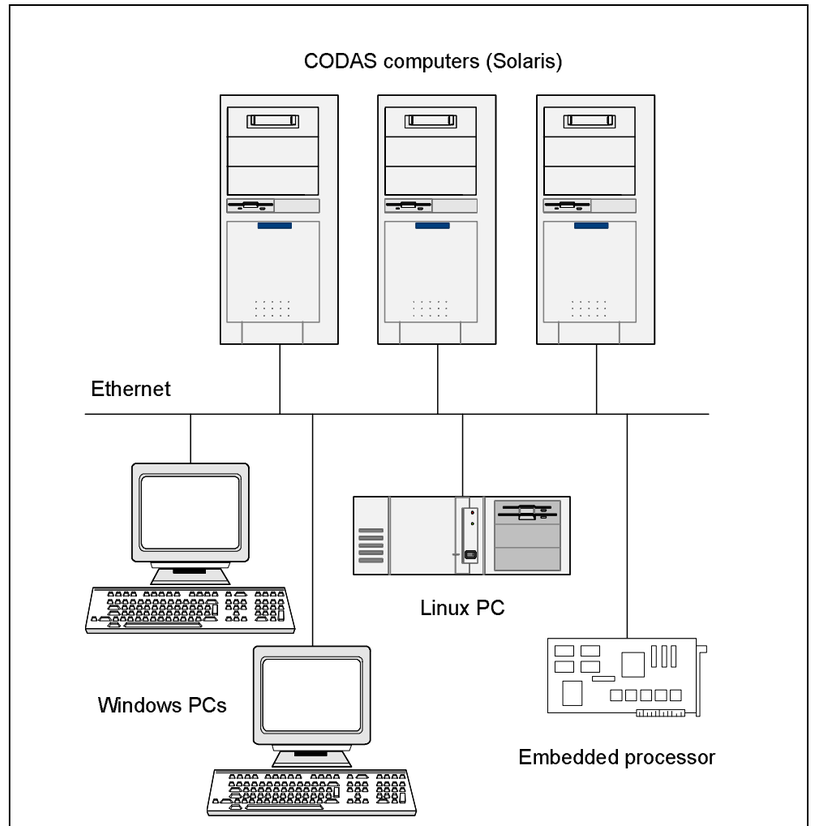

Figure 2. A variety of 'black box' systems are connected

Past Experience - Custom Message Protocol

Our previous efforts at interfacing PC-based systems to CODAS involved using an existing message protocol used within the JET project. This protocol is layered on TCP/IP and uses small, fixed-size headers and binary data items. A number of drawbacks were identified:

- ✘ Binary data caused difficulties - differences in byte ordering.
- ✘ A full low level protocol stack needed to be ported or re-implemented.
- ✘ The message protocol was designed for small messages, not for large data streams, for which short messages lowered performance.

File Transfer Protocol (FTP)

The early PC implementations used a file-based mechanism for data transfer. This led us to consider a simple network equivalent using FTP.

- ✓ elimination of the message server layer
- ✓ use of standard FTP server software
- ✗ it would not be easily extensible e.g. for error reporting;
- ✗ FTP is an awkward protocol to implement

Remote Procedure Call (RPC)

We considered several forms of RPC.

- ✓ potential re-use of existing software support
- ✗ no obvious cross-platform standard
- ✗ all we examined carried a significant infrastructure burden
- ✗ they were not designed for large data streams we would still have to design an application layer on top

BRIEF PROTOCOL DETAILS

The black-box system acts as an HTTP server and software on the CODAS computers as HTTP client. Each interface supported by a server (e.g. status or logging) corresponds to a Uniform Resource Locator (URL). A method call on an interface is implemented as a GET or POST request to the URL, with parameters passed using the same mechanism as submitting a web form with multiple fields.

Polled Monitoring

- A state GET request obtains a simple plain text response, in which each line contains one item name and its value.
- A status GET request obtains either an empty response, indicating that the instrument is operational; or a 'Structured Reason' (see later) containing one or more error reports.

Setting Parameters

- Equipment parameters are sent using POST requests.
- Read-back uses a GET.

Data Acquisition

- Data acquisition is initiated by one or more POST requests, typically one for each channel.
- Data collection is a GET request for each channel; the server returns either collected data or a Structured Reason for data acquisition error.

Structured Reasons

The 'black box' protocol uses a representation of undesirable states and processing exceptions called Structured Reasons[6]. (See figure 3.)

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<reason>
 <source
  uri="http://meta.jet.efda.org/DF/task/kg1cp"/>
 <text>Lasers not ready for pulse</text>
 <sub>
  <reason>
   <source
    uri="http://meta.jet.efda.org/diag/KG1/DCN"/>
   <text>DCN laser amplitude too low</text>
   <error domain="http://jet.efda.org/codas/errors/"
        number="13435003"/>
  </reason>
 </sub>
</reason>
```

***Figure 3**. Example of a Structured Reason serialised in XML, featuring a hierarchy of errors, human-readable text and codified information.*

BENEFITS AND EXPERIENCE

The choice of HTTP as the basis for the protocol carried a number of advantages, and no significant disadvantages.

Rich and Easily Extensible

The JET 'black box' protocol is a 'thin' specialising layer on top of HTTP:

- ✓ Bespoke HTTP header line to convey an item of metadata.
- ✓ MIME types are used to describe the content of the message body which allows for specialised data formats to be used..

Widely Deployed Standard

- ✓ Developers are likely to be familiar with the standard.
- ✓ There are several robust and well-supported open source and commercial HTTP implementations available.

CURRENT STATE

- Nearly 80 deployments of the 'black box' protocol in use.
- More than 60 HTTP servers on Windows PCs.
- Nearly 20 HTTP servers on Linux based systems.
- Several more HTTP servers are planned.
- The Windows PCs mainly use a framework developed at JET [8].
- Linux black boxes are using Apache, Tcl and Python.
- CODAS HTTP client (Solaris) use software components deployed into configured control applications using the Codas Component Framework (CFW)[9].
- The JET General Acquisition Program (GAP)[10] contains another HTTP client using the Curl library[11].

ACKNOWLEDGEMENTS

This work has been conducted under the European Fusion Development Agreement.

REFERENCES

[1] R. Fielding et al., "Hypertext Transfer Protocol - HTTP/1.1", IETF RFC 2616, September 2004, http://www.ietf.org/rfc/rfc2616.txt

[2] http://www.jet.efda.org

[3] http://www.efda.org

[4] http://www.iter.org

[5] H. van der Beken, et al., "CODAS: The JET Control and Data Acquisition Systems", Fusion Technology 11(1), pp. 120-137

[6] C.H.A. Hogben, F.S.Griph, "Interfacing to JET Plant Equipment Using the HTTP Protocol, Issue 2", August 2008, EFD-R(08)001, http://www.iop.org/Jet/article?EFDR08001.pdf

[7] T. Berners-Lee, R. Fielding, L. Masinter, "Uniform Resource Identifiers (URI): Generic Syntax", IETF RFC 2396, August 1998, http://www.ietf.org/rfc/rfc2396.txt

[8] P.J.L. Heesterman, et al., "The JetFsm Data Acquisition Framework, and Proposed Usage for ITER", PCaPAC, October 2008

[9] F.S. Griph, C.H.A. Hogben, M.A. Buckley, "A generic component framework for real-time control", IEEE Trans. Nuc. Sci. 51(3), June 2004, pp 558-564

[10] H. van der Beken, et al., "Data acquisitions at JET -- experience and progress", IEEE Trans. Nuc. Sci. 36(5), October 1989, pp 1639-1646

[11] http://curl.haxx.se/libcurl