A Light for Science

ESRF

European Synchrotron Radiation Facility

EDNA : A framework for Advanced On-line Data Analysis

Olof Svensson

Data Analysis Unit / Instrument Support and Development Division

ESRF

Outline

- Advanced On-line Data Analysis
- EDNA
  - The collaboration
  - The framework
- Current ODA applications based on EDNA
  - MX
  - non-MX
- Future developments
  - Workflow tools

On-line Data Analysis I

- Data analysis performed as quickly as possible in order to provide feedback to the user(s):
  - Quick calculations for data visualisation
  - Quality of the data generated by the experiment – is the experiment successful?
  - Use of results for planning the experiment : e.g. MX strategy calculation
- Data analysis during experiment
  - Users leave with reduced data (in addition to raw data)

On-line Data Analysis II

- Combination of data acquisition and data processing
- Automation
  - Software - and hardware
  - Robustness crucial
- Impact on beamline efficiency :
  - Should introduce added value
  - No slow-down of experiments
  - No or little impact on beamline scientists / users workload

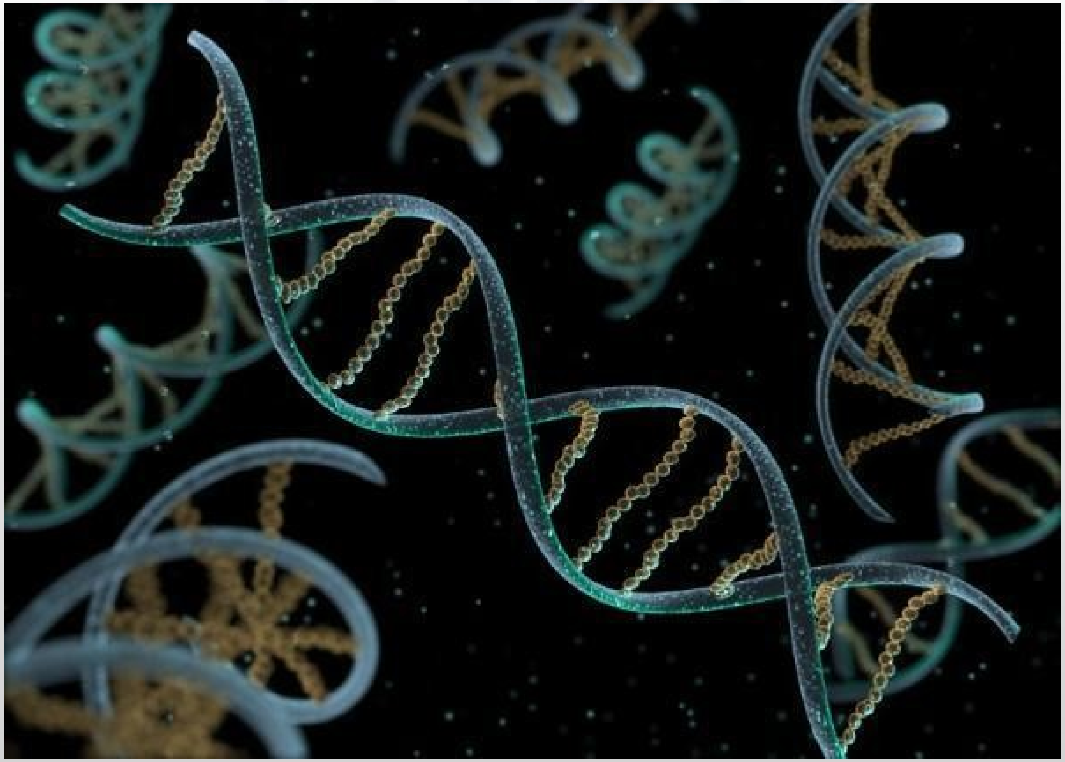
Advanced On-line Data Analysis - Examples

- Non-trivial data analysis :
  - Azimuthal integration followed by peak fitting and/or integration
  - Tomographic reconstruction
- Complex feedback to data acquisition :
  - MX data collections strategy
  - Modification of data collection strategy during data acquisition
- A combination of the two :
  - MX strategy data collections taking into account radiation damage and kappa geometries

On-line data analysis: Challenges

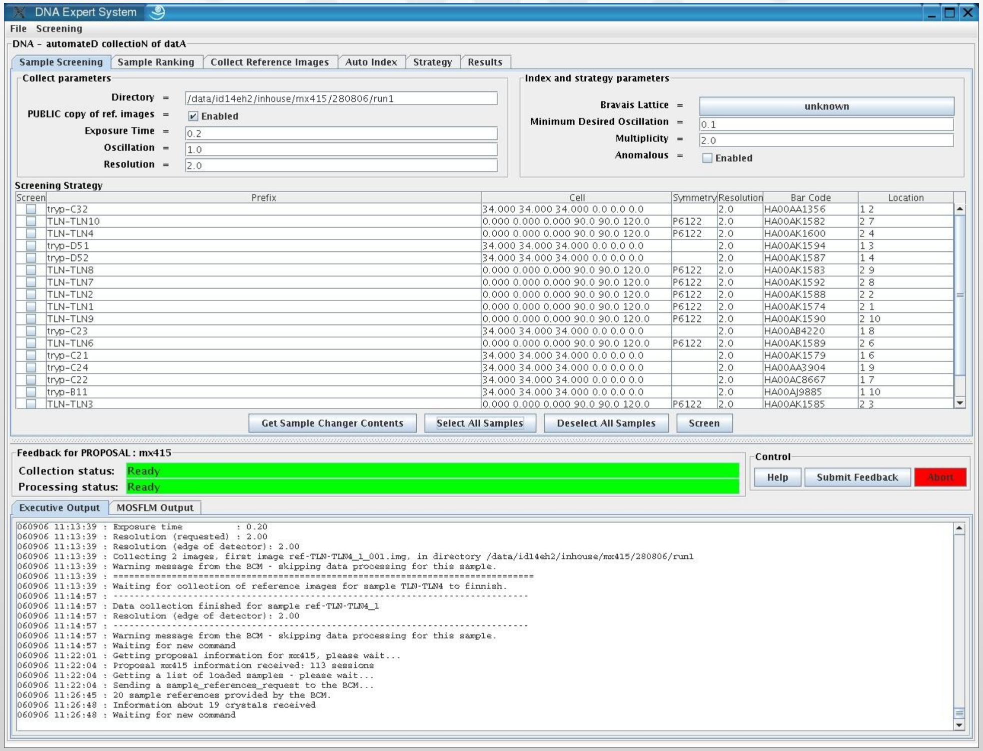
- Moving target:
  - What's today's state-of-the-art science might not be so tomorrow
- Limited resources:
  - Re-use existing scientific software
  - Collaborate
  - → Avoid re-inventing the wheel
- Automation
  - Robustness crucial
- Conflicting challenges!

In the beginning there was DNA...



… or RNA or something else, if you speak about life...

In the beginning there was DNA...



DNA - automateD collectioN of datA

Collect parameters

Directory = /data/id14eh2/inhouse/mx415/280806/run1
PUBLIC copy of ref. images = ☑ Enabled
Exposure Time = 0.2
Oscillation = 1.0
Resolution = 2.0

Index and strategy parameters

Bravais Lattice = unknown
Minimum Desired Oscillation = 0.1
Multiplicity = 2.0
Anomalous = ☐ Enabled

Screening Strategy

Screen	Prefix	Cell	Symmetry	Resolution	Bar Code	Location
☐	tryp-C32	34.000 34.000 34.000 0.0 0.0 0.0		2.0	HA00AA1356	1 2
☐	TLN-TLN10	0.000 0.000 0.000 90.0 90.0 120.0	P6122	2.0	HA00AK1582	2 7
☐	TLN-TLN4	0.000 0.000 0.000 90.0 90.0 120.0	P6122	2.0	HA00AK1600	2 4
☐	tryp-D51	34.000 34.000 34.000 0.0 0.0 0.0		2.0	HA00AK1594	1 3
☐	tryp-D52	34.000 34.000 34.000 0.0 0.0 0.0		2.0	HA00AK1587	1 4
☐	TLN-TLN8	0.000 0.000 0.000 90.0 90.0 120.0	P6122	2.0	HA00AK1583	2 9
☐	TLN-TLN7	0.000 0.000 0.000 90.0 90.0 120.0	P6122	2.0	HA00AK1592	2 8
☐	TLN-TLN2	0.000 0.000 0.000 90.0 90.0 120.0	P6122	2.0	HA00AK1588	2 2
☐	TLN-TLN1	0.000 0.000 0.000 90.0 90.0 120.0	P6122	2.0	HA00AK1574	2 1
☐	TLN-TLN9	0.000 0.000 0.000 90.0 90.0 120.0	P6122	2.0	HA00AK1590	2 10
☐	tryp-C23	34.000 34.000 34.000 0.0 0.0 0.0		2.0	HA00AB4220	1 8
☐	TLN-TLN6	0.000 0.000 0.000 90.0 90.0 120.0	P6122	2.0	HA00AK1589	2 6
☐	tryp-C21	34.000 34.000 34.000 0.0 0.0 0.0		2.0	HA00AK1579	1 6
☐	tryp-C24	34.000 34.000 34.000 0.0 0.0 0.0		2.0	HA00AA3904	1 9
☐	tryp-C22	34.000 34.000 34.000 0.0 0.0 0.0		2.0	HA00AC8667	1 7
☐	tryp-B11	34.000 34.000 34.000 0.0 0.0 0.0		2.0	HA00AJ9885	1 10
☐	TLN-TLN3	0.000 0.000 0.000 90.0 90.0 120.0	P6122	2.0	HA00AK1585	2 3

Feedback for PROPOSAL : mx415

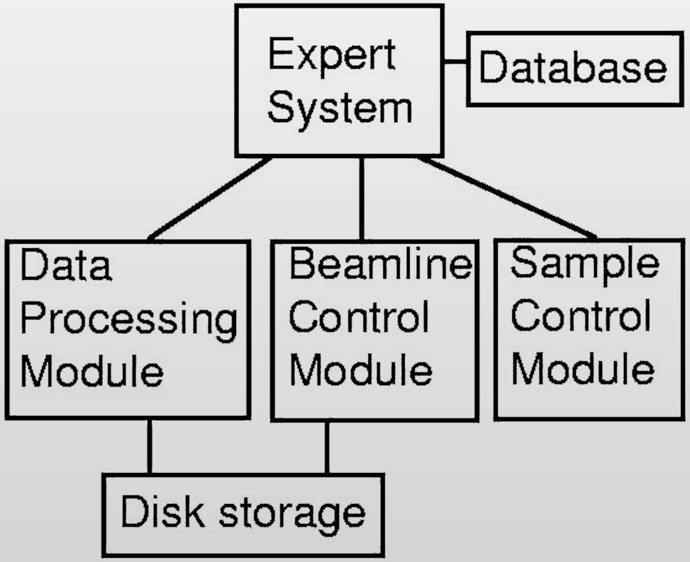
Collection status: Ready
Processing status: Ready

```
060906 11:13:39 : Exposure time          : 0.20
060906 11:13:39 : Resolution (requested) : 2.00
060906 11:13:39 : Resolution (edge of detector): 2.00
060906 11:13:39 : Collecting 2 images, first image ref-TLN-TLN4_1_001.img, in directory /data/id14eh2/inhouse/mx415/280806/run1
060906 11:13:39 : Warning message from the BCM - skipping data processing for this sample.
060906 11:13:39 : ============================================================================
060906 11:13:39 : Waiting for collection of reference images for sample TLN-TLN4 to finnish.
060906 11:14:57 : ----------------------------------------------------------------------------
060906 11:14:57 : Data collection finished for sample ref-TLN-TLN4_1
060906 11:14:57 : Resolution (edge of detector): 2.00
060906 11:14:57 : ----------------------------------------------------------------------------
060906 11:14:57 : Warning message from the BCM - skipping data processing for this sample.
060906 11:14:57 : Waiting for new command
060906 11:22:01 : Getting proposal information for mx415, please wait...
060906 11:22:04 : Proposal mx415 information received: 113 sessions
060906 11:22:04 : Getting a list of loaded samples - please wait...
060906 11:22:04 : Sending a sample_references_request to the BCM...
060906 11:26:45 : 20 sample references provided by the BCM.
060906 11:26:48 : Information about 19 crystals received
060906 11:26:48 : Waiting for new command
```

… if you speak about EDNA!

Beginning of automated MX characterisation... … or DNA in one slide

- Kick-off meeting in 2001
- Initial collaborators :
  - ESRF
  - Daresbury SRS
  - MRC LMB Cambridge
- Initially no external funding
- Meaning of ”DNA” :
  - automate**D** collectio**N** of dat**A**
- **Main development period 2001 – 2005**
- More collaborators and more developers entered the project, mainly thanks to external fundings : BioXHIT and e-HTPX.
- **Major component of the 2008 BESSY innovation award**



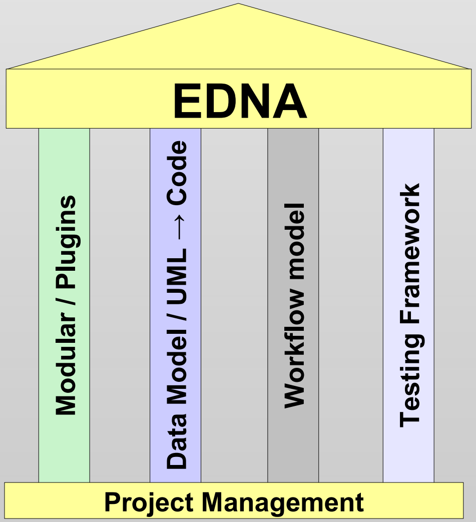
Why DNA became EDNA

- Many positive experiences learned from the DNA project :
  - Collaborative project : scientists / developers from several facilities working towards a common goal
  - Milestones achieved : automated MX characterisation

- However also many negative experiences :
  - The choice of name...
  - No project agreement, minimal project management
  - Not modular : difficult to add new features, difficult for new developers to enter the project
  - MX hardwired

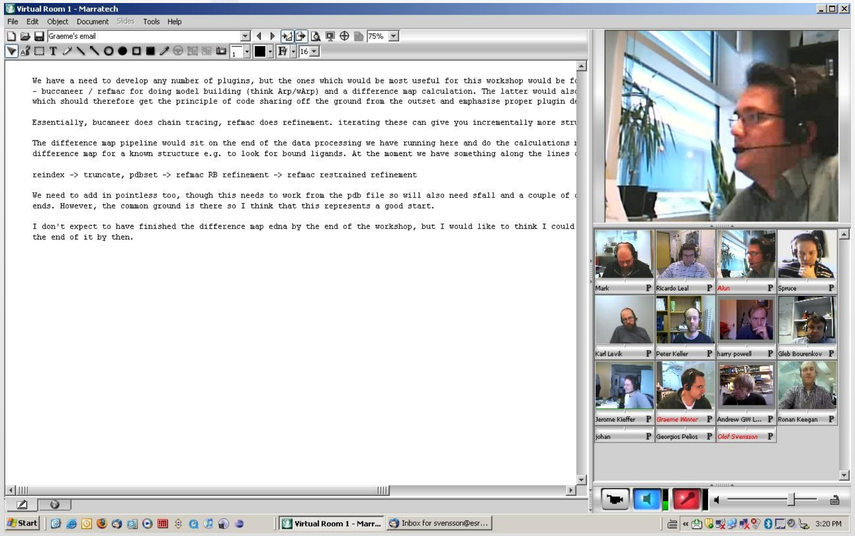
EDNA

- Project started in 2007
- Many collaborators and ideas brought over from the DNA project however no shared code
- Major development goals - collaborative developments :
  - Project agreement, project management
  - Workflow developments
  - Modular
  - Data model framework
  - Testing framework
  - Not hard-wired to MX



EDNA Collaboration

- EDNA is about collaboration:
  - Code sharing (SVN)
  - Coding conventions
  - Code reviews
  - Open source (LGPL, GPL)
  - Bug tracker
  - Wiki : http://www.edna-site.org
  - Memorandum of Understanding
  - Executive committee
  - Project manager / coordinator
  - Regular meetings / video conferences



We have a need to develop any number of plugins, but the ones which would be most useful for this workshop would be fo
- buccaneer / refmac for doing model building (think Arp/wArp) and a difference map calculation. The latter would also
which should therefore get the principle of code sharing off the ground from the outset and emphasise proper plugin de

Essentially, bucaneer does chain tracing, refmac does refinement. iterating these can give you incrementally more stru

The difference map pipeline would sit on the end of the data processing we have running here and do the calculations r
difference map for a known structure e.g. to look for bound ligands. At the moment we have something along the lines o

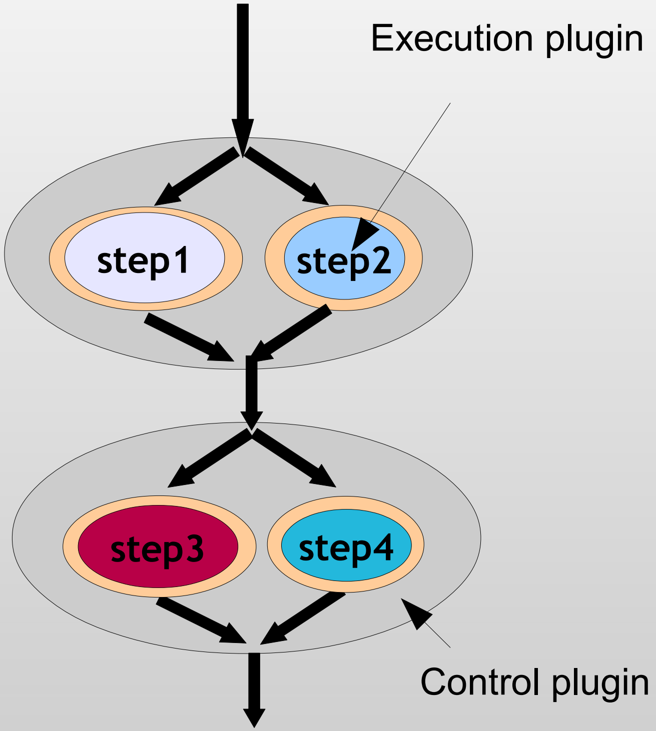
reindex -> truncate, pdbset -> refmac RB refinement -> refmac restrained refinement

We need to add in pointless too, though this needs to work from the pdb file so will also need sfall and a couple of o
ends. However, the common ground is there so I think that this represents a good start.

I don't expect to have finished the difference map edna by the end of the workshop, but I would like to think I could
the end of it by then.

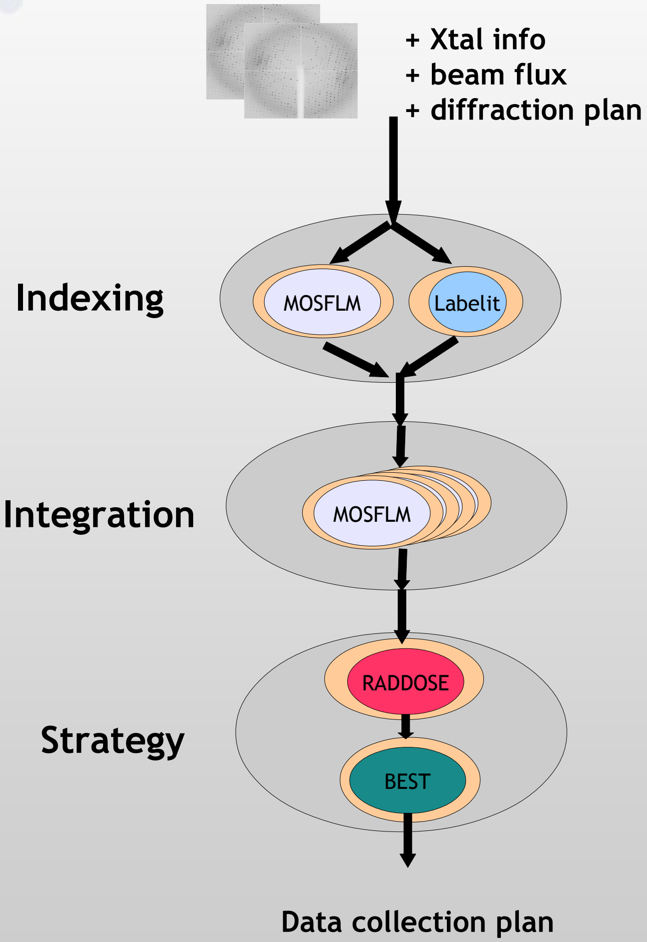
EDNA Modularity : Plugins and their hierarchy

- Plugin base class :
  - Configuration, working directory, etc.
- Execution plugins :
  - Execution of external programs, e.g. (bash) scripts
- Controller plugins:
  - Control of execution plugins
  - Parallel execution
  - Synchronisation



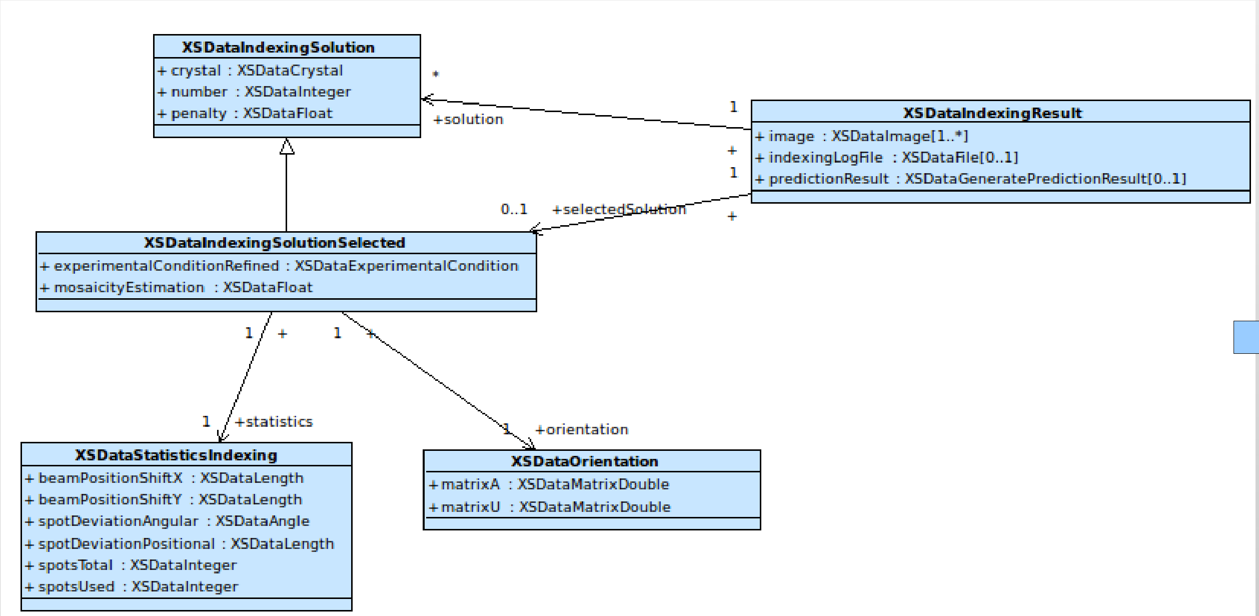
Example EDNA workflow : MXv1 Characterisation

- MX sample characterisation taking into account radiation damage
- Indexing using MOSFLM or Labelit
- Parallel integration of reference images
- If flux + beamsize:
  - RADDOSE for estimating radiation damage
- BEST strategy calculation
  - taking into account radiation damage
  - multi-subwedge data collection strategies



EDNA Data Modelling Framework

- Experiments produce data
- Data cannot in general be analysed without meta-data
- Complex experiments → complex meta-data



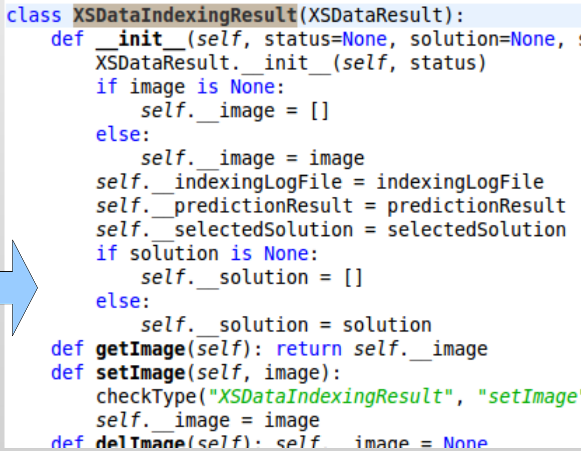
XSDataIndexingSolution
+ crystal : XSDataCrystal
+ number : XSDataInteger
+ penalty : XSDataFloat

XSDataIndexingResult
+ image : XSDataImage[1..*]
+ indexingLogFile : XSDataFile[0..1]
+ predictionResult : XSDataGeneratePredictionResult[0..1]

XSDataIndexingSolutionSelected
+ experimentalConditionRefined : XSDataExperimentalCondition
+ mosaicityEstimation : XSDataFloat

XSDataStatisticsIndexing
+ beamPositionShiftX : XSDataLength
+ beamPositionShiftY : XSDataLength
+ spotDeviationAngular : XSDataAngle
+ spotDeviationPositional : XSDataLength
+ spotsTotal : XSDataInteger
+ spotsUsed : XSDataInteger

XSDataOrientation
+ matrixA : XSDataMatrixDouble
+ matrixU : XSDataMatrixDouble



```
class XSDataIndexingResult(XSDataResult):
    def __init__(self, status=None, solution=None, s
        XSDataResult.__init__(self, status)
        if image is None:
            self.__image = []
        else:
            self.__image = image
        self.__indexingLogFile = indexingLogFile
        self.__predictionResult = predictionResult
        self.__selectedSolution = selectedSolution
        if solution is None:
            self.__solution = []
        else:
            self.__solution = solution
    def getImage(self): return self.__image
    def setImage(self, image):
        checkType("XSDataIndexingResult", "setImage"
        self.__image = image
    def delImage(self): self.__image = None
```

EDNA Testing Framework

- Unit, execution and regression tests:

```
 [UnitTest]: ########################################################################
 [UnitTest]: Result for EDTestSuitePluginExecuteAll : FAILURE
 [UnitTest]:
 [UnitTest]:  Number of executed test suites in this test suite : 5
 [UnitTest]:
 [UnitTest]:
 [UnitTest]:   Total number of test cases executed with SUCCESS : 124
 [UnitTest]:   Total number of test cases executed with FAILURE : 8
 [UnitTest]:
 [UnitTest]:
 [UnitTest]: OBS! The following test methods ended with failure:
 [UnitTest]:
 [UnitTest]:   EDTestCasePluginExecuteControlCharForReorientationv2_0_noKAPPA_ :
 [UnitTest]:     testExecute :
 [UnitTest]:       Plugin failure assert: should be False, was True FAILURE: Expected different from obtained - identifier
/mntdirect/_scisoft/users/svensson/tmp/EDTestSuitePluginExecuteAll_20110114-093729/tmpPW7olu
 [UnitTest]:
...
 [UnitTest]:
 [UnitTest]:
 [UnitTest]: Total number of test methods executed with SUCCESS : 129
 [UnitTest]: Total number of test methods executed with FAILURE : 8
 [UnitTest]:
 [UnitTest]:                                   Runtime : 1108.997 [s]
 [UnitTest]: ########################################################################
```

- Test suites executed automatically every night
- Continuous integration: nightly builds

Zen of EDNA

- Robustness is more important than performance
- Modularity brings maintainability
- Data-structures should be separated from code
- Flexibility comes with modularity & data-structures
- Community and collaboration to fight bugs
- Data parallelism above multi threading
- Performance & parallelism are provided by the EDNA-kernel

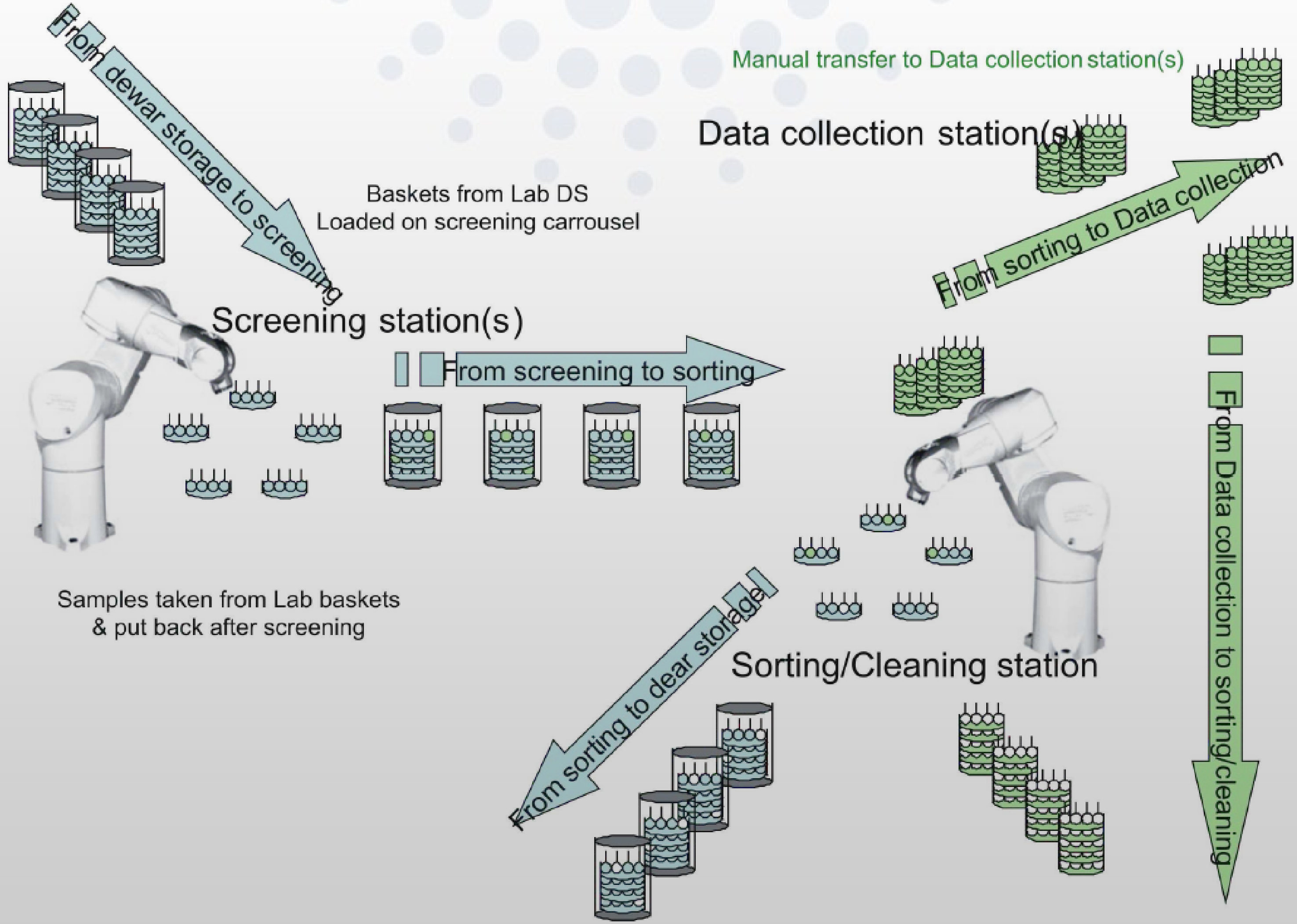
Edna toolbox

- > 100 execution plugins:
  - Generic command line execution, Image conversion, movied ...
  - HDF5 writers for stack of images, map of spectra
  - Conditional branching, accumulator of information
- 5 real applications available from repository (2 demo)
  - BioSaxs
  - DiffractionCT
  - Dimple
  - Fullfield XANES
  - MX v1 & v2
  - Demo projects: Ccp4 and Raw photography
- 3 kind of launchers: Command line, Parallel, Tango

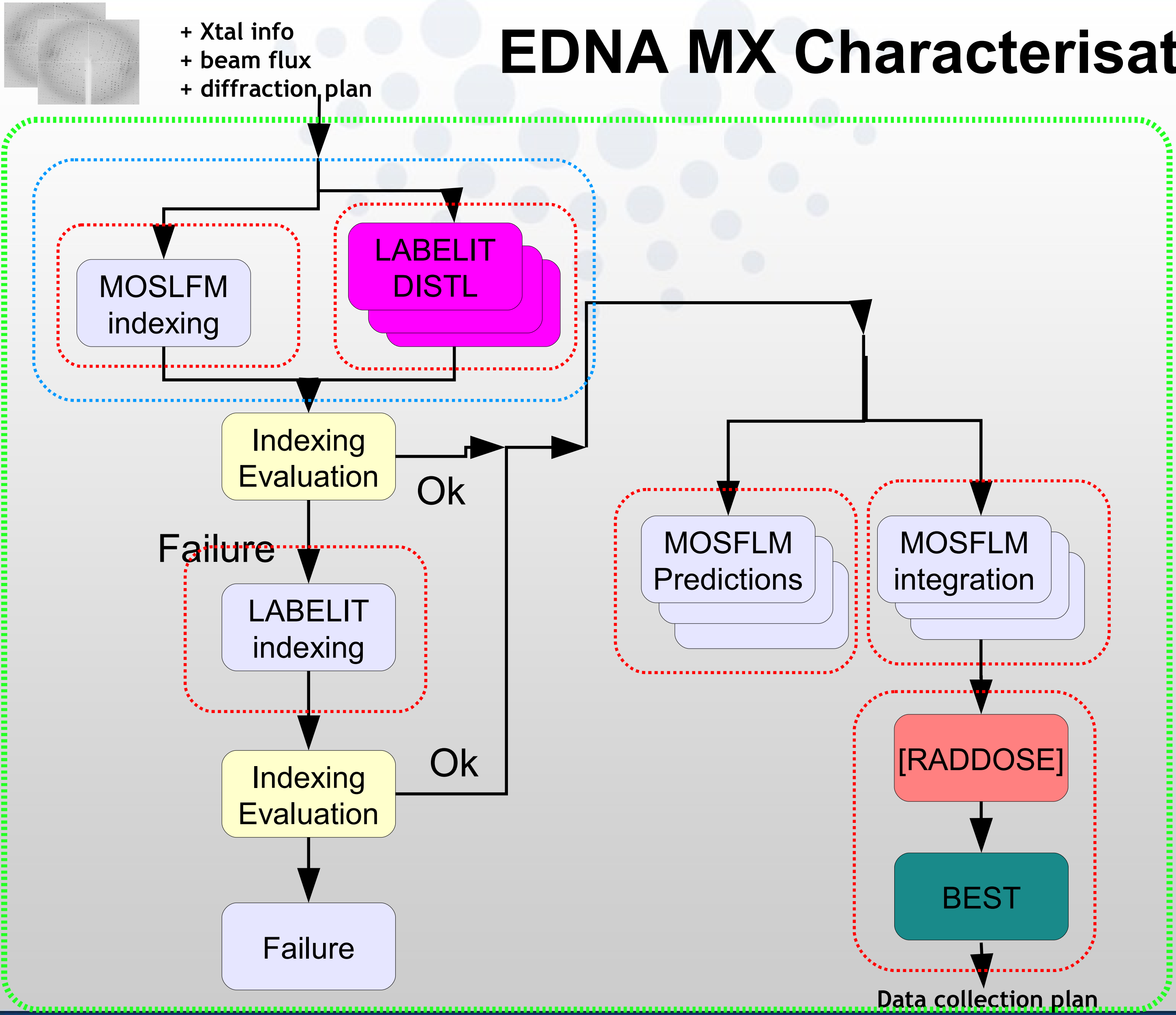
Scientific EDNA workflows @ ESRF

- Macromolecular crystallography:
  - Characterisation taking into account radiation damage (MOSFLM, Labelit,RADDOSE, BEST)
  - Connection with experiment data base (ISPyB)
  - Parallel execution of characterisation (GRID data processing)
  - Parallel creation of image thumbnails
- Diffraction Computed Tomography
  - SPD: Image correction, fast azimuthal integration
  - Sinograms saved in HDF5 format
- Small Angle Scattering
  - Image correction and fast azimuthal integration
- Full Field XAS
  - Image correction (dark, flat)
  - Image alignment (offset measurements by FFT)
  - HDF5 output

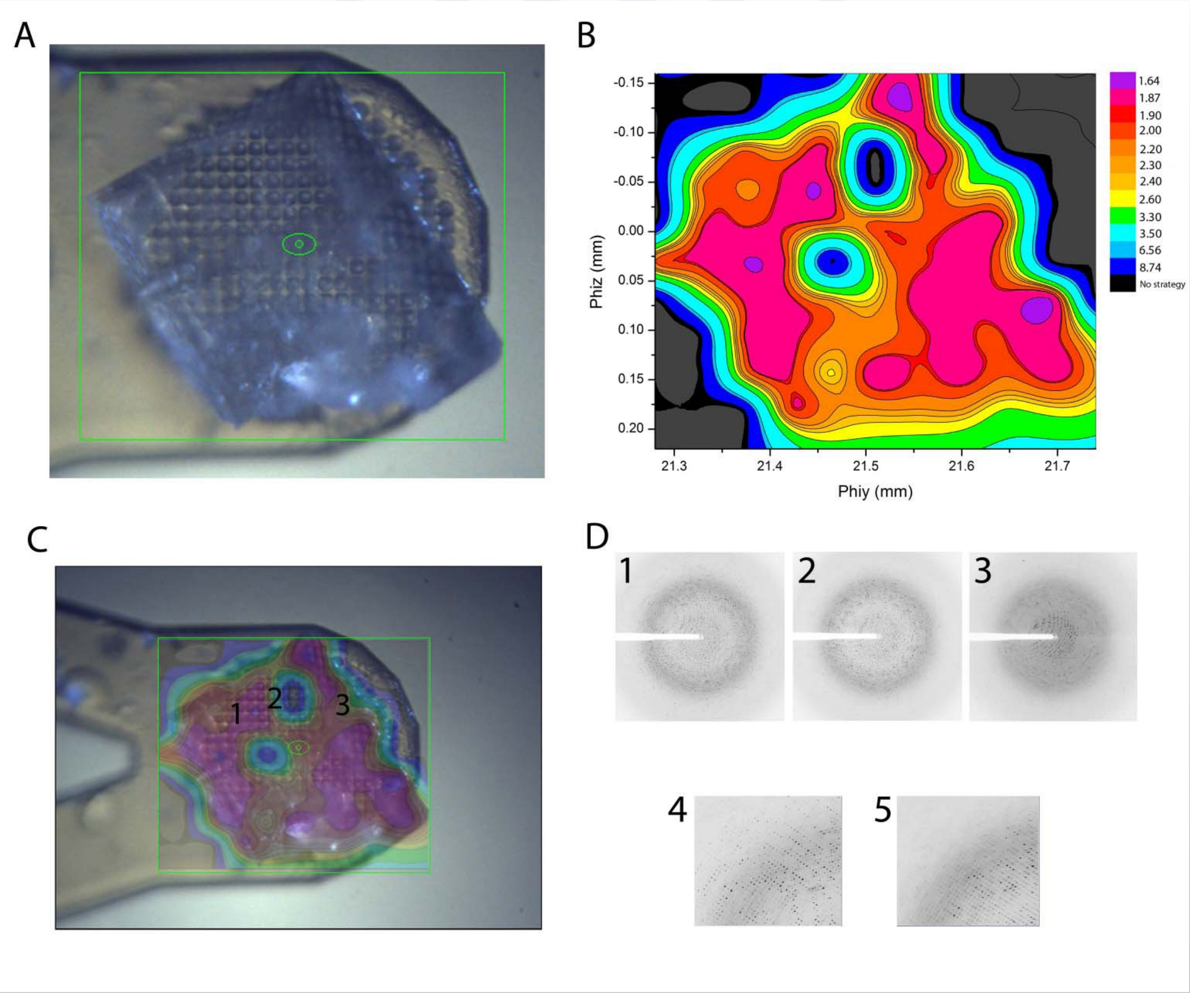
Challenge for the ESRF Upgrade : Massively Automated Sample Selection Integrated Facility


From dewar storage to screening
Baskets from Lab DS
Loaded on screening carrousel
Manual transfer to Data collection station(s)
Data collection station(s)
From sorting to Data collection
Screening station(s)
From screening to sorting
From Data collection to sorting/cleaning
Samples taken from Lab baskets
& put back after screening
From sorting to dear storage
Sorting/Cleaning station

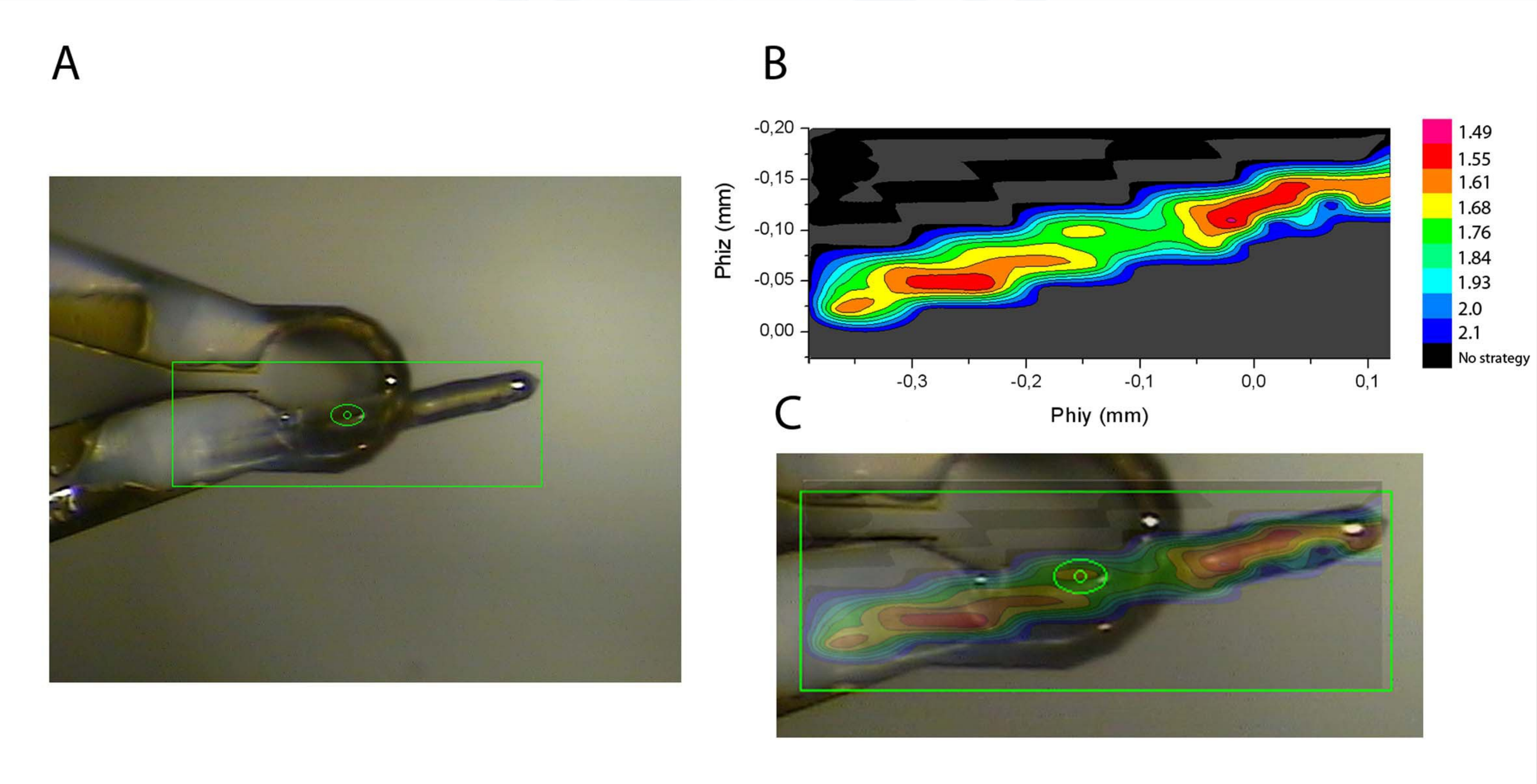
EDNA MX Characterisation



MX Grid Scans I

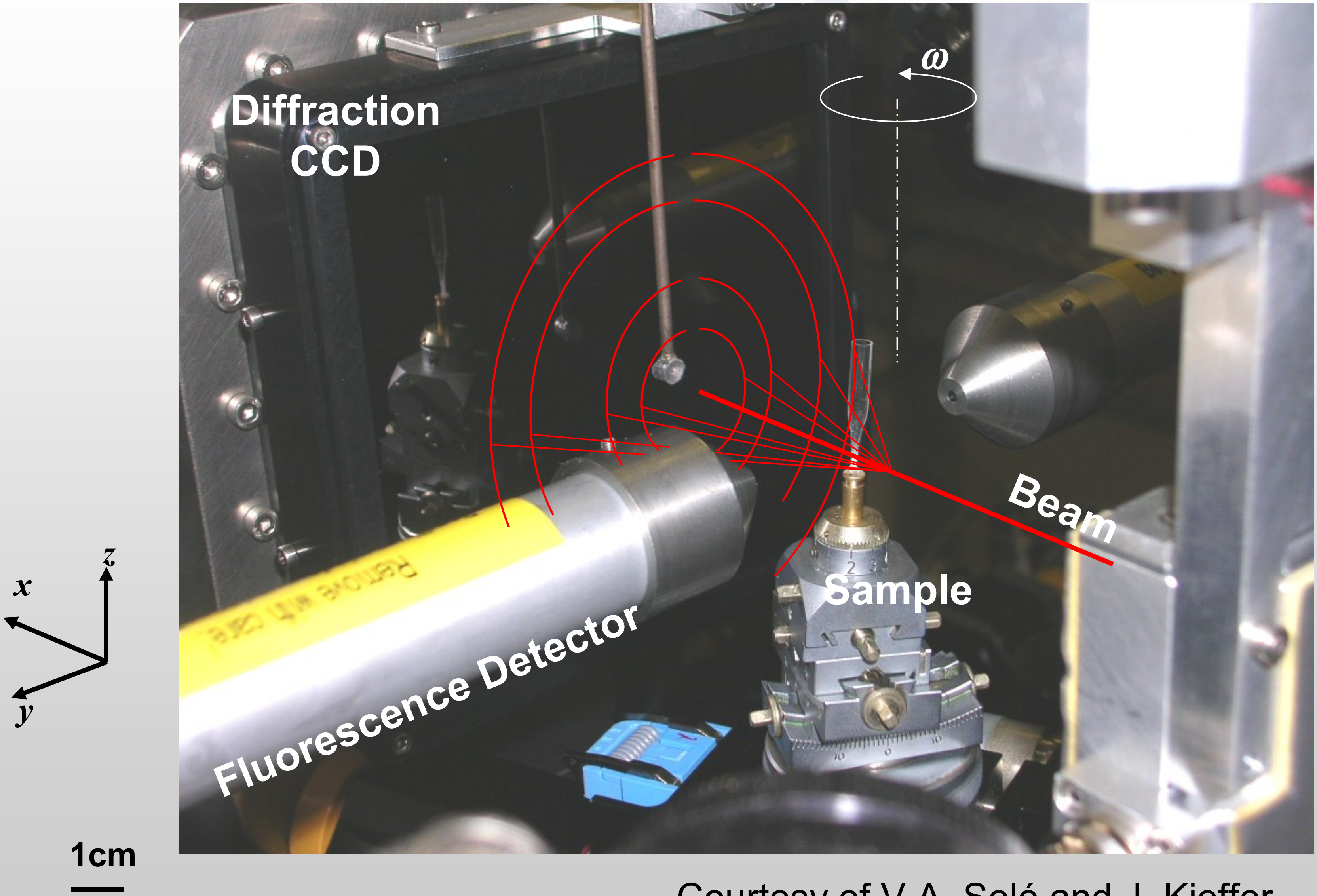


MX Grid Scans II



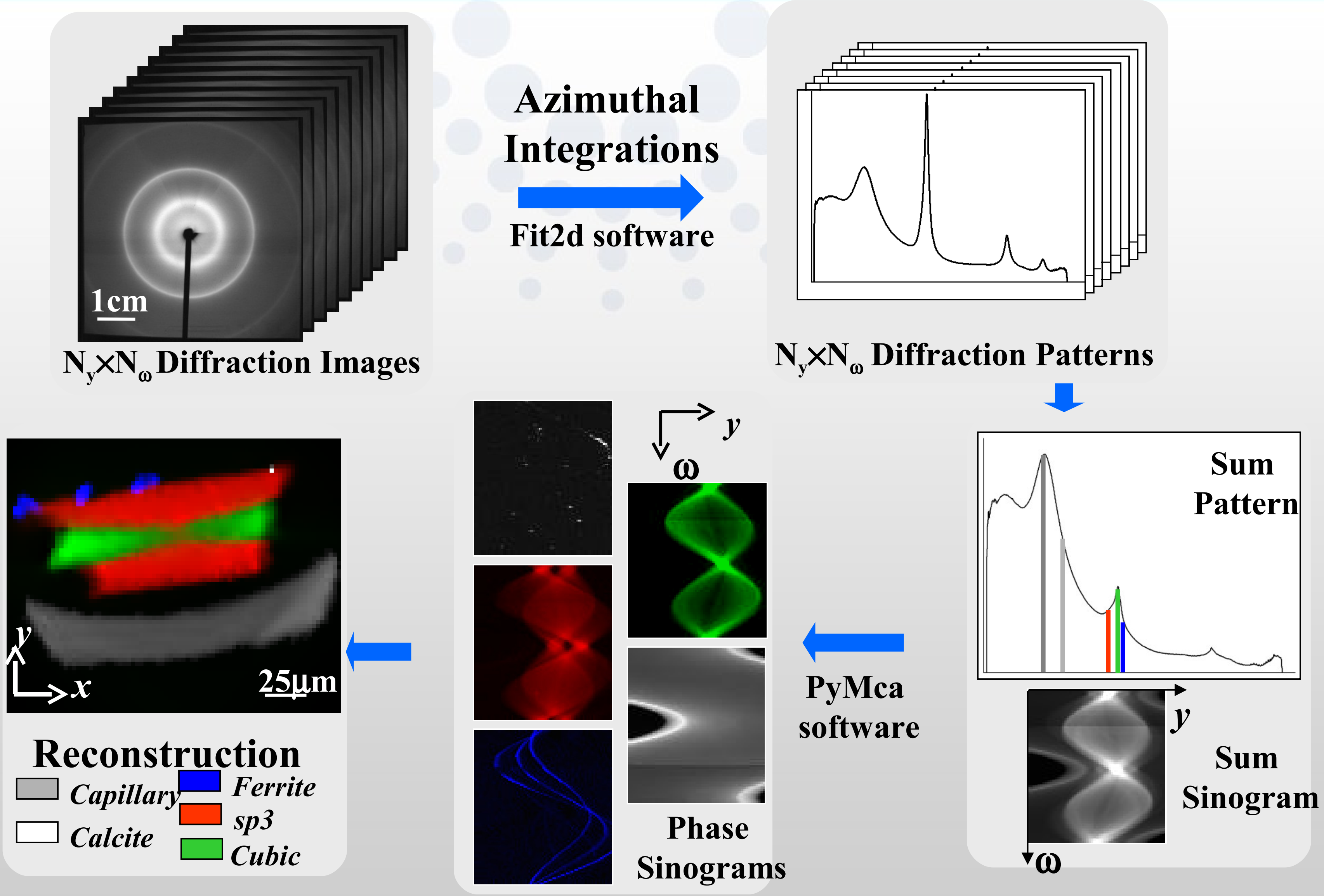
Bowler M.W. , Guijarro M. , Petitdemange S. , Baker I. , Svensson O. , Burghammer M. , Mueller-Dieckmann C. , Gordon E.J. , Flot D. , McSweeney S.M. , Leonard G.A. - Diffraction cartography: Applying microbeams to macromolecular crystallography sample evaluation and data collection Acta Crystallographica D 66, 855-864 (2010)

ID22 – Fluorescence-Diffraction Tomography



Courtesy of V.A. Solé and J. Kieffer

Original idea: 20h data acquisition, 30h treatment



$N_y \times N_\omega$ Diffraction Images

Azimuthal Integrations

Fit2d software

$N_y \times N_\omega$ Diffraction Patterns

Sum Pattern

Sum Sinogram

PyMca software

Phase Sinograms

Reconstruction

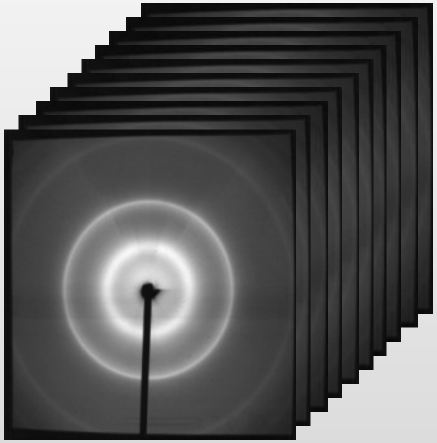
▨ *Capillary*
□ *Calcite*
🟦 *Ferrite*
🟥 *sp3*
🟩 *Cubic*

Acknowledgements: Pierre Bleuet CEA - Grenoble

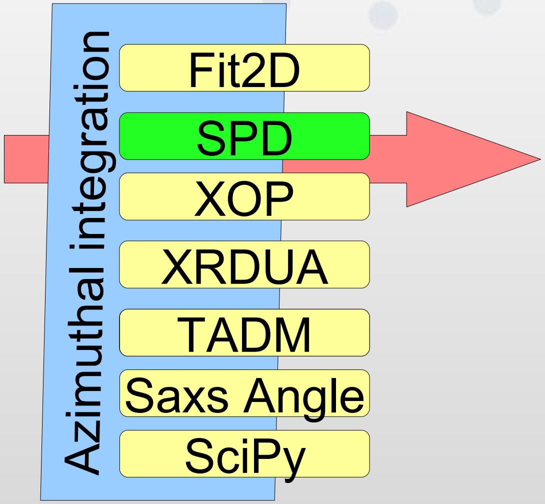
Online data Analysis for NINA

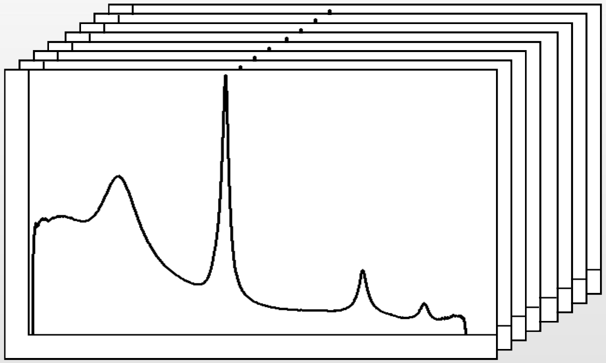
- Reduce each image (2D) to a spectrum (1D)
- Store a 2D mapping of spectra

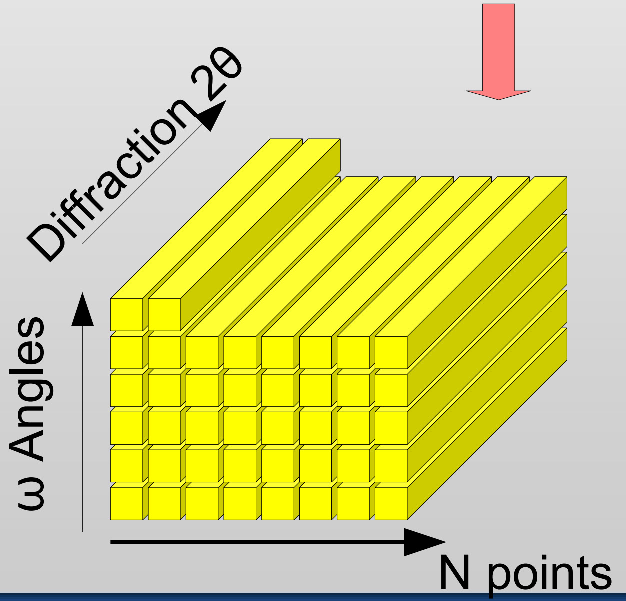
N x **ω** 1D diffraction patterns



N x **ω** 2D
Diffraction
Images







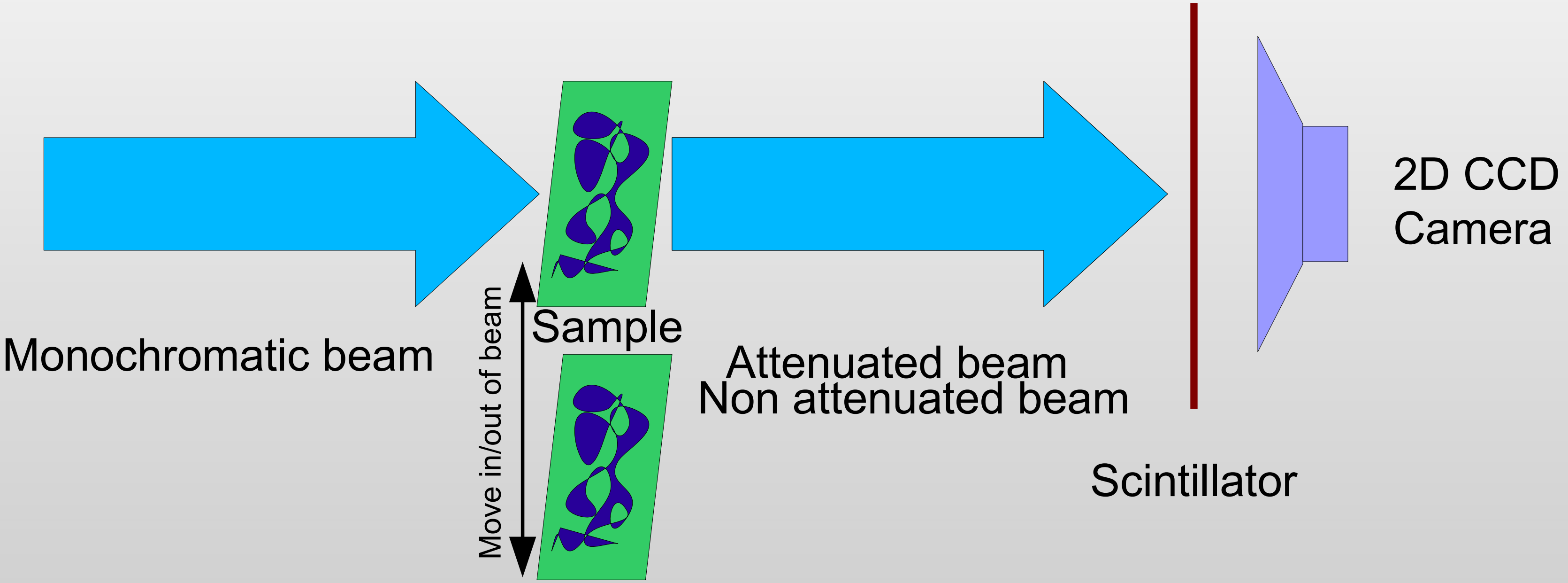
Performance: 4 images / s, full treatment in 40 min.

Courtesy of J. Kieffer

FullField XAS

Scan in Energy on monochromator
around absorption edge of a given element

Sample size: 1 mm x 1 mm x 1 µm
Resolution: 500 nm

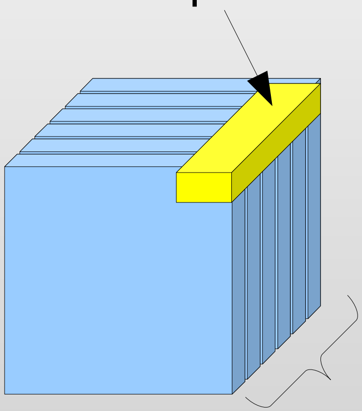


- Measure a (couple of) flats at each energy to correct for scintillator's response
- Align the sample at each position to correct sub-micron position change

Courtesy of J. Kieffer

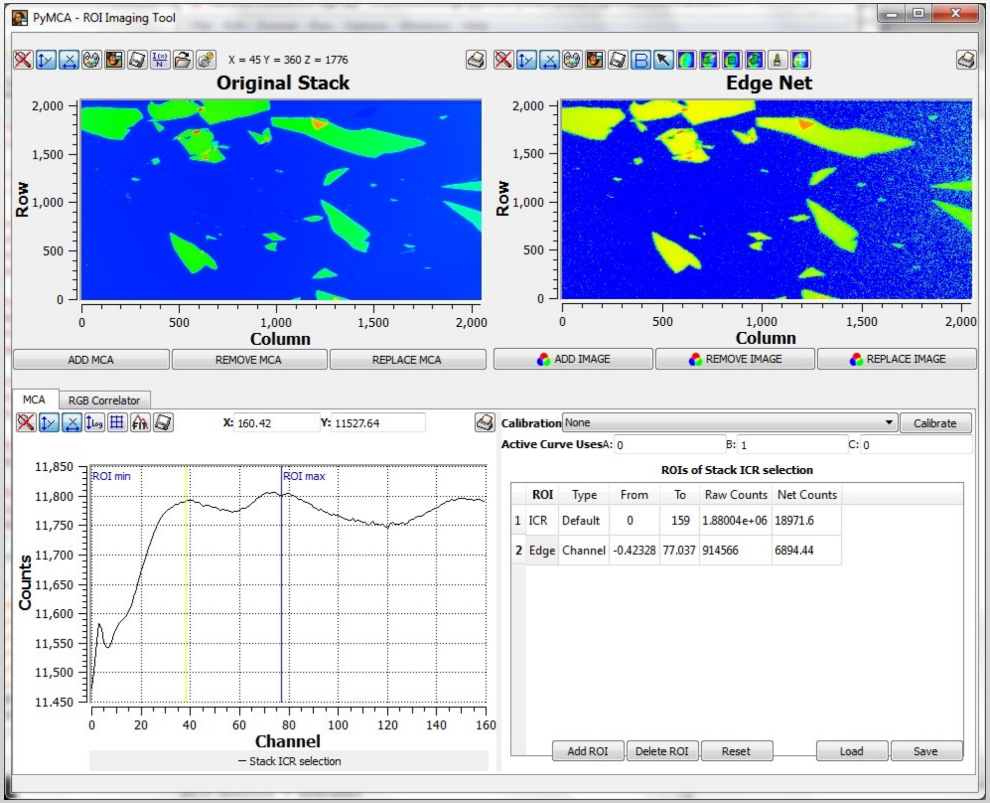
FullField XAS Results

- Tested online & offline in December 2010 (ID21)
- Image Alignment by FFT OK (→ ID10)
- Nexus compliance



Xanes Spectra

Stack of images



PyMCA - ROI Imaging Tool

X = 45 Y = 360 Z = 1776

X: 160.42 Y: 11527.64

Calibration: None

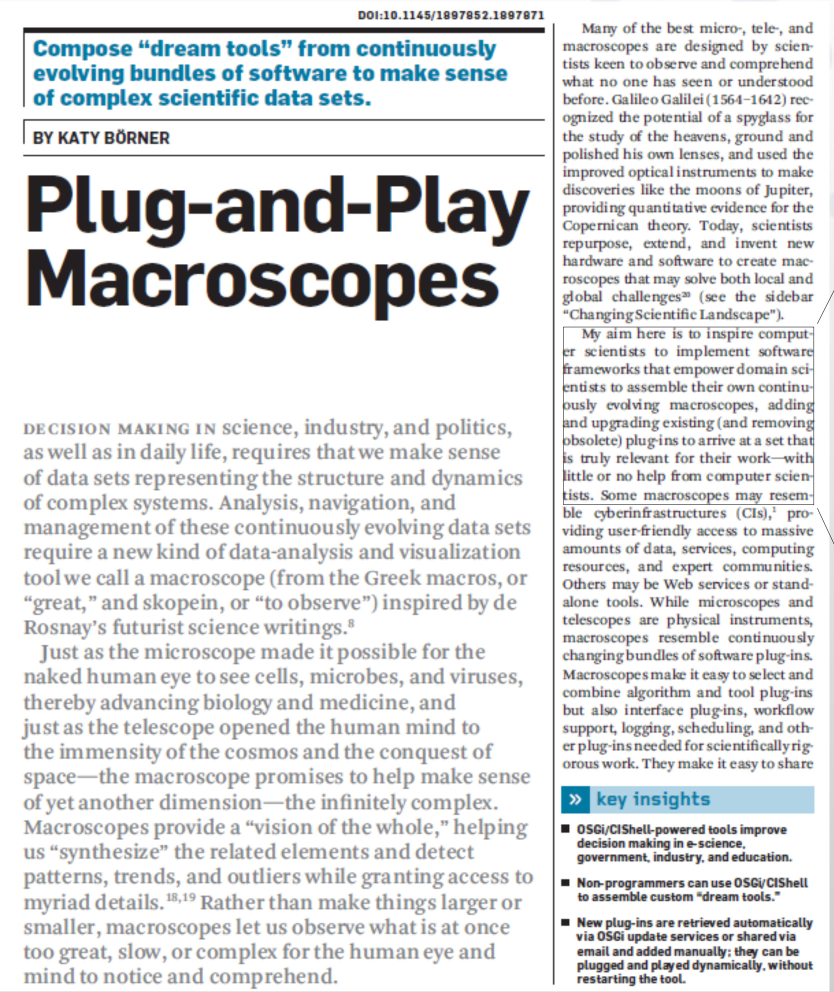
Active Curve Uses A: 0 B: 1 C: 0

ROIs of Stack ICR selection

	ROI	Type	From	To	Raw Counts	Net Counts
1	ICR	Default	0	159	1.88004e+06	18971.6
2	Edge	Channel	-0.42328	77.037	914566	6894.44

Courtesy of J. Kieffer and A. Sole

Workflow tools



DOI:10.1145/1897852.1897871

Compose "dream tools" from continuously evolving bundles of software to make sense of complex scientific data sets.

BY KATY BÖRNER

Plug-and-Play Macroscopes

DECISION MAKING IN science, industry, and politics, as well as in daily life, requires that we make sense of data sets representing the structure and dynamics of complex systems. Analysis, navigation, and management of these continuously evolving data sets require a new kind of data-analysis and visualization tool we call a macroscope (from the Greek macros, or "great," and skopein, or "to observe") inspired by de Rosnay's futurist science writings.[8]

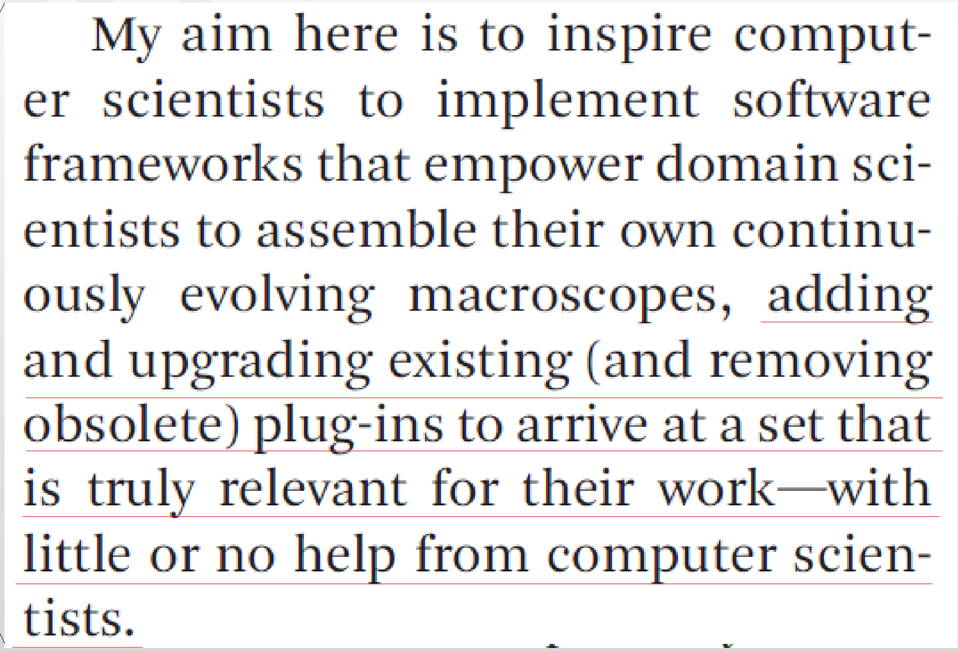
Just as the microscope made it possible for the naked human eye to see cells, microbes, and viruses, thereby advancing biology and medicine, and just as the telescope opened the human mind to the immensity of the cosmos and the conquest of space—the macroscope promises to help make sense of yet another dimension—the infinitely complex. Macroscopes provide a "vision of the whole," helping us "synthesize" the related elements and detect patterns, trends, and outliers while granting access to myriad details.[18,19] Rather than make things larger or smaller, macroscopes let us observe what is at once too great, slow, or complex for the human eye and mind to notice and comprehend.

Many of the best micro-, tele-, and macroscopes are designed by scientists keen to observe and comprehend what no one has seen or understood before. Galileo Galilei (1564–1642) recognized the potential of a spyglass for the study of the heavens, ground and polished his own lenses, and used the improved optical instruments to make discoveries like the moons of Jupiter, providing quantitative evidence for the Copernican theory. Today, scientists repurpose, extend, and invent new hardware and software to create macroscopes that may solve both local and global challenges[20] (see the sidebar "Changing Scientific Landscape").

My aim here is to inspire computer scientists to implement software frameworks that empower domain scientists to assemble their own continuously evolving macroscopes, adding and upgrading existing (and removing obsolete) plug-ins to arrive at a set that is truly relevant for their work—with little or no help from computer scientists. Some macroscopes may resemble cyberinfrastructures (CIs),[1] providing user-friendly access to massive amounts of data, services, computing resources, and expert communities. Others may be Web services or standalone tools. While microscopes and telescopes are physical instruments, macroscopes resemble continuously changing bundles of software plug-ins. Macroscopes make it easy to select and combine algorithm and tool plug-ins but also interface plug-ins, workflow support, logging, scheduling, and other plug-ins needed for scientifically rigorous work. They make it easy to share

» key insights

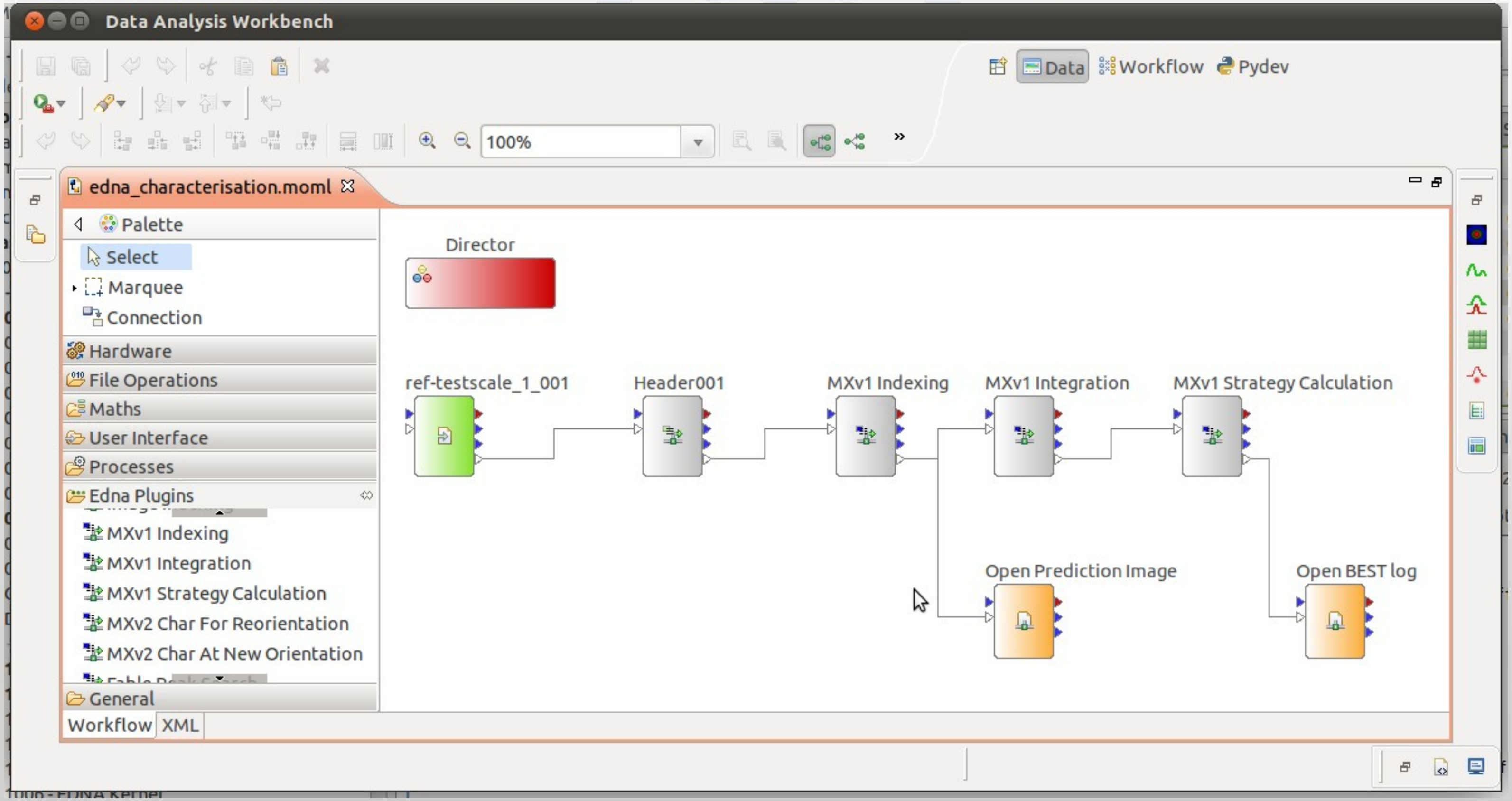
- **OSGi/CIShell-powered tools improve decision making in e-science, government, industry, and education.**
- **Non-programmers can use OSGi/CIShell to assemble custom "dream tools."**
- **New plug-ins are retrieved automatically via OSGi update services or shared via email and added manually; they can be plugged and played dynamically, without restarting the tool.**



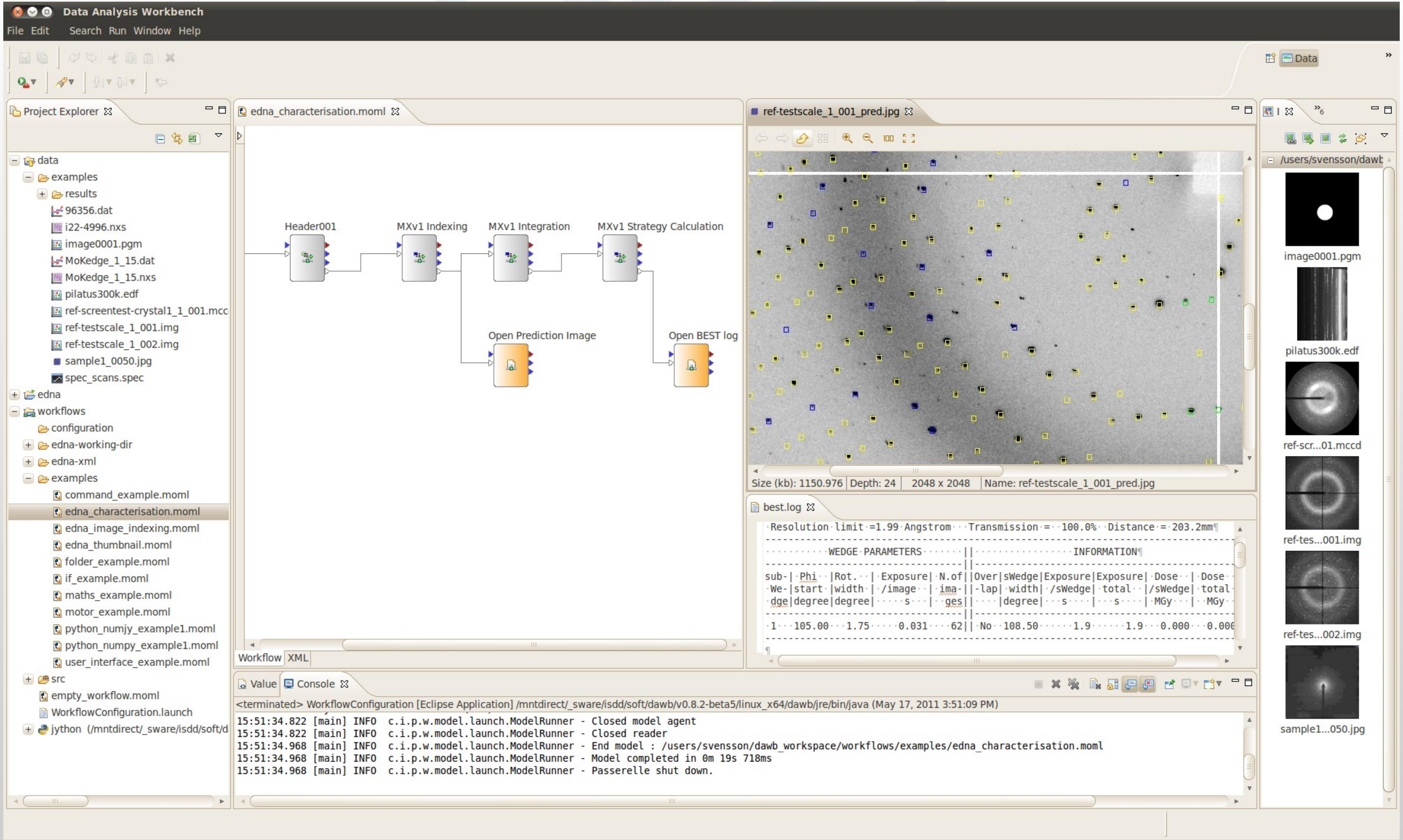
My aim here is to inspire computer scientists to implement software frameworks that empower domain scientists to assemble their own continuously evolving macroscopes, adding and upgrading existing (and removing obsolete) plug-ins to arrive at a set that is truly relevant for their work—with little or no help from computer scientists.

http://cacm.acm.org/magazines/2011/3/105316-plug-and-play-macroscopes/fulltext

EDNA Workflow Tool - Example



Data Analysis WorkBench - DAWB



best.log

```
 Resolution limit =1.99 Angstrom   Transmission =  100.0%  Distance = 203.2mm
-----------------------------------------------------------------------------------
          WEDGE PARAMETERS       ||                INFORMATION
---------------------------------||------------------------------------------------
sub-| Phi  |Rot.  | Exposure| N.of||Over|sWedge|Exposure|Exposure| Dose  | Dose
 We-|start |width | /image  | ima-||-lap| width| /sWedge| total  |/sWedge| total
 dge|degree|degree|    s    |  ges||    |degree|   s    |   s    | MGy   |  MGy
---------------------------------||------------------------------------------------
 1   105.00   1.75     0.031    62|| No  108.50      1.9      1.9   0.000   0.000
-----------------------------------------------------------------------------------
```

<terminated> WorkflowConfiguration [Eclipse Application] /mntdirect/_sware/isdd/soft/dawb/v0.8.2-beta5/linux_x64/dawb/jre/bin/java (May 17, 2011 3:51:09 PM)

```
15:51:34.822 [main] INFO  c.i.p.w.model.launch.ModelRunner - Closed model agent
15:51:34.822 [main] INFO  c.i.p.w.model.launch.ModelRunner - Closed reader
15:51:34.968 [main] INFO  c.i.p.w.model.launch.ModelRunner - End model : /users/svensson/dawb_workspace/workflows/examples/edna_characterisation.moml
15:51:34.968 [main] INFO  c.i.p.w.model.launch.ModelRunner - Model completed in 0m 19s 718ms
15:51:34.968 [main] INFO  c.i.p.w.model.launch.ModelRunner - Passerelle shut down.
```

http://www.dawb.org

Acknowledgements

ESRF

ISDD Software group

- Andy Götz

DAU

- Claudio Ferrero
- Jérôme Kieffer
- Matthew Gerring
- Armando Solé
- Rainer Wilcke

BCU

- Gilles Berruyer
- Matias Guijarro
- Antonia Beteva
- Cyril Guilloud
- Darren Spruce

Exp div.

- Alexander Popov (MX)
- Matthew Bowler (MX)
- Sean McSweeney (MX)
- Gordon Leonard (MX)
- Elspeth Gordon (MXPress)
- Stephanie Monaco (MXPress)
- Ricardo Leal (MX)
- Peter Boesecke (ID2)
- Jon Wright (ID11)
- Andy Kink (ID11)
- Manfred Burghammer (ID13)
- Louiza Zerad (ID14-3)
- Adam Round (ID14-3)
- Alexander Rack (ID19)
- Barbara Fayard (ID21)
- Muriel Salome (ID21)
- Remy Tucoulou (ID22)

EDNA Collaborators

- Alun Ashton (DLS)
- Karl Levik (DLS)
- Mark Basham (DLS)
- Graeme Winter (DLS)
- Ronan Keegan (DLS)
- Peter Chang (DLS)
- Sandor Brockhauser (EMBL G)
- Andrew McCarthy (EMBL G)
- Gleb Bourenkov (EMBL H)
- Thomas Schneider (EMBL H)
- Al Kikney (EMBL H)
- Andrew Leslie (MRC LMB)
- Harry Powell (MRC LMB)
- Peter Keller (Global Phasing)
- Gerard Bricogne (Global Phasing)
- Owe Muller (BESSY)
- Michel Hellmig (BESSY)
- Erwin de Ley (Isencia)
- Karsten Thoms (Itemis)