LHSG-CT-2003-503420

BioXHIT

A European integrated project to provide a highly effective technology platform for Structural Genomics.

Life Sciences, Genomics and Biotechnology for Health

WP5: Ms 5.2.6 Proposed New Features for Database Visualiser

Due date of deliverable: **31.04.2007**
Actual submission date: **31.04.2007**

Start date of project: **1.1.2004** **Duration: 60 months**

Organisation name of lead contractor for this deliverable: CCLRC-CCP4
Daresbury Laboratory, Warrington WA4 4AD UK **Author** Peter Briggs

Identification of additional functionality for the Database Visualiser (BIOXHIT milestone 5.2.6)

Peter Briggs, CCP4

1 Introduction

This report describes proposed extensions to the initial implementation of the visualiser application associated with the tracking database, implemented in a previous deliverable. The proposals came from consultations with potential and actual users of the visualiser, and are intended to improve its usefulness for visualising the history of structure solution projects.

A subset of extensions will be implemented as part of deliverable D 5.2.14, reported separately.

2 Proposed Additional Functions

This section outlines the set of suggested data items that could be added to the current tracking database.

Proposed Data	Description
Additional panes to display extra information	Add new "panes" to show the list of jobs (similar to that in the current CCP4i), information about the files associated with a particular job and so on.
Remember setup between sessions	For example, the viewer could remember the last project that the user was looking at.
Show the names of files on the linking lines	… or have some way of accessing the files via text or an icon associated with a line that connects two job nodes
Display associated job attributes	For example, show an icon that indicates that a job has an associated notebook entry. Clicking on the icon could allow the attributes to be manipulated (e.g. clicking on the notebook icon would open a window for viewing or editing the notebook entry for that job). Another useful attribute to show would be if the job has any data harvesting files associated with it.
Provide data-centric view	Instead of the current view within the visualiser (which is a "process-centric" view that shows the steps taken joined by data flow), implement an alternative view that shows the data files as nodes joined by lines that represent the jobs.
Preferences options	Provide a way for the user to customise the viewer, for example by setting the default orientation or the colours used for the different job statuses. Other possibilities: • Choice of "rendering engine" (currently defaults to dot, but could be e.g. circo) • Choice of file format for outputting the views to file e.g. gif, jpeg etc • Choice of font style and size (currently fixed to be the "regular" CCP4i font).
Offer alternative graph layouts	By default the dot program is used to render the layout of graphs, however it might be useful to provide alternative layouts using the other Graphviz programs such as circo
Indicate when project is read-only	Provide some indicator to the user when a project is read-only (for example, some text in the message bar and/or an icon of a padlock, or whatever).

Customisation of colours	The visualiser currently uses green to represent FINISHED jobs and red for FAILED jobs, however this can cause problems for users with red-green colour-blindness. Possibilities include: using different colours, or colours with differing hues; using other visual indicators as well as/instead of colours to indicate status (e.g. a "tick" or a "cross" icon); allowing users to set their own preferred colours.
Display "subjobs" from tracking database	Allow expansion of a job node to display the substeps within that job, or else shrink the view of multiple subjobs back into one job. This requires the implementation of substeps or subjobs in the tracking database.
Option to "View Files from Job" and "Rerun" jobs	Add an option to view the input and output files associated with a job, and to rerun a job (cf the functionality in CCP4i).
Summarise logfiles in bubblehelp for jobs	The bubblehelp associated with each job could include a brief automatically-generated summary of the job, for example by showing the initial and final Rfree values from a Refmac run (Ronan Keegan).
Edit history data	Enable the user to manually create or remove links between jobs.
Search/selection tools	Addtional tools could be added to help navigate the view, for example: selection tools could restrict the view to a range of job numbers or dates (e.g. "show only jobs run on 16th April 2007")
Deposition functionality	Allow the user to specify jobs as particular steps in the structure solution process, to help with deposition. One way to do this would be to allow the user to mark particular jobs in the view as the final refinement step, the data reduction step and so on. The viewer would then be able to collect harvesting and other data files, run checking programs to check for consistency, and pass the information to starKey to be written into the output XML. This would extend beyond the the viewer, but we could for example create a processor that could turn the starKey XML into a command for running PDBEXTRACT or collect the correct files for deposition
Expert system functionality	An "expert system" helper program could use the connections between the jobs to try and suggest what tasks could usefully be performed next.
Add a key or legend to the display	This would be a small key that would explain what the colours of the boxes meant etc (similar to the key on a graph explaining which line is which).
Enlarge or reduce nodes depending on "significance"	Nodes that represent more "significant" jobs (for example, REFMAC or MOLREP jobs) could be shown as larger than those with less significance (for example analysis or conversion jobs). Conversely, the less significant jobs could be shown as smaller nodes
Show inputs/outputs from jobs with no parents/children	For jobs that have no children, show nodes that represent the output files for that job (and similarly for jobs with no parents, show the input files).
Annotation of jobs	Allow the user to easily add a descriptive note to a job.
Enable launching of external programs to view files	It would be useful to be able to launch Coot or CCP4mg directly from the visualiser in order to be able to view files from the jobs.

3 Selection and implementation of new features

It is the work associated with deliverable De 5.2.14 which will select which functions to implement, based on an assessment of priority and usefulness.