

Portal Design, Synchrotron and HPC Services in e-HTPX

– A resource for High Throughput Protein Crystallography

Rob Allan (r.j.allan@dl.ac.uk), Colin Nave (c.nave@dl.ac.uk), **Ronan Keegan** (r.m.keegan@dl.ac.uk),
Dave Meredith (d.j.meredith@dl.ac.uk), Martyn Winn (m.d.winn@dl.ac.uk), Graeme Winter
(g.winter@dl.ac.uk), CCLRC Daresbury Laboratory, Warrington
Oleg Dolomanov (oleg@ebi.ac.uk) European Bioinformatics Institute,
Welcome Trust Genome Campus, Hinxton, Cambridge
Ludovic Launer (launer@embl-grenoble.fr), MRC France, ESRF, Grenoble
Paul Young (pyoung@ysbl.york.ac.uk), York Structural Biology Laboratory, York University
Ian Berry (ian@strubi.ox.ac.uk), Oxford Protein Production Facility, Oxford University

Abstract

The e-HTPX portal/hub has been designed to provide a single point of access for the coordination and remote execution of protein crystallography experiments. The portal acts as an access gateway and Web Service response hub to key synchrotron laboratory data collection facilities, and Grid accessible HPC services intended to aid the process of determining protein structures. The portal greatly simplifies experimental planning by first guiding the user through a compulsory sequence of Web Service communications with the synchrotron, and by providing an interface to input necessary data. The portal can also be used to monitor data collection experiments and access beam-line data. Two portal architectures have been designed in order to suit different research environment needs; a) the service-site portal, which is maintained by each synchrotron, offers a standard service for the majority clients and, b) the client-site portal, which can be installed at a client institution, and allows configuration and close integration of the portal functionality with a client institutional LIMS (Laboratory Information Management System). The client-site portal also places responsibility on the users for storage of their own, potentially sensitive data. Important to the e-HTPX high throughput services is the speedup and automation of data collection and processing. These requirements have been addressed with the application of HPC and the development of automated software designed to imitate the behaviour of an expert user on the synchrotron beam-line.

1.0 Overview

The volume of data coming from structural genome projects has generated a demand for new methods to obtain structural and functional information about biological proteins and macromolecules. This has led to a demand for high throughput techniques to determine the structure of important proteins. The e-HTPX project [1] is a distributed computing infrastructure designed to help structural biologists remotely plan, initiate and monitor experiments for protein crystallographic structure determination. Key services for e-HTPX have been developed by leading UK e-Science, synchrotron radiation and protein manufacture laboratories. The services developed for the project define the 'e-HTPX Workflow' as illustrated in Figure 1. Remote access to these services is implemented by a collection of Web and Grid services, each developed at the corresponding service-site. Client access to the e-HTPX Web Services is through the e-HTPX Portal, which provides an easy to use interface to input necessary data and also hides the user from the

complexities of the underlying distributed computing infrastructure.

2.0 e-HTPX Workflow

The initial stages of the e-HTPX workflow are centred on project planning, target selection and protein production (e.g. OPPF [2]). This involves the on-line completion and submission of requests to protein production facilities for the growth of specific protein crystals. Web Services have been developed to enable the remote monitoring of the progress of crystal-growth and to plan the delivery of crystals and associated safety information to a specified synchrotron.

Following crystal delivery to the synchrotron, the user accesses services designed to facilitate remote data collection. This stage includes services for deciding the best data collection strategy and services for data collection / monitoring. The key processes included in data collection are:

1. Matching crystals with appropriate experimental data that was previously sent to the synchrotron via the portal.
2. Automatically screening crystals at an unattended beamline.
3. Returning screening results to the scientist, with a recommended experiment plan.
4. Full data collection on crystals approved by the scientist and the return of: experimental statistics, compressed reflection files (MTZ format) and raw data.

In order to achieve remote data collection, the e-HTPX services link to a database situated at the synchrotron (ISPyB [3]). The data stored within ISpyB is then passed to DNA [4] which is a system for automating data collection and processing at synchrotron beamlines.

Using e-HTPX for the data collection stage provides the following benefits:

1. For the user, they do not have to travel to the synchrotron facility in order to collect their data.
2. For the synchrotron, the automated nature of e-HTPX allows efficient scheduling beam-time, i.e. the synchrotron may undertake experiments at times when scientists could not attend. This maximises the throughput and utilisation of the synchrotron resources.

3.0 e-HTPX System Architecture and e-Science Technologies

3.1 Portal

The e-HTPX portal is a configurable client web application designed to provide a single point of access to the underlying e-HTPX Web Service hierarchy. The portal provides GUI interfaces to input necessary data, and hides the user from the complexities of parsing WSDL files in order to generate Web Service client stubs. The portal also acts as a data repository and project management system, since input data and Web Service responses are stored and managed within a relational database.

Two portal architectures have been designed in order to suit different research environment needs; a) the service-site portal and b) the client-site portal. Architecturally, both of the portal implementations are equivalent clients to the underlying Web Services. The service site portal, which is hosted by the synchrotron, minimizes the

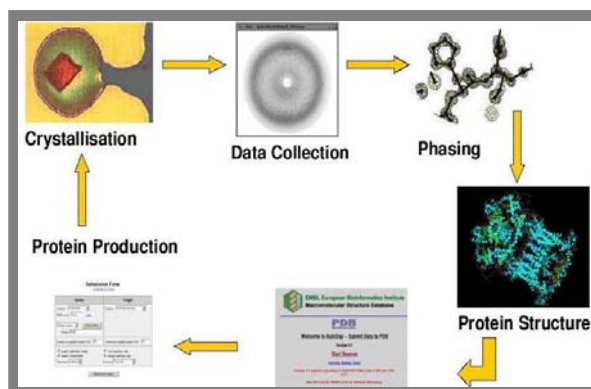


Figure 1. The e-HTPX Workflow spans all stages of protein crystallography experiments, from protein production to solution of the digital structure of the protein and deposition of the protein into a protein data bank.

When data collection is completed, HPC data processing services for determining the three-dimensional structure of the target protein are also provided and can be accessed through job submission Web Services and portal interfaces. Following post-data collection processing, the protein structure information determined in academic projects will eventually be deposited into public databases such as the Protein Data Bank provided by the European Bioinformatics Institute (EBI [5]).

requirements placed on the user (requiring only a web-browser). Conversely, the client site portal is intended to be installed and managed by a remote client institution (most probably installed behind an institutional firewall). The system architecture of the client site portal is illustrated in Figure 2 which provides details regarding the software stack, the typical configuration of the services hosted at synchrotron, and the sequential operations involved in invoking a service and polling for a response (follow steps A to F in the diagram - see section 3.2 for further details).

The client site portal was designed to satisfy additional client requirements that are not addressed by the service site portal. These include, a) the ability to store potentially sensitive data in the clients home laboratory database server, this is especially relevant to industrial clients (refer to

Figure 2), and b) the ability to integrate the e-HTPX Web Services and the corresponding portal interfaces with a laboratories information management system. For portal software development, we have adopted an object-oriented (OO) programming language, principally Java [6], Java-Servlets and JSP pages. The OO language can 'plug and play' to match the modular approach to the Web Service hierarchy. It is anticipated that the service site portal will be adopted by most (i.e. occasional) users and by clients who may wish to trial the core portal functionality, especially since no requirements are placed directly on the user. However, for more frequent users (or following initial trials with the service-site portal) the configurable client portal, will provide greater flexibility.

3.2 Web Services

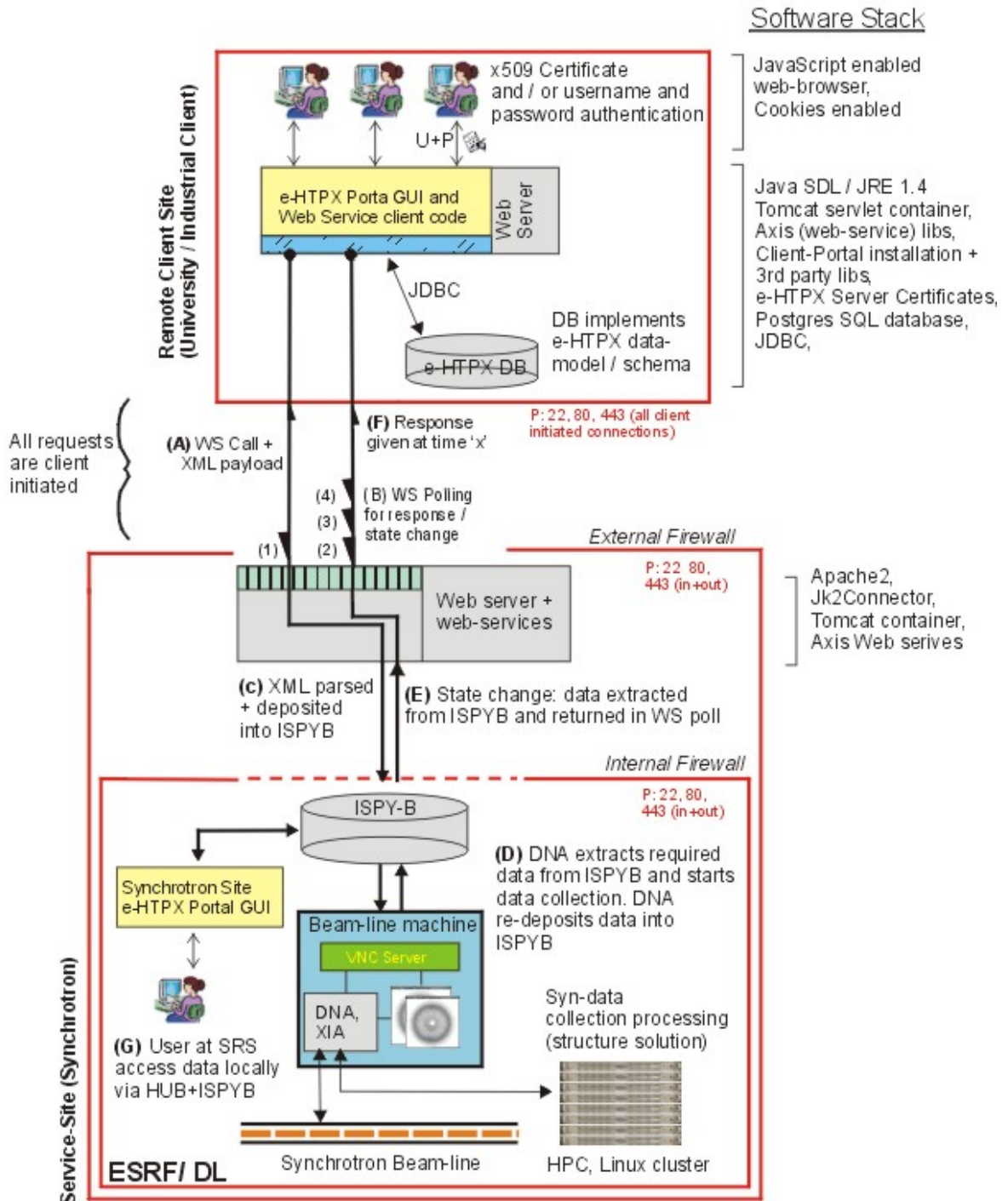
e-HTPX experiments requires numerous communications between the remote client and the various e-HTPX services hosted at the different laboratories involved in the project. Web Services were therefore chosen as the primary communication technology as the protocol allows clients to operate from behind an institutional firewall using standard HTTP(S) (which requires less firewall configuration compared to other protocols, e.g. Globus Toolkit v2 [7], which requires multiple multi-directional port openings). To further circumvent client firewall restrictions, all communications to e-HTPX services are client initiated, where the client invokes a service and polls for responses / results at a later date (no Web Service call backs are made to the client). This architectural model cleanly separates the service provider from the client. Security is currently provided by combining HTTPS server authentication with a BASIC client authentication scheme. The client sends encrypted data to the e-HTPX service (using the service's public key certificate to encrypt the data - X509 v3). Only the service provider can de-crypt the request with their private key. Once the SSL handshake is established,

the client is authenticated with a BASIC authentication scheme. A stronger client authentication is currently under development involving WS-Security specification.

With regard to Web Service style, a combination of RPC (remote procedure call) and document / messaging style services have been implemented depending on the data requirements of the service. For those services requiring complex input data, a comprehensive message model has been designed and implemented using XML schema. The message model describes a wide variety of data from diffraction plans to administration information. For e-HTPX, the main advantages provided by defining XML schemas are threefold: a) since XML is a powerfully descriptive language, the data required by the e-HTPX services can be precisely described and constrained, b) platform independence - individual service sites are free to implement their own services according to an agreed standard, and c) the client can create their own API's to work with e-HTPX from a range of standard XML parsing tools (such as Java's JAX-B parser [8], where the generated API can be used for data validation and binding i.e. transforming XML documents to Java object content trees for use in a clients own application). Supplementary data that cannot be encapsulated within the SOAP body (e.g. binary 'MTZ' files used in structure solution services) can be transmitted with the Web Service requests as SOAP attachments (SwA [9]). In doing this however, auto-generated WSDL files (such as those generated by Apache Axis Java2WSDL tool [10]) lack information about the attachments: this information has to be manually documented within the WSDL file and the service has to check for the inclusion of the required attachments.

As an exemplar, the following section describes one of the e-HTPX services in detail which involves remote access to HPC services (Bulk Molecular Replacement Service).

Figure 2 - Client Site Portal Architecture
(Follow Steps A to F)



↙ = Web service polling requests (indicating call direction). The web-service continuously polls until a response is ready at time 'x.' The small arrowhead represents the receipt of call-delivery. ↗ = HTTP(S)

(A) to (F) = When polling, an outgoing request is sent to the web-services (A). A response is then outwardly polled for (B). In this case, 3 polls are indicated by the repeating arrows 2) 3) 4). When a response is available, it can be returned in the response of the successful poll (F) (firewalls allow return of data along same connection if it is initiated from within the firewall).

4.0 An e-HTPX Web Service: Bulk Molecular Replacement

The successful completion of an X-ray diffraction experiment on the synchrotron beam-line initiates the process of determining the 3-dimensional atomic structure of the target protein. The data contained in the diffraction images is quickly reduced to a manageable form (MTZ file), which is then processed by several different programs to gradually extrapolate and build the structure. One of the most commonly used and computationally intensive parts of this process is Molecular Replacement (MR). As such, the utilisation of HPC technologies and resources can be of great benefit to this process and enhance the likelihood of its success.

4.1 Overview of Molecular Replacement

Many protein structures are solved using this method. The method involves the retrieval of a suitable homologous template structure from the database of all previously solved protein structures (Protein Data Bank, PDB [11]) using sequence matching and structure matching services such as those provided by European Bioinformatics Institute (EBI [5]). Using this template structure and the experimental data extracted from the X-ray diffraction images collected from exposing a crystal made up of the target protein to a beam of X-rays, it is possible to derive the 3-dimensional structure of the target protein. This approach is made possible by the fact that proteins that carry out a particular task in one species are found to be homologous to a protein that carries out a similar task in another species. All proteins can then be grouped into 'families' related by their structural make-up. With currently more than 20000 structures in the database, an unknown structure may have several related structures already deposited.

The difficulty lies in selecting and preparing a suitable template structure. The sequence identity between the target structure and the template structure is used as an indication of the suitability of the template structure. This is then used to modify and align the template structure with the target structure in an effort to optimise the chances of finding a solution. There are many different ways of deriving the sequence identity and, in cases where the homology between the target and template structures is relatively low (~30 %), choosing a good sequence identity and subsequent alignment is crucial if the target structure is to be determined.

4.2 The BulkMR Service

To aid this process we have developed a 'brute force' method for doing Molecular Replacement, BulkMR, which utilises HPC resources. With a large amount of computing resources, it is possible to retrieve and prepare many template structures to be used in Molecular Replacement. Each homologous model retrieved from the PDB can be manipulated in several different ways and then fed into an MR program to see if it can generate a solution for the structure of the target protein. The whole process is automated so that a user or an automated protein structure determination pipeline system can feed in the collected X-ray experimental data (in the form of an MTZ file) and retrieve an initial 3-D model for the structure of their target protein when the process has completed.

The BulkMR system takes advantage of much of the expertise that has been developed in the CCP4 [12] suite of programs and also forms part of the automation efforts currently taking place within this project.

4.3 Architecture of the BulkMR Service

For e-HTPX purposes, the system has been deployed as a Web Service on a dedicated Beowulf cluster housed at Daresbury Laboratory. This cluster is situated behind the firewall at Daresbury and is accessible via a gateway server in the Daresbury DMZ. The gateway server has access through the site firewall to the cluster using firewall IP-recognition techniques. BulkMR jobs can be submitted to this cluster in two distinct ways. The goal is to allow for modularity of the service so that it can be called as either a stand-alone application, (e.g. via the portal web-interface), or form part of an automated structure solution pipeline.

A WSDL description of the service has been made available so that it can be parsed and the service can be called as a function of some encompassing automated service such as the e-HTPX structure solution pipeline. The client code can communicate with the service on the cluster via an Apache2 server [13] running on the gateway machine. The input data along with the associated experimental data files are transmitted via HTTPS in a SOAP document with the associated data files included as attachments (refer to section 3.2). A results directory is created on the cluster to hold the output data from the job and the URL of this location is returned to the client so that it can then poll this URL to check results as they are produced and to determine when the job has completed. To allow the external client (automated system or user) access to the results folder on the cluster, its location is made accessible via an inverse-proxy

service running on the gateway machine. The URL contains a complex hashed name component, making it impossible to find if it is not previously known.

The Web Service endpoint is also accessible via the e-HTPX portal interface, which can be located at the server-site or client-site (see section 3.1). The interface prompts the user to enter various parameters and upload their experimental data files. The portal then makes the Web Service call to the cluster machine on behalf of the user. As the job proceeds, the remote user can access the output via the portal and upon completion of the job, the user is notified via email.

4.4 HPC technologies

The dedicated e-HTPX cluster for the BulkMR service is a 9-node dual Xeon processor machine with a gigabit interconnect. Linux is the installed operating system and job submission on the cluster is handled by Sun Grid Engine (SGE). We found SGE to be the most suitable queuing system for our needs as it supported both interactive and batch

submission of jobs to the cluster nodes. A secondary function of the cluster is to serve as a compute resource for synchrotron users at Daresbury. Many of the jobs run by these users require interactive control and SGE facilitates this. The BulkMR code is a python script, which calls upon several CCP4 programs to perform the template preparation, Molecular Replacement and refinement of the resulting model structures. The code generates SGE submission shell scripts for the each of the jobs so that they can be submitted to the cluster nodes. The 'master' python script keeps a handle on all of the various job details, results and files so that the processing involved in any resulting solution can be tracked and repeated if necessary.

The service has also been set-up to run on a condor pool at Daresbury Laboratory, utilising unused cycles on many of the workstations that are located there. When the e-HTPX service is rolled out to academic and industrial clients it is envisaged that the BulkMR service will be installed on Grid-available compute clusters such as those operated by the National Grid Service (NGS).

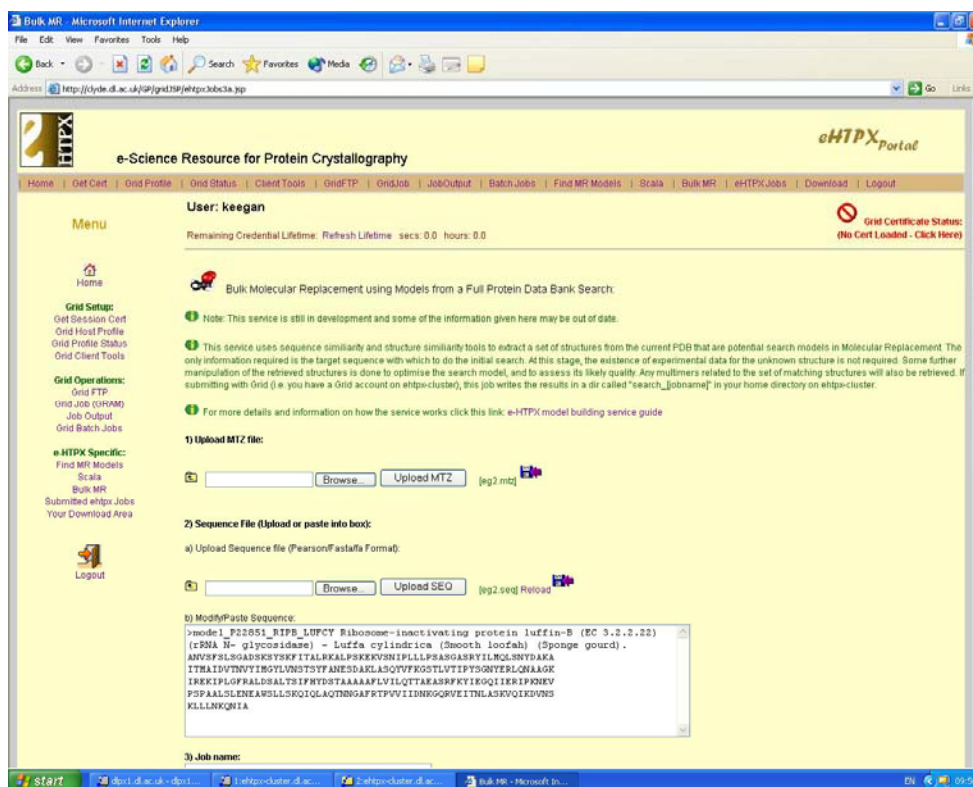


Figure 3 – BulkMR web interface hosted in the e-HTPX portal

References:

- [1] e-HTPX: An e-Science Resource For High Throughput Protein Crystallography: www.e-HTPX.ac.uk/
- [2] Oxford Protein Production Facility: <http://www.oppf.ox.ac.uk/>
- [3] ISPyB Project: http://www.esrf.fr/exp_facilities/BM14/escience/ispyb/ispyb.html
- [4] DNA Project: <http://www.dna.ac.uk/>
- [5] European Bioinformatics Institute: <http://www.ebi.ac.uk>
- [6] Java Technology: <http://java.sun.com/>
- [7] The Globus Alliance: <http://www.globus.org/>
- [8] Java Architecture for XML Binding (JAXB): <http://java.sun.com/XML/jaxb/>
- [9] SOAP Messages With Attachments: <http://www.w3.org/TR/2000/NOTE-SOAP-attachments-20001211>
- [10] Axis, Apache Web Services Project: <http://ws.apache.org/axis/>
- [11] Protein Data Bank: <http://www.rcsb.org/pdb/>
- [12] Collaborative Computational Project 4: <http://www.ccp4.ac.uk>
- [13] The Apache2 Web Server Project: <http://httpd.apache.org/>