

# Core CCP4 Activities

## OUTLINE

- Core core activities
- Projects not covered elsewhere - ccp4i, libraries, data harvesting, parallelisation
- Licensing
- Future CCP4 releases:
  - What?
  - When?

## Core team:

Martyn Winn

Alun Ashton

Peter Briggs

Charles Ballard

Maeri Howard-Eales

## Staff on projects:

Pryank Patel

Ronan Keegan

Chris Morris



# Core activities

- User support
  - Mainly helpdesk at `ccp4@dl.ac.uk`
  - Handle ~20 a week, with varying amounts of follow-up
- Maintenance of the suite
  - New platforms
  - Bug fixing
  - Processing external contributions
  - Updating interfaces, documentation, examples
- Central services
  - Web pages, email lists, finances
- Publicity
  - Study weekend, meetings, posters, newsletters



before SW

and after



M.D.Winn, York, March 3rd 2004

# Central services

Email lists:

ccp4bb, ccp4-dev, ccp4-wg1, ccp4-wg2, autostruct, etc.

Web site:

www.ccp4.ac.uk, www.ccp4.ac.uk/maxinf, etc.

ftp server:

CCP4 suite, newsletters, itcl-python, Coot

CVS server:

CCP4 suite, ccp4mg, DNA

Bugzilla:

core, ccp4mg, mole  
available but need account

On-line programs:

CGI, web services  
when CCP4 installation unavailable, or not up-to-date



# From academic collaboration to virtual organisation

Lots of little jobs ...

... what is the point?

Single point of contact for user  
Exchange of software tools  
Repository for developer - donate and forget  
Continuity

## CCP4 in context:

5 core staff + 3 project staff + Keith + 3 to come

Mosfilm development group - 3

CCI - 7 ( + 2 PDA + 5 students )

EBI/MSD - 27

CCDC - 50



# Suite developments

## Various oddments:

- updating programs for crystal level
- use of dataset-specific cell dimensions in programs
- incorporating new programs: bulk, ncont, pdbcure, coord\_format, tlsextract, dtrek2scala, pdb-extract, etc. (involves some degree of CCP4-isation)
- handling of full spacegroup names
- changes to TLS stuff, e.g. output of vectors for ccp4mg, relaxation of TLS format
- changes to mapslicer, e.g. setting contour levels
- cif2mtz accepts more mmCIF files
- porting to new platforms, compilers



# ccp4i

## Expand scope

- New module: “Graphics and Viewing Utilities”
  - includes AstexViewer, Loggraph, Mapslicer, Rasmol, Topdraw
- New interfaces: areaimol, clustalW, polarrfn, data harvesting tool
- Mosflm integration task (with HRP)
- Lots of updates to existing tasks

## New tools

- Bubble help
- Handling multiple logfiles, e.g. from arp/warp
- Scatter plots in ccp4i

## Longer-term developments

- Development of Database handler (not yet live)
- Development of MTZ hierarchical viewer (with AB)
- Inclusion of updates from ccp4mg



# ccp4i - future plans

Future plans tied up with automation / BioXHIT / Mole

Meanwhile, on-going maintenance/development:

## General:

- Improve presentation of tasks and modules to user
- Improved administration of projects
- Extend CCP4i "command library", see e.g. map\_utils (ExtendMap etc)
- Provide standard "plug-in" interface folders, see e.g. "UniqueifyFrame1"

## Specific:

- Improvements to AMoRe interface
- Bring SSM into e.g. BEAST to perform automatic alignments
- Matthews task - add options to calculate mol.weight from sequence



# Libraries

## Done:

cmtz, cmap

csym - based around syminfo.lib

library\_file, library\_utils

cparser, ccp4\_program

## To do:

independent library distribution

test scripting interfaces (python, tcl, perl)

replace ccp4i that relies on parsing mtzdump, mapdump



# Data Harvesting

Mainly as part of EU Temblor - Pryank Patel:

More data harvested from CCP4 programs

More columns converted from MTZ

(see [autodep.ebi.ac.uk/sf.html](http://autodep.ebi.ac.uk/sf.html))

More user control over harvesting, Data Harvest Management Tool

Validation of data for deposition



All this in consultation with EBI/MSD, and RCSB.

General drive for more data to be deposited, e.g. “Proposed data items for X-ray structure determinations” agreed Berlin 2002



# Parallelisation and Distributed Computing

Mainly as part of e-HTPX - Ronan Keegan:

MPI version of Beast in 5.0

MPI version of Scala under development

Parallel I/O of MTZ files - cmtzlib\_MPI under development

Task farming script for integration in Mosflm

Task farming approaches for automation?

Aims:

Code-parallelisation of slow programs

Code-parallelisation of programs for real-time feedback

Task farming often more appropriate

Aim to make best use of cheap clusters



# Licensing

Summary of new CCP4 licence. Three sections with default conditions:

Part 0: LGPL + additional right

Part 1: as before

Part 2: as before

In all cases, Copyright retained by contributing author/institution.

For any part, can override with individual conditions. Restrictive licences, such as GPL, are discouraged....

Agreement with contributing authors:

CCP4 has non-exclusive right to distribute source code and binaries.

Developers accept CCP4 distribution policy, inc. commercial licence.



# FFTw licence

CCP4 have licensed FFTw 2.1.5 from MIT for distribution within CCP4.

It can be distributed from DL and official mirror sites.

CCP4 users do not require any additional licences.

However, CCP4 users do not have the right to re-distribute this version of FFTw or binaries that include it.



# CCP4 Licence status

- Licence revised following queries from some companies. Subsequently addressed desires of developers.
- Sought agreement between CLRC, academic and commercial users, developers, Executive members.
- Agreement reached several times ....
  
- Waiting for agreement on latest queries (turnaround time with Sales Contracts (RAL) several weeks)
- When commercial licence agreed, need to generate academic licence
- Timescale???



# CCP4 5.0

- Code ready
- Bug fixing continues, but has reached equilibrium level
- Waiting on licence

Beta release available to developers:

<ftp://ftp.ccp4.ac.uk/pub/pjx/ccp4/ccp4-5.0-beta1.tar.gz>

36 unique downloads recorded 20/02/04 - 01/03/04



# CCP4 releases

## Provides:

Ease of installing many programs at once.

New programs/libraries can be tried because they're there.

User forced to upgrade all at once.

Standard environment for project management, i/o style, etc.

Versioning

Consistency within CCP4 and to some 3rd-party software

## Release strategy:

Frequent enough to keep up-to-date

Too frequent is annoying for users

Overhead associated with creating release

Never achieve nbugs = = 0

Aim for about 1 per year (yes, I know ....)



# The shape of things to come ...

Lots in the pipeline:

ccp4mg

pirate, buccaneer

new Mosflm GUI

Coot

SSM

BP3, CRUNCH

Phaser

Phil's OO point group scorer

+ update to usual suspects

+ stuff from automation project



# Library development

From 5.0, we have:

Clipper - for grown-up applications

MMDB - for coordinate applications and ccp4mg

CMTZ, CMAP, etc. - for i/o, fine control of data, jiffy applications

other C libraries - general CCP4 utilities

FFTw

Next stage:

Re-write other libraries: fftlib, harvlib, libhtml

Include other libraries, e.g. mmut, cctbx, libraries from PE.

Make use of scripting interfaces (python, tcl)

Want to replace many Fortran applications:

Duplicated functionality, e.g. act/contact/distang/angles

Indecipherable code, e.g. mtz2various, truncate

Good but unused, e.g. restrain



# MDW's hitlist

To re-write or junk:

act, angles, contact, distang  
coordconv

mtz2various, xdldataman  
mtzmnf, mtzMADmod, freerflag

restrain  
surface/volume



# Scenario 1: the quick hit

Late 2004 - CCP4 5.1:

All teething problems with 5.0 dealt with  
pirate

1 or 2 replacement jiffies based on new library  
anything else which is ready

Mid 2005: - CCP4 6.0:

Include ccp4mg (release 2.0??)

Include cctbx, phaser (MR), bp3, crunch, buccaneer

New Mosflm GUI

Developments from automation project

Aim to use new libraries as soon as possible.

Defer other large developments.



# Scenario 2: the slow burn

Early 2005 - CCP4 6.0:

pirate and buccaneer?

several replacement jiffies/scripts based on new library

Include ccp4mg (release 2.0??)

Include cctbx, phaser (MR), bp3, crunch

New Mosflm GUI

Aim for another big release.



# Long-term issues

Change of emphasis to modern tools.

E.g. function objects accessed from applications, scripts, MG.

What are boundaries to graphics, automation, traditional suite ??

Databases are great. Do we make this a dependency of CCP4?

Simplify or expand GUI?

Workflows.

Flexibility vs. focus.



**The End**



M.D.Winn, York, March 3rd 2004