CCP4(i)/BIOXHIT Database Project:

Scope, Aims, Plans, Status and all that jazz

Peter Briggs, Wanjuan Yang

Outline of this talk

- Background: BIOXHIT and the CCP4 contribution
- Scope and aims
- Proposed project deliverables: system architecture
- Project Database Components
- Current status of this project
- Choice of technologies
- Links to other projects
- Summary

Background: the BIOXHIT Project

- **BIOXHIT**
  - Bioxtallography on a Highly Integrated Technology Platform for European Structural Genomics
  - EU Framework Programme 6 “Integrated Project”
  - 20+ partner institutions

- **Aim**
  - *“To provide platform for high-throughput structure determination from crystallisation to structure solution”*

- **Timeframe**
  - started 1st January 2004 for 4 years
  - problems with recruitment meant late start for CCP4

- **Website**
  - **`http://www.bioxhit.org`**

CCP4 Contribution to BIOXHIT

WP 5.2: Data Management & Project Tracking in Structure Solution:

- *"To fill the need for project tracking within the BIOXHIT structure solution software pipeline"*
- Pipeline covers software components post-data processing (scaling and merging, phasing, model building, refinement)
- Complementary to PIMS and DNA

Staff for CCP4 effort at Daresbury:

- Peter Briggs
  - *project coordinator for CCP4*
- Wanjuan (Wendy) Yang
  - *full time programmer*

History of CCP4 Database

- **Originally: minor project to increase accessibility & functionality of CCP4i job database**
    - Job database records details of tasks run
        - associated files, parameters, date, status …
        - no additional information e.g. relationships between jobs or crystallographic data
        - only accessible via CCP4i

- **BIOXHIT: expanded the remit to include:**
    - extended tracking i.e. relationships between jobs
    - crystallographic data
    - does not commit to providing a general data model for structure solution
        - but a CCP4 data model will be required

Scope

- **Deal with inputs to and outputs from software components post data processing up to structure validation and deposition**
  - aka “the software pipeline”

- **Within this:**
  - tracking information
    - steps taken (= programs run, decisions made, associated input/output files or other “data objects”)
  - crystallographic data (application-specific and “generic”)

- **Target users:**
  - “single user” performing manual/automated/mixed procedures
  - other modes of operation not requested/investigated

Aims

- Implement system for both manual and automated structure determination
    - *Use CCP4i as a starting point*
    - *Accommodate non-CCP4(i) applications*
    - *Small/lightweight database system to support single applications*

- Implement multiple database backends
    - *e.g. don't force user to have mySQL*

- Gather as much information as possible automatically
    - *e.g. can using this system give you tracking "for free"?*

- Recognise that structure determination will most likely not be performed exclusively within a single software package and that data will most likely not be stored in a single database
    - *exchange of data between systems requires standards for transfer e.g. standards developed in BIOXHIT WP 5.1*

Proposed Project Deliverables

Project Database Handler

- broker application to mediate interactions between database and client applications
- hides implementation of backend
- aim to provide client APIs to handler from different languages
- deal with multiple users/clients within/outside CCP4 system

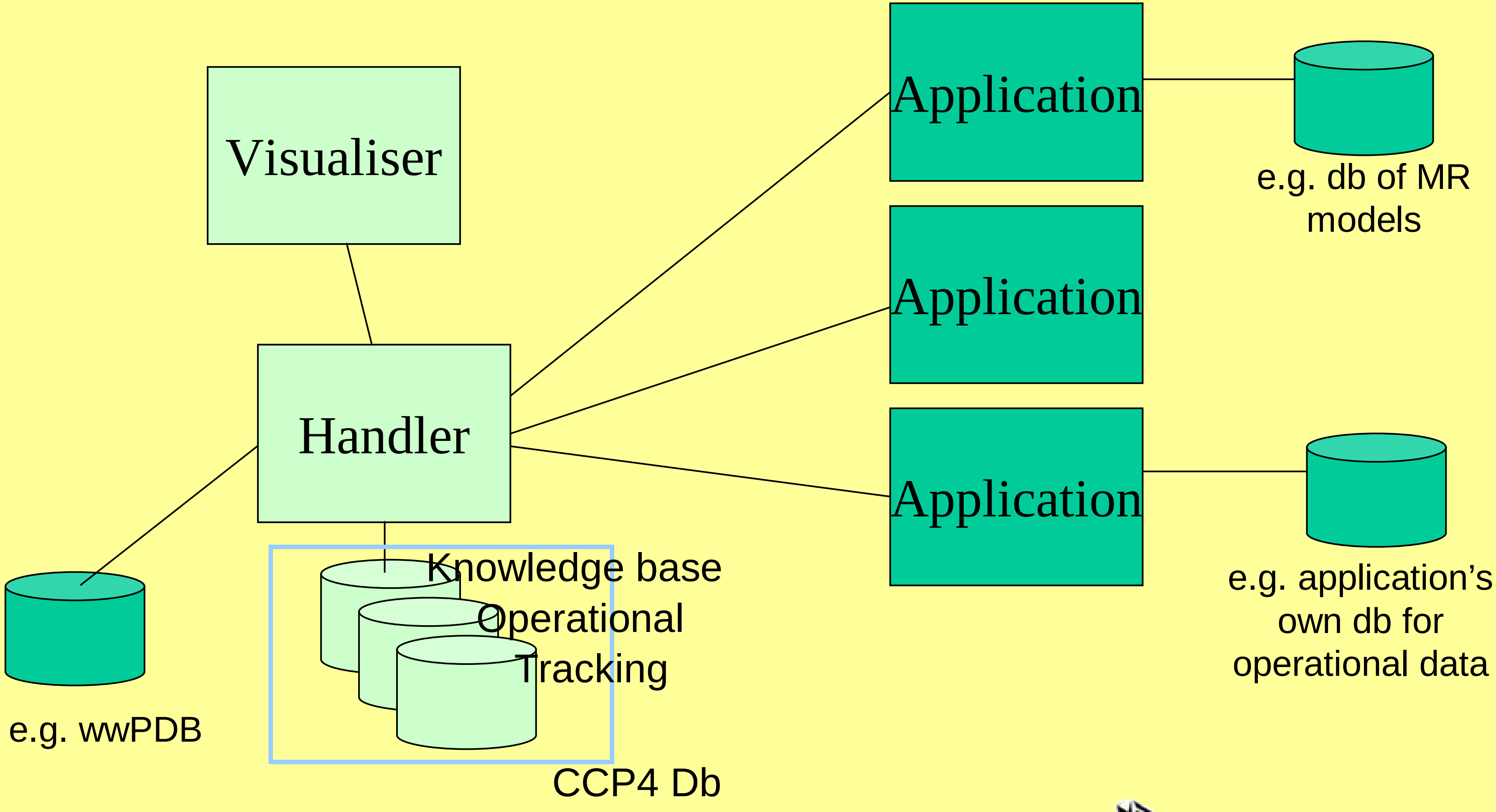
Project Database

- "tracking"/project history (steps in the determination process)
- project data ("knowledge base") & data history
- application-specific ("operational") data
- aim to provide database schema and multiple implementations

Visualisation Tools

- provide views of data to facilitate review and analysis

Architecture: how this fits together



Project Database Components

Operational ("internal") database

- application specific data & representations
- e.g. CCP4i parameter files, XIA python objects
- not intended to be shared between different applications

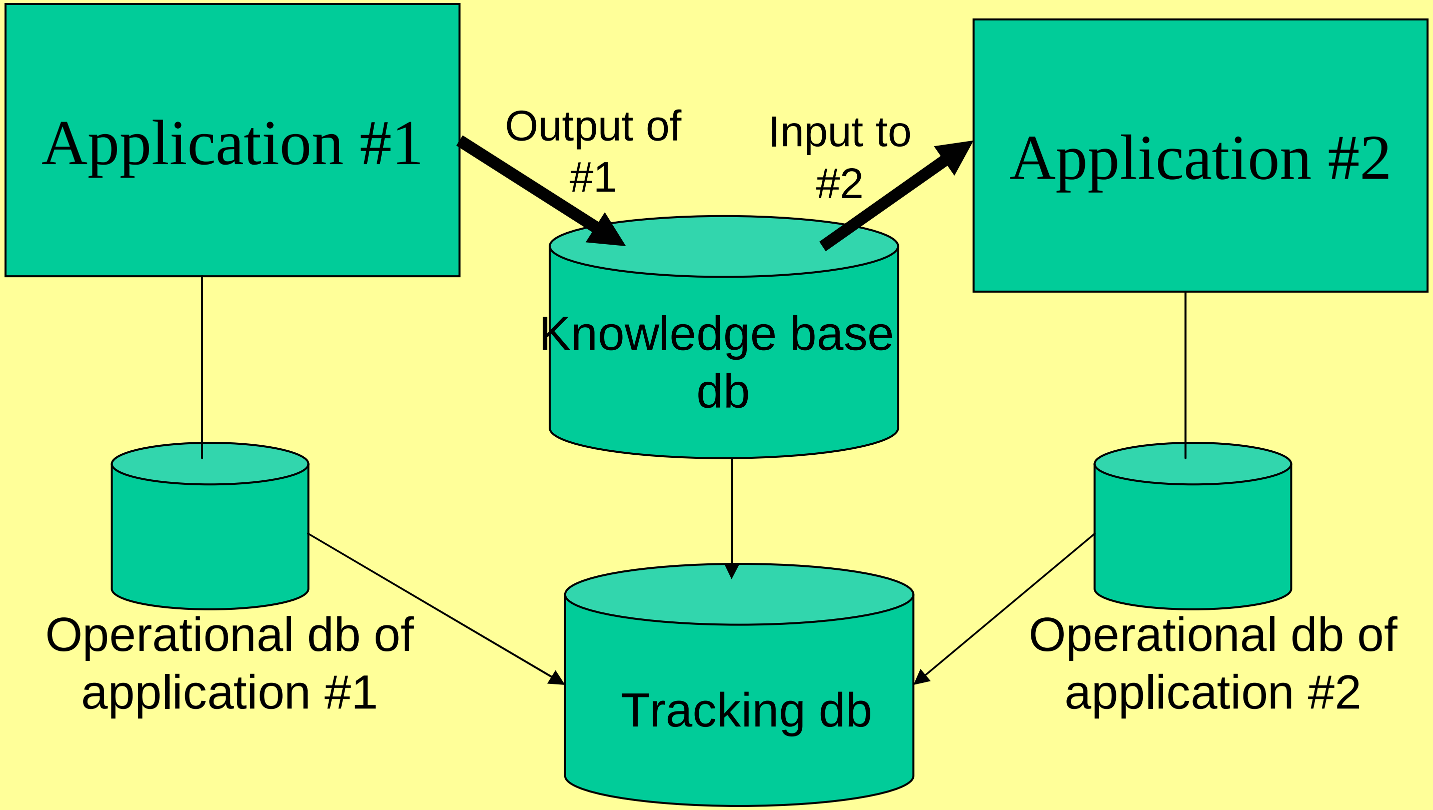
Knowledge base ("exchange" database)

- common crystallographic data items used within the software pipeline
- shared between different applications
- will require relevant info from earlier stages (crystallisation/data collection)
- must also provide information for final deposition

Tracking database

- project history
- contains links to the data in internal and external dbs
- relationships between data items/steps taken

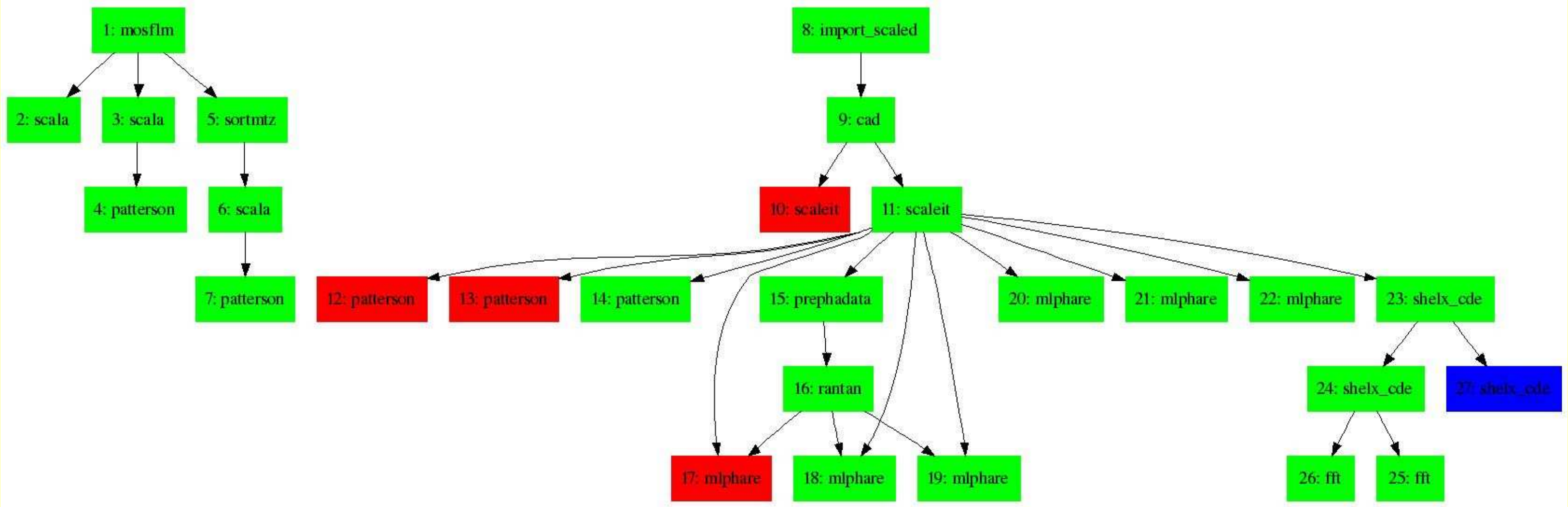
Database components: schematic



* Assumes handler layer is transparent

Data tracking and visualisation

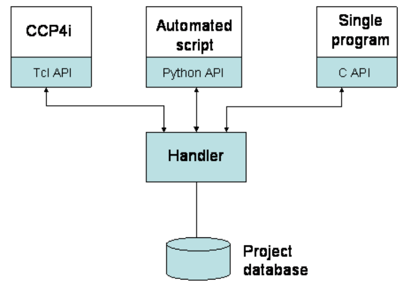
Structure determination can generate large quantities of data rapidly:



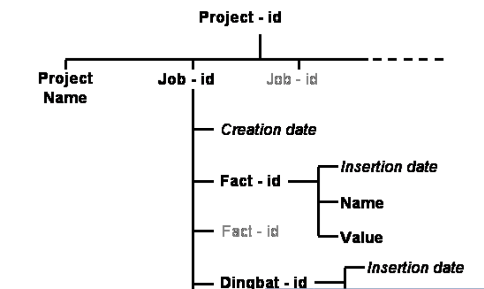
Tracking required to relate data items to each other:

- logical flow = steps taken e.g. programs run, decisions made
- data provenance = where did the data come from?

Prototyping of components

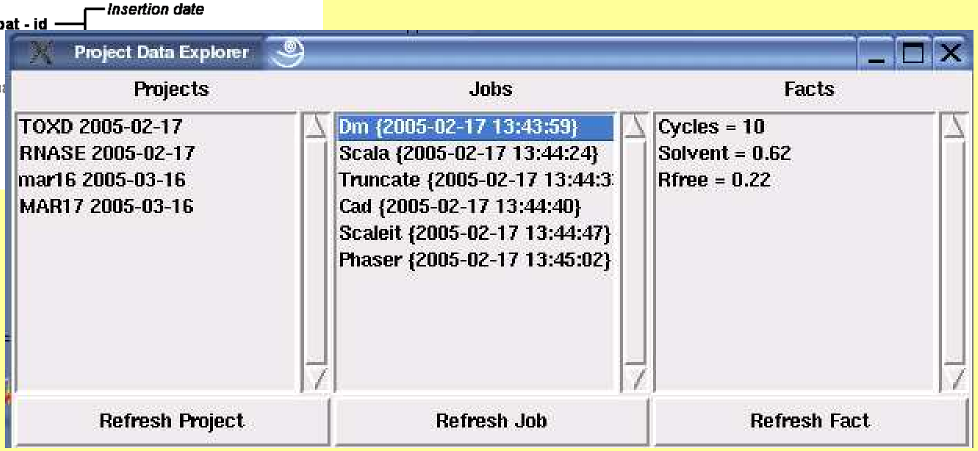


Prototype client-server



Prototype basic database schema

Project Data Explorer



TOXD 2005-02-17
RNASE 2005-02-17
mar16 2005-03-16
MAR17 2005-03-16

Dm {2005-02-17 13:43:59}
Scala {2005-02-17 13:44:24}
Truncate {2005-02-17 13:44:3
Cad {2005-02-17 13:44:40}
Scaleit {2005-02-17 13:44:47}
Phaser {2005-02-17 13:45:02}

Cycles = 10
Solvent = 0.62
Rfree = 0.22

Current status of components

Work so far focused on prototyping tracking databases

- revealed need for operational and knowledge base
- little work on exchange database
  - will be informed by this meeting

What we can say now:

- contents of exchange database will be dictated by
  - interfaces between applications
  - data needed for deposition
- needs of applications will also dictate what information is required from downstream/upstream databases
  - i.e. development should driven by applications' requirements

Choice of technologies

Handler is written in Python

- APIs provided in Python and Tcl

MySQL backend for prototype

- robust & easy to develop
- allows us to duck concurrency issues
- may not be good long-term choice

XML used as messaging technology within CCP4 db

- BIOXHIT heavily committed to XML
- CCP4 automation and related projects also likely to use XML

Use socket communications between server and clients

Links to other projects

Current collaborators:

- **e-HTPX/XIA**: Graeme Winter
- **Happy:** Dan Rolfe/Charles Ballard
- **CCP4i**: Peter Briggs

Other relevant projects:

- **PIMS**: requirement to able to exchange data

- Other projects interfacing with CCP4(i):
  - **CCP4MG** & **Coot**
  - **CRANK** (Steven Ness/Leiden)
  - **MOSFLM** (transferring data from processing)

Summary

- **CCP4/BIOXHIT db aims to provide**
  - small/lightweight db system to support applications
  - database will divide into
    - operational data (application-specific)
    - knowledge base/exchange db (crystallographic information common to all applications)
    - tracking of data

- **Cannot provide**
  - data model for all structure solution software applications
  - central database implementation

- **Current status**
  - prototype handler, db schema and visualiser implemented
  - next stage is to revise and expand db schema
    - development is dependent on input from application developers