Crystallographic structure refinement in PHENIX

Pavel Afonine

Computation Crystallography Initiative
Physical Biosciences Division
Lawrence Berkeley National Laboratory

CCP4 workshop, May 22-28, 2008

Outline

- PHENIX software

- Crystallographic structure refinement – brief overview

- Introduction to *phenix.refine* (structure refinement part of PHENIX)

What is PHENIX?

- PHENIX = Python-based Hierarchical ENvironment for Integrated Xtallography
- Actively developed package for automated structure solution
- Solid background:
  - Xplor / CNS:
- New approaches:
  - Modern programming concepts (Python, C++) and new algorithms
  - Modularization: accelerated development through re-use
  - Integration: combination of heterogeneous algorithms
- Designed to be used by both novices and experienced users
- Long-term development and support

Who is PHENIX?

Collaboration between several groups:

- Los Alamos National Lab
  Tom Terwilliger, Li-Wei Hung (SOLVE / RESOLVE, Ligandfit, Autobuild …)
  Paul Langan, Marat Mustyakimov, Benno Schoenborn (Tools for Neutron crystallography) (separate funding, MNC)

- Cambridge University, UK
  Randy Read, Airlie McCoy, Laurent Storoni (PHASER)

- Duke University
  Jane & David Richardson, Ian Davis, Vincent Chen (MolProbity, hydrogens)

- Lawrence Berkeley National Lab
  Paul Adams, Pavel Afonine, Ralf Grosse-Kunstleve, Nigel Moriarty, Nicholas Sauter, Peter Zwart (CCI Apps: phenix.refine, phenix.elbow, phenix.xtriage,…)

- Texas A&M University
  Tom Ioerger, Jim Sacchettini, Erik McKee (TEXTAL)

Paul Adams – project director

PHENIX: what’s inside?

▪ Solve, Resolve	model building, density modifications and more
▪ Ligandfit	build ligands into density
▪ Autobuild	Solve/Resolve + phenix.refine = from starting phases to complete and refined model
▪ AutoMR	Phaser + Autobuild = refined model
▪ phenix.refine	structure refinement
▪ phenix.elbow	build library files (cif) for ligands
▪ phenix.xtriage	comprehensive data analysis
▪ phenix.pdbtools	set of tools for PDB file manipulation
▪ phenix.hyss	substructure solution

… many other

What is phenix.refine ?

phenix.refine

- Highly-automated state-of-the-art structure refinement part of PHENIX

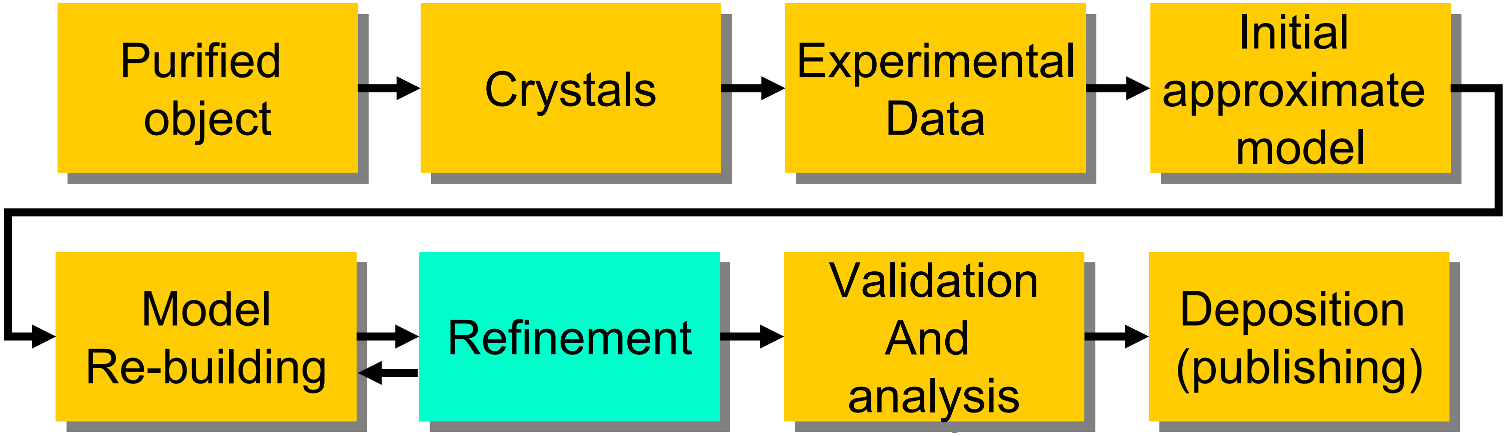
- Under active development by *Paul Adams, Pavel Afonine, Ralf Grosse-Kunstleve, Nigel Moriarty, Peter Zwart*

- Works everywhere (Linux, Mac, Windows)

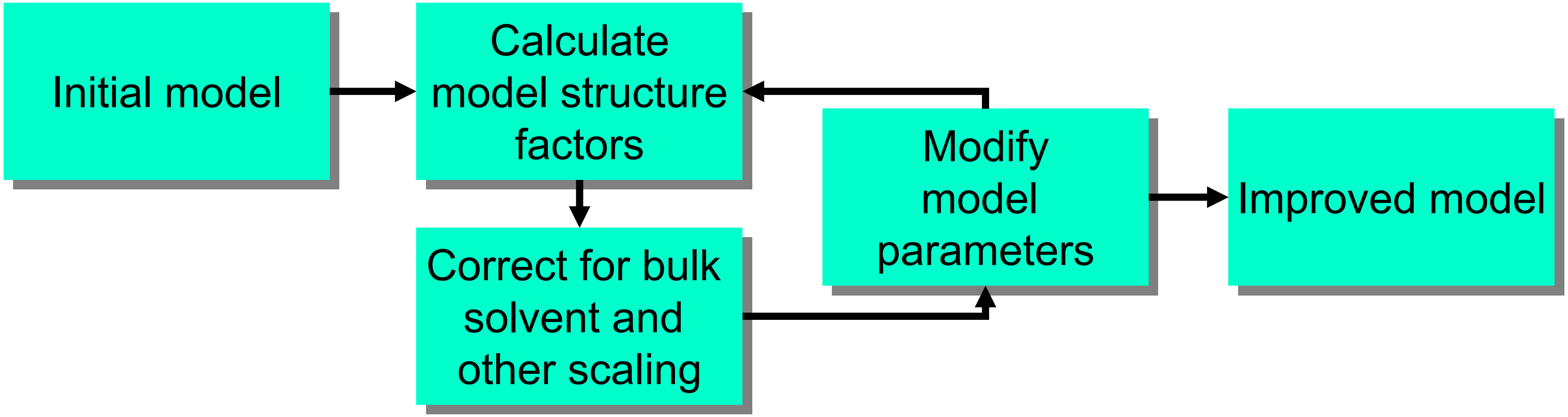
- “One click” installation

Structure refinement

- **Structure determination work-flow**

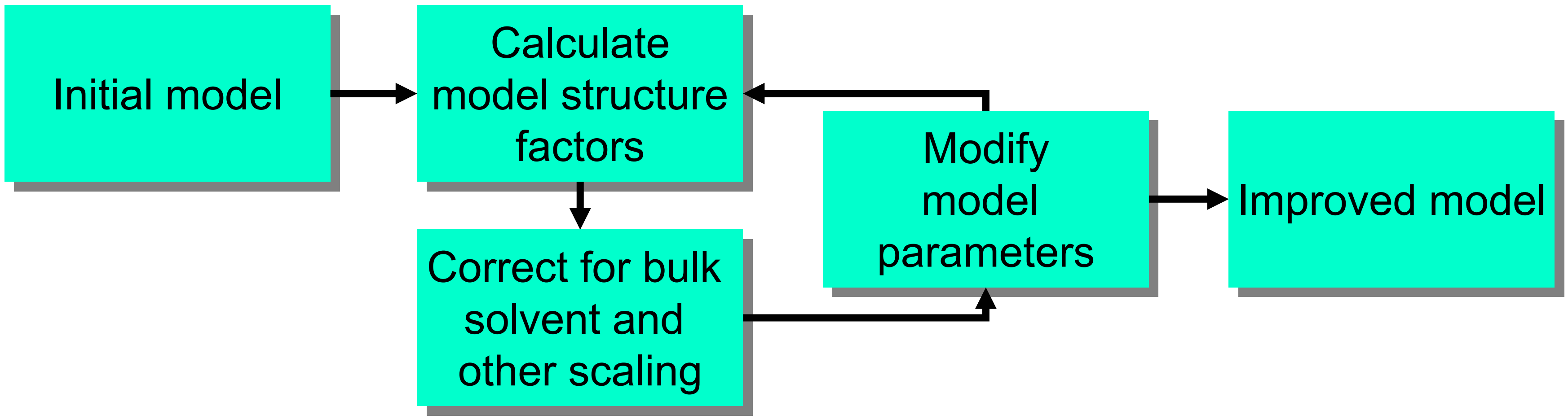


- **Structure refinement**


Initial model
Calculate model structure factors
Correct for bulk solvent and other scaling
Modify model parameters
Improved model

Structure refinement

- **Structure refinement**


Initial model
Calculate model structure factors
Correct for bulk solvent and other scaling
Modify model parameters
Improved model

=

- **Structure refinement:** vary ***model parameters*** in order to optimize a goal (target) function:

$$E_{\mathrm{TOTAL}} = E_{\mathrm{DATA}} + wE_{\mathrm{RESTRAINTS}}$$

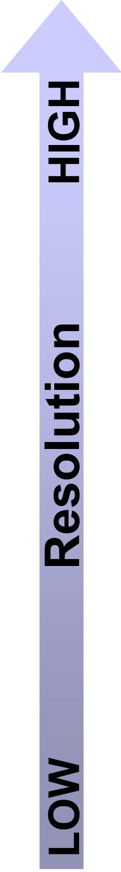
E_{DATA} – a function that relates a model to experimental data.

$E_{\mathrm{RESTRAINTS}}$ – an a priori knowledge that may be introduced to compensate for the lack of experimental data (finite resolution) (and to improve the data-to-parameters ratio).

(Atomic) Model parameters

- **Choice for model parameterization is a function of experimental data quality**

Higher data resolution – More information – More detailed model parameterization



Subatomic (< 0.9Å): xyz (3), ADP (6), occupancy (1), multipolar or IAS ~ 20-30

High (0.9-1.6Å): xyz (3), ADP (6), occupancy (1) = 10

Medium (1.6-3.0Å): xyz (3), ADP (1), occupancy (1) = 5

Low (2.8-4.0Å): xyz (3 for individual or 0.3 for torsion angles), ADP (1 for individual or 1 per group), occupancy (1)

Very low: Rigid body (6 parameters per group), TLS (20 parameters per group), group isotropic B (1 parameter per selected group of atoms)

Refinement target function

- **Structure refinement:** vary model parameters in order to optimize a goal (target) function:

$$E_{\mathrm{TOTAL}} = E_{\mathrm{DATA}} + wE_{\mathrm{RESTRAINTS}}$$

Optimization algorithms:

- gradient-driven minimization
- simulated annealing

E$_{\mathrm{DATA}}$ – “X-ray target” (or Neutron), a function that relates a model to experimental data

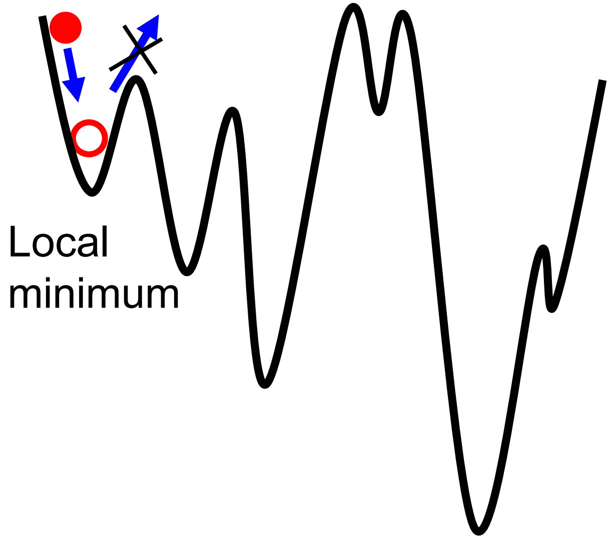
E$_{\mathrm{RESTRAINTS}}$ – a priori knowledge that may be introduced to compensate for the lack of experimental data (finite resolution) and to improve the data-to-parameters ratio.

Refinement target optimization

▪ Minimization

- Follows the local gradient

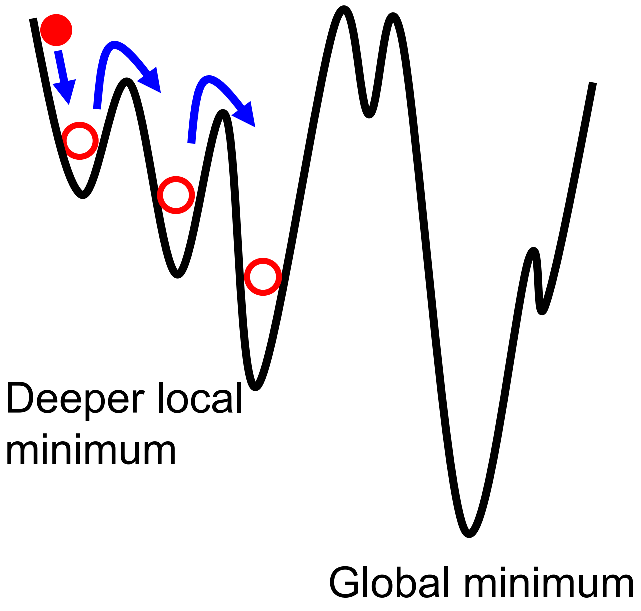
- The target function depends on many parameters - many local minima in addition to the global minimum.



▪ Simulated annealing (SA)

- Optimization method which is good at escaping local minima.

  - Increased probability of finding a better solution because motion against the gradient is allowed.
  - Probability of uphill motion is determined by the temperature.



E_{DATA} : X-ray target

$$E_{\text{TOTAL}} = E_{\text{DATA}} + wE_{\text{RESTRAINTS}}$$

- **Least-Squares function**

$$E_{\text{DATA}} = \sum_{\mathbf{s}} \mathbf{w}_s \left(F_{\mathbf{s}}^{\text{CALC}} - kF_{\mathbf{s}}^{\text{OBS}}\right)^{\mathbf{2}}$$

- Widely used in small molecule crystallography

- Used in macromolecular crystallography in the past

- **Better choice: Maximum-Likelihood target**

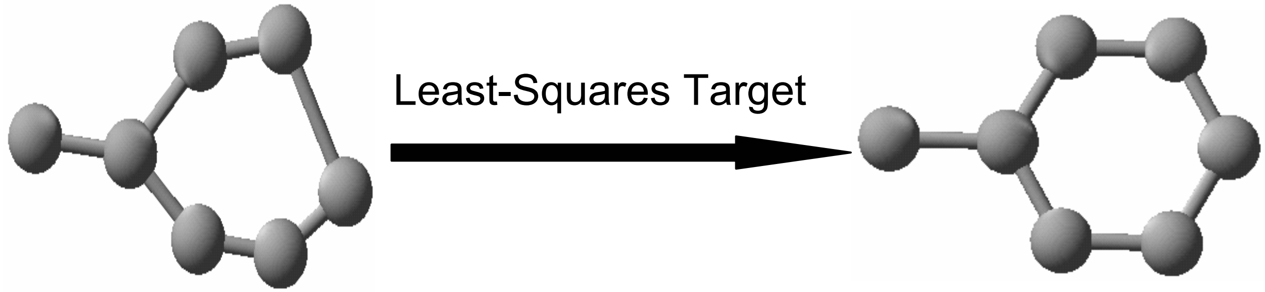
$$E_{\text{DATA}} = \sum_{\text{s}} \; (1 - K_{\mathbf{s}}^{cs}) \left(-\frac{\alpha_{\mathbf{s}}^2 \left(F_{\mathbf{s}}^{CALC}\right)^2}{\varepsilon_{\mathbf{s}} \beta_{\mathbf{s}}} + ln\left(I_0 \left(\frac{2\alpha_{\mathbf{s}} F_{\mathbf{s}}^{CALC} F_{\mathbf{s}}^{OBS}}{\varepsilon_{\mathbf{s}} \beta_{\mathbf{s}}} \right) \right) \right) +$$

$$+ K_{\mathbf{s}}^{cs} \left(-\frac{\alpha_{\mathbf{s}}^2 \left(F_{\mathbf{s}}^{CALC}\right)^2}{2\varepsilon_{\mathbf{s}} \beta_{\mathbf{s}}} + ln\left(cosh\left(\frac{\alpha_{\mathbf{s}} F_{\mathbf{s}}^{CALC} F_{\mathbf{s}}^{OBS}}{\varepsilon_{\mathbf{s}} \beta_{\mathbf{s}}} \right) \right) \right)$$

E_{DATA}: Why Maximum-Likelihood?

- **Removable Errors** (never the case for macromolecular model, common for small molecules)

Complete model *before* refinement

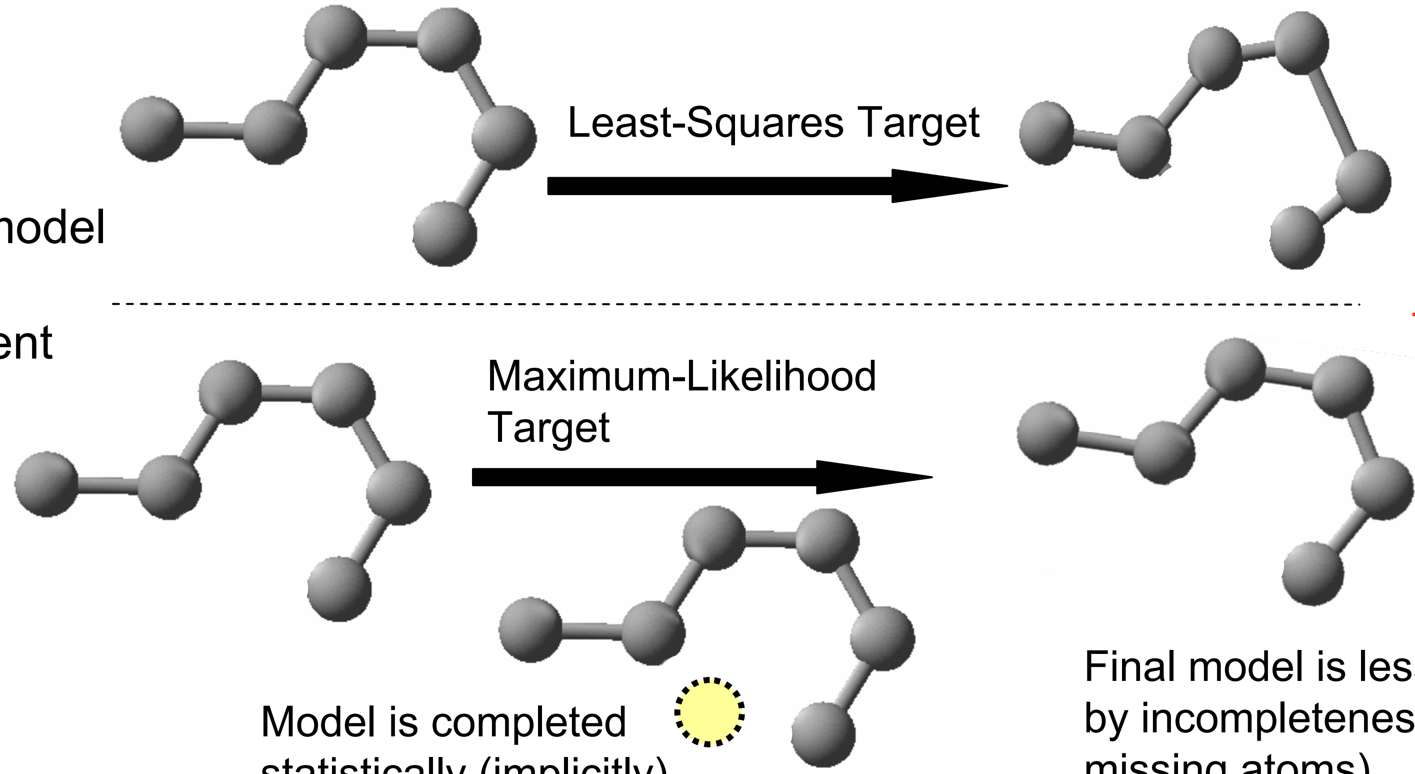


Least-Squares Target

Complete model *after* refinement

- **Irremovable Errors** (always the case for macromolecular models)

Partial model *before* refinement



Least-Squares Target

Maximum-Likelihood Target

Partial model *after* refinement

Model is completed statistically (implicitly)

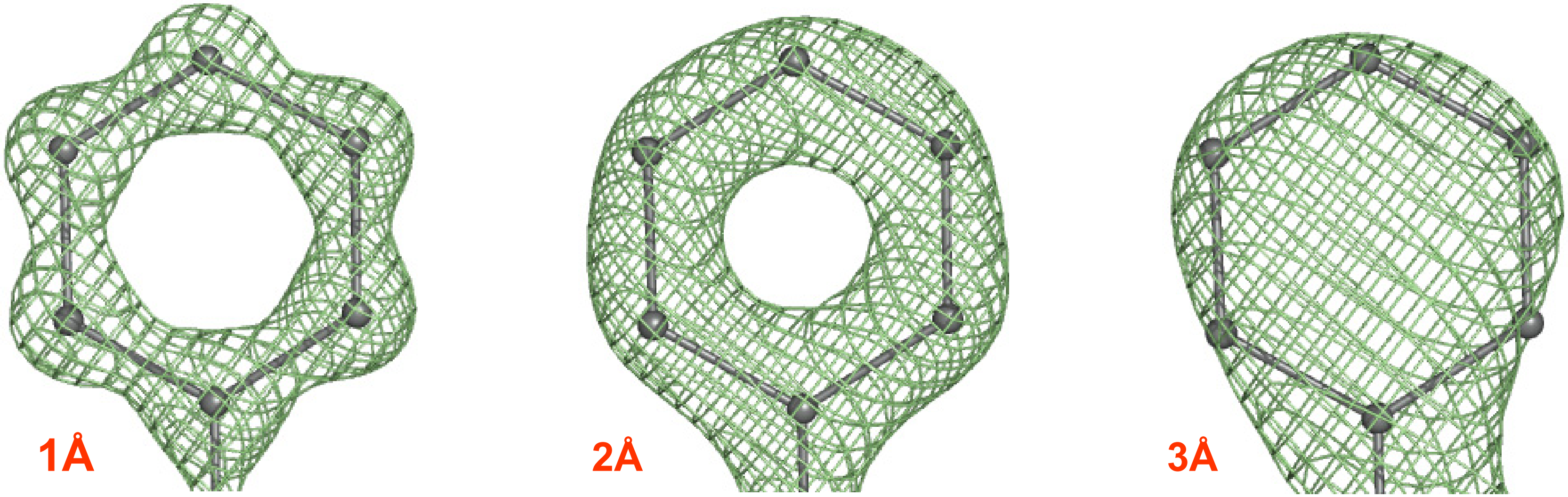
Final model is less affected by incompleteness (by missing atoms)

Restraints

$$E_{\mathrm{TOTAL}} = E_{\mathrm{DATA}} + wE_{\mathrm{RESTRAINTS}}$$

- **Refinement of individual coordinates**

Fourier images at different data resolution



→ *A priori* chemical knowledge is introduced (restraints) to keep the model chemically correct while fitting it to the experimental data at lower resolution (less resolution, stronger the weight *W*):

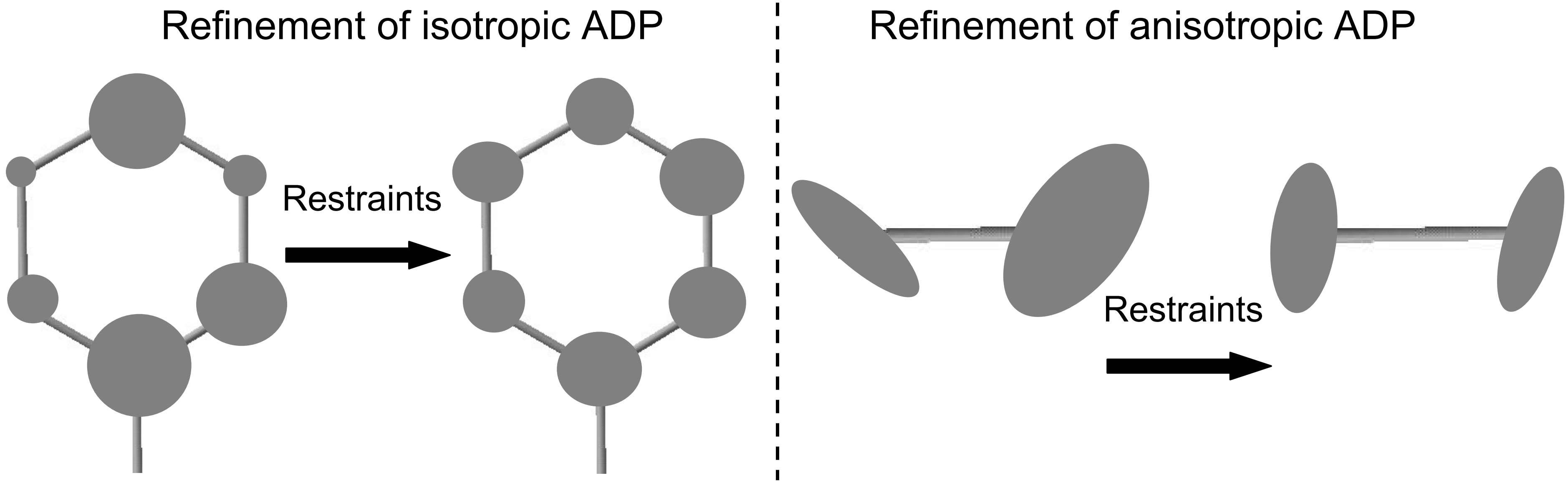
$$E_{\mathrm{RESTRAINTS}} = E_{\mathrm{BOND}} + E_{\mathrm{ANGLE}} + E_{\mathrm{DIHEDRAL}} + E_{\mathrm{PLANARITY}} + E_{\mathrm{NONBONDED}} + \ldots$$

→ Higher resolution – less restraints contribution (can be completely unrestrained at subatomic resolution, higher than ~0.9 Å for well ordered parts)

Restraints

$$E_{\mathrm{TOTAL}} = E_{\mathrm{DATA}} + wE_{\mathrm{RESTRAINTS}}$$

- **Refinement of individual ADP (Atomic Displacement Parameters, B-factors)**



Refinement of isotropic ADP

Refinement of anisotropic ADP

Restraints target for individual isotropic ADP refinement

$$\mathbf{E}_{\mathrm{ADP}} = \sum_{\mathrm{i=1}}^{\mathrm{N_{atoms}}} \left[\sum_{\mathrm{j=1}}^{\mathrm{M_{atoms}}} \frac{\mathbf{1}}{\mathbf{r}_{\mathrm{ij}}^{\text{distance\_p ower}}} \left. \frac{\left(\mathbf{U}_{\mathrm{i}} - \mathbf{U}_{\mathrm{j}}\right)^2}{\left(\dfrac{\mathbf{U}_{\mathrm{i}} + \mathbf{U}_{\mathrm{j}}}{\mathbf{2}}\right)^{\text{average\_po wer}}} \right|_{\text{sphereR}} \right]$$

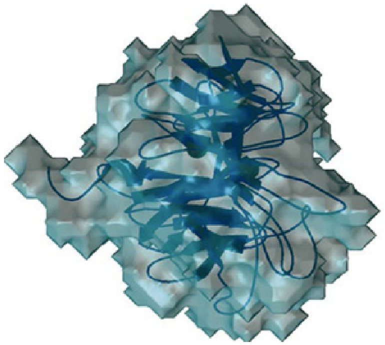
Refinement decisions

- **Parameterization:**
  - Coordinates: restraints *vs* constraints (Rigid body or its special case - Torsion angles)
  - ADP: aniso/isotropic, groups, individual, TLS
  - NCS: constrained, restrained, ignored
- **Optimization algorithm:**
  - Simulated annealing
  - Minimization (first or second derivatives methods)
- **Target function:**
  - Chemical information (chemical restraints, NCS similarity)
  - Maximum likelihood
  - Experimental phases

phenix.refine

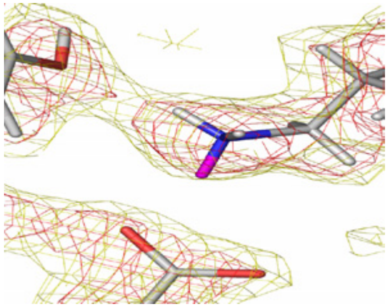
phenix.refine: single program for a very broad range of resolutions

Low



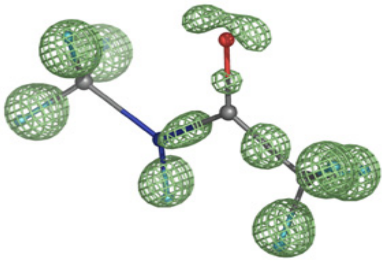
- Group ADP refinement
- Rigid body refinement
- Torsion Angle dynamics

Medium and High



- Restrained refinement (xyz, ADP: isotropic, anisotropic, mixed)
- Automatic water picking

Subatomic

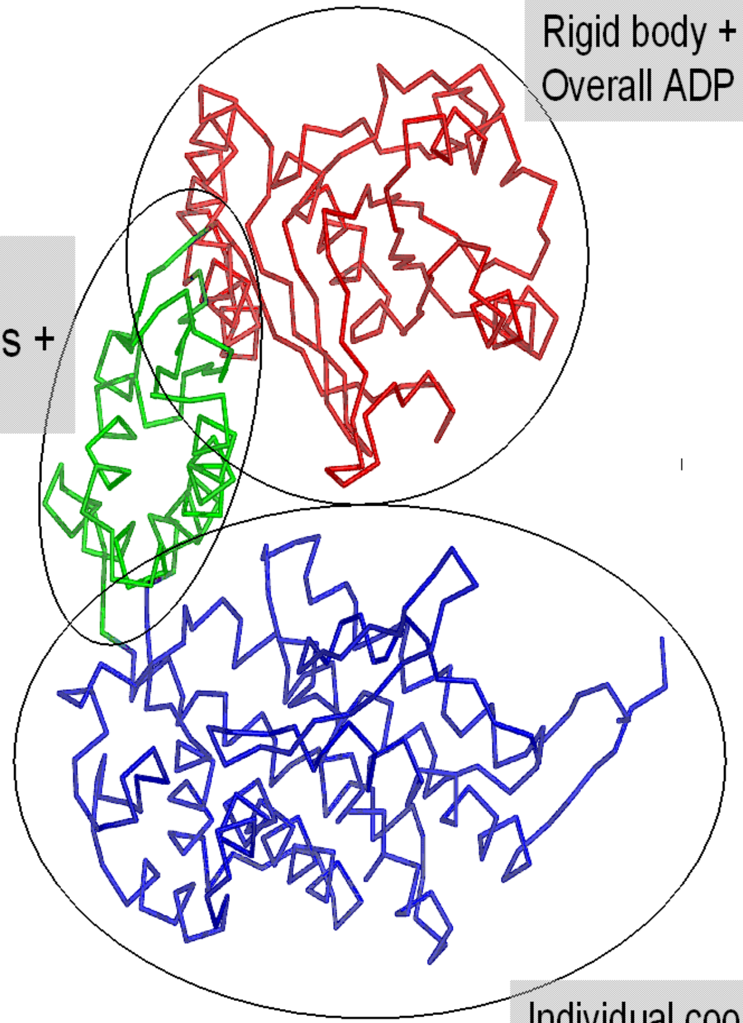


- Bond density model
- Unrestrained refinement
- FFT or direct
- Explicit hydrogens

- Automatic NCS restraints
- Simulated Annealing
- Occupancies (individual, group, automatic constrains for alternative conformations)

- TLS refinement
- Use hydrogens at any resolution
- Refinement with twinned data
- X-ray, Neutron, joint X-ray + Neutron
- Built-in water picking and refinement

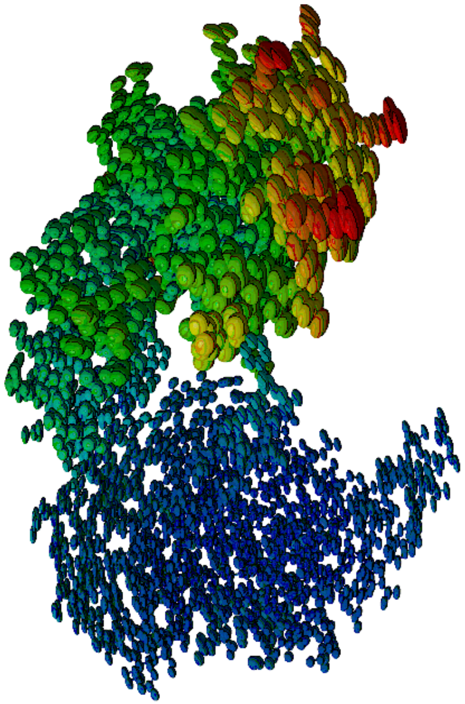
Refine any part of a model with any strategy: ***all in one run***



Rigid body +
Overall ADP

Individual
coordinates +
TLS

Individual coordinates +
TLS + restrained isotropic
ADP



+ Automatic water picking

+ Simulated Annealing

+ Add and use hydrogens

Running phenix.refine

Designed to be very easy to use:

Refinement of individual coordinates, B-factors, and occupancies for some atoms:

```
% phenix.refine model.pdb data.hkl
```

Add water picking and Simulated Annealing to default run above:

```
% phenix.refine model.pdb data.hkl simulated_annealing=true \
ordered_solvent=true
```

Refinement of individual coordinates and B-factors using neutron data:

```
% phenix.refine model.pdb data.hkl scattering_dictionary=neutron
```

To see all parameters (more than 200):

```
% phenix.refine --show_defaults=all
```

Running phenix.refine

```
% phenix.refine model.pdb data.hkl parameters_file
```

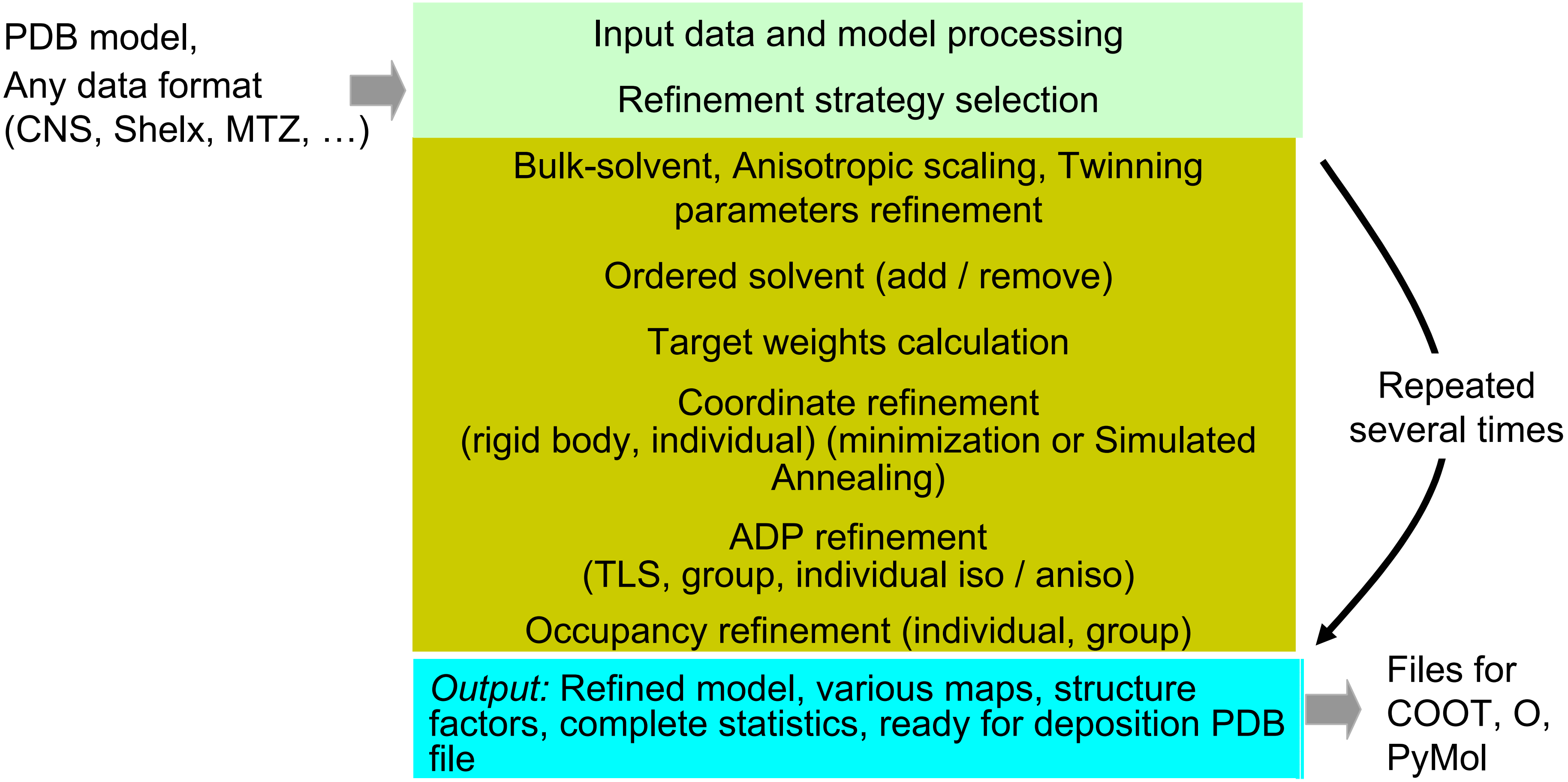
where parameter_file contains following lines:

```
refinement.main {
   high_resolution = 2.0
   low_resolution = 15.0
   simulated_annealing = True
   ordered_solvent = True
   number_of_macro_cycles = 5
}
refinement.refine.adp {
    tls = chain A
    tls = chain B
}
```

Equivalent command line run:

```
% phenix.refine model.pdb data.hkl xray_data.high_resolution=2
xray_data.low_resolution=15 simmulated_annealing=true
ordered_solvent=True adp.tls=“chain A” adp.tls=“chain B”
main.number_of_macro_cycles=5
```

Refinement flowchart



PDB model,
Any data format
(CNS, Shelx, MTZ, …)

Input data and model processing

Refinement strategy selection

Bulk-solvent, Anisotropic scaling, Twinning parameters refinement

Ordered solvent (add / remove)

Target weights calculation

Coordinate refinement
(rigid body, individual) (minimization or Simulated Annealing)

ADP refinement
(TLS, group, individual iso / aniso)

Occupancy refinement (individual, group)

Repeated several times

Output: Refined model, various maps, structure factors, complete statistics, ready for deposition PDB file

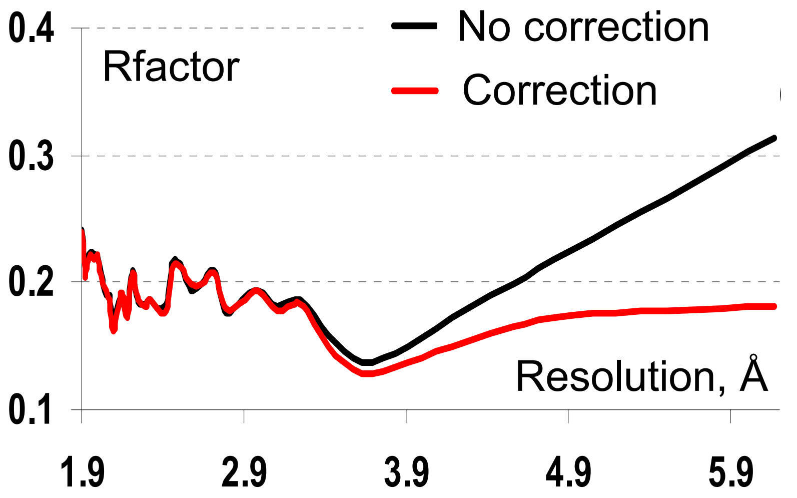
Files for COOT, O, PyMol

Bulk Solvent : facts

- Macromolecular crystals contain ~20 - 80% of solvent, most of it is disordered and is called bulk solvent.

- Bulk solvent significantly contributes to low resolution reflections (~4-6Å and lower).

Effect on total R-factor: from invisible to several percents (function of data resolution).



- *Flat Bulk Solvent Model* is currently the best. It assumes the constant electron density distribution outside of macromolecular region with k_{SOL}~0.35e/Å^3 and smearing factor B_{SOL}~50Å^2.

- Total model structure factor used in refinement and map calculation:

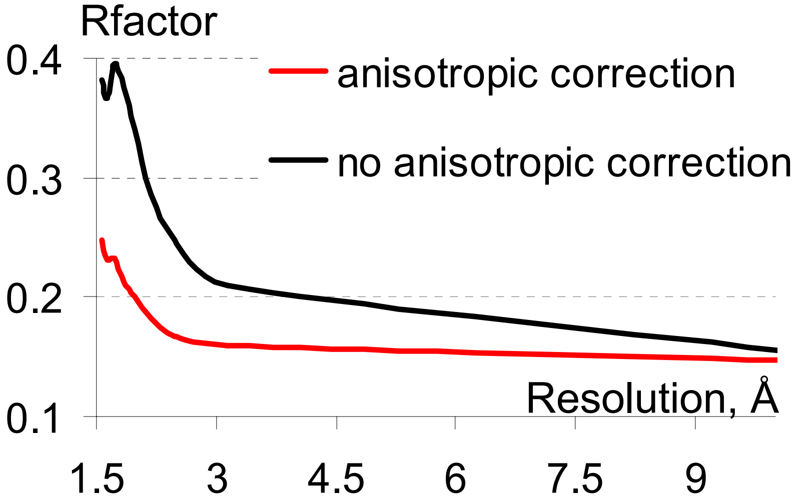
$$\mathbf{F}_{\mathrm{MODEL}} = k_{\mathrm{OVERALL}} e^{-\mathbf{s}\mathbf{U}_{\mathrm{CRYSTAL}}\mathbf{s}^{t}} \left(\mathbf{F}_{\mathrm{CALC\_ATOMS}} + k_{\mathrm{SOL}} e^{-\frac{B_{\mathrm{SOL}}\, s^{2}}{4}} \mathbf{F}_{\mathrm{MASK}} \right)$$

Effect of anisotropic scaling ($\mathbf{U}_{\mathbf{CRYSTAL}}$)

- Total model structure factor used in refinement and map calculation:

$$\mathbf{F}_{\mathrm{MODEL}} = k_{\mathrm{OVERALL}} e^{-\mathbf{s}\mathbf{U}_{\mathrm{CRYSTAL}}\mathbf{s}^{t}} \left(\mathbf{F}_{\mathrm{CALC\_ATOMS}} + k_{\mathrm{SOL}} e^{-\frac{B_{\mathrm{SOL}}\, s^{2}}{4}} \mathbf{F}_{\mathrm{MASK}} \right)$$

- 2MHR model from PDB

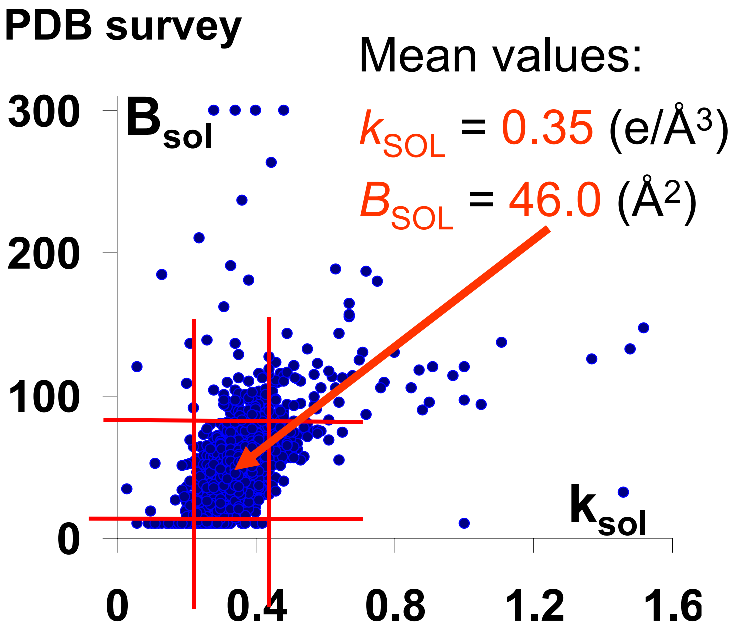


Significant impact on total R-factors:

no correction: Rwork ~ 25%

correction: Rwork ~ 17% , $\mathbf{U}_{\mathbf{CRYSTAL}}$ = (6.5 -9.1 3.8 0 0 0)

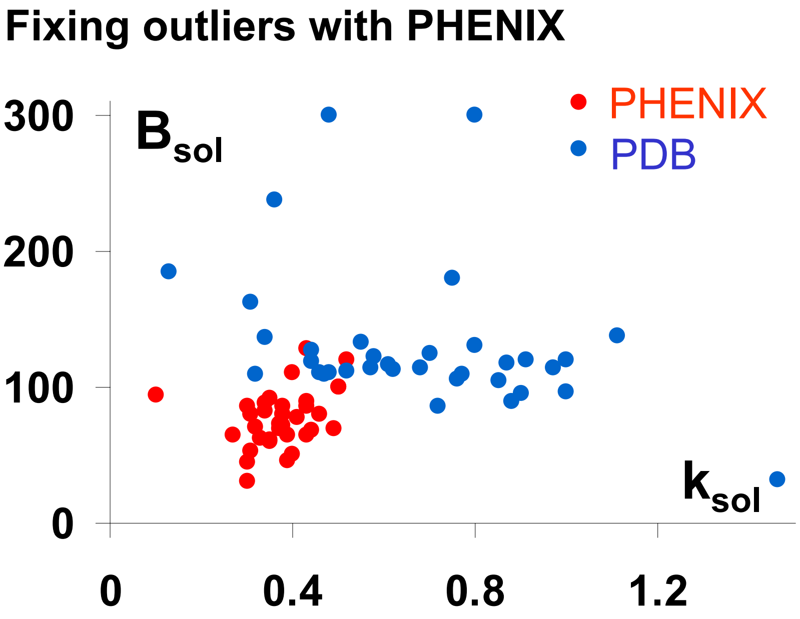
Bulk-solvent: robust implementation combined with anisotropic scaling

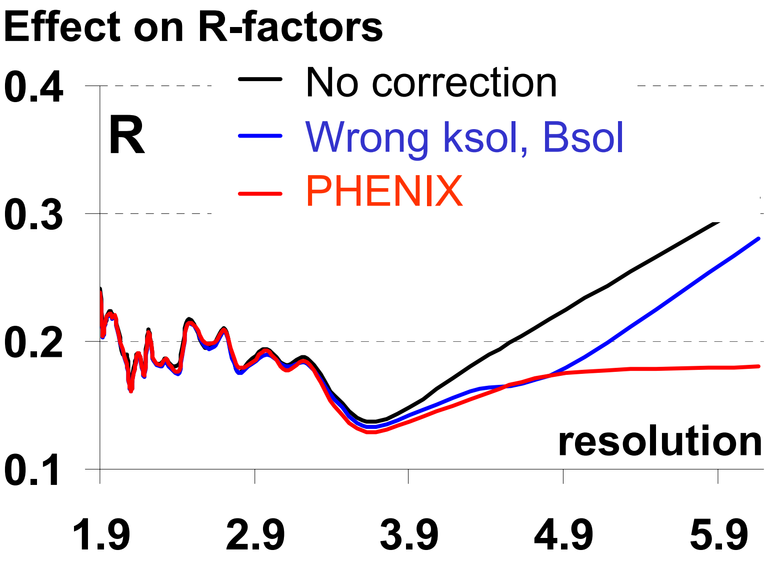

PDB survey

Mean values:

k_{SOL} = 0.35 (e/Å^3)

B_{SOL} = 46.0 (Å^2)


Fixing outliers with PHENIX

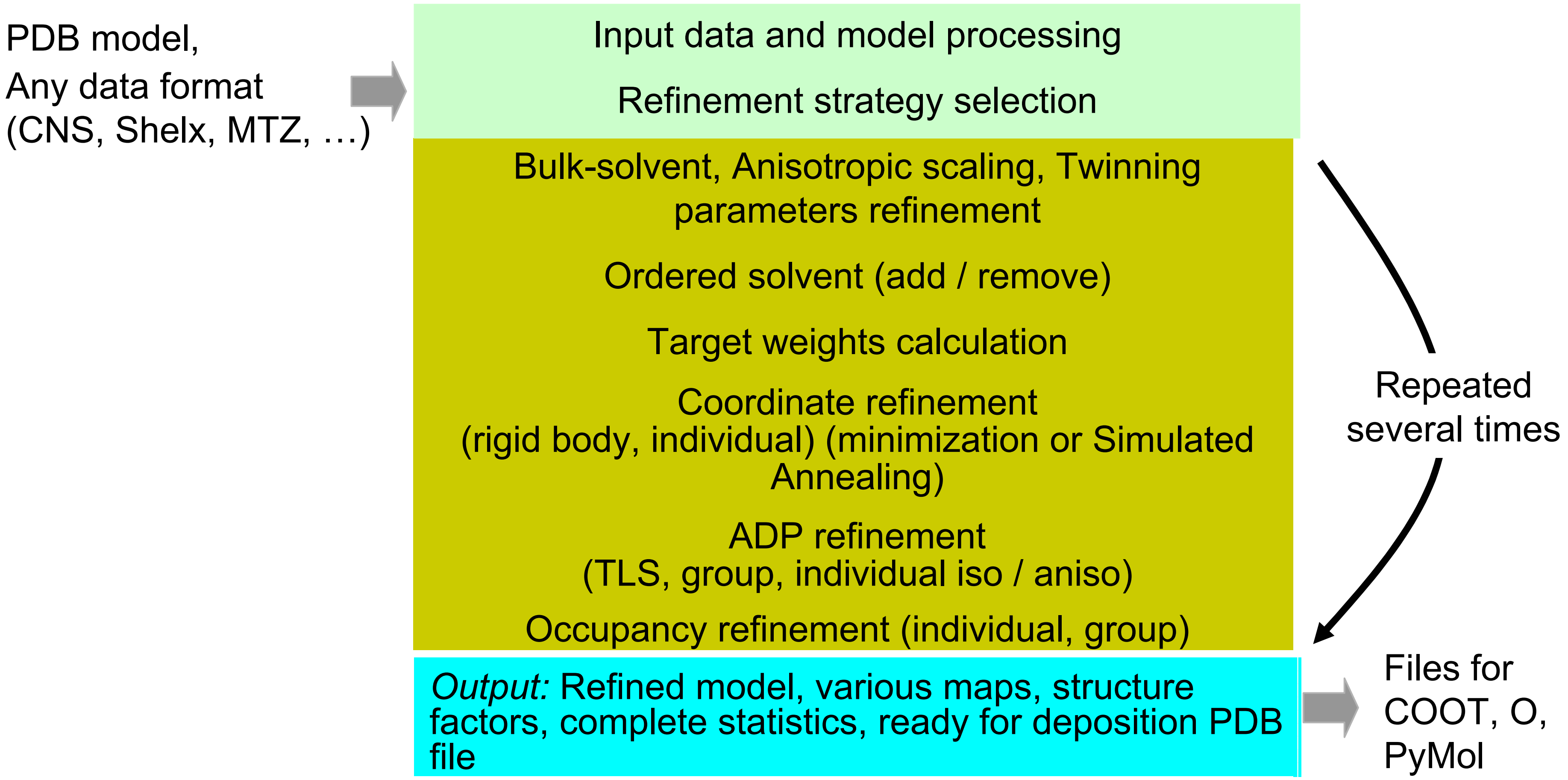

Effect on R-factors

Acta Cryst. (2005). D61, 850-855

A robust bulk-solvent correction and anisotropic scaling procedure

P.V. Afonine, R.W. Grosse-Kunstleve & P.D. Adams

Refinement flowchart



PDB model,
Any data format
(CNS, Shelx, MTZ, …)

Input data and model processing

Refinement strategy selection

Bulk-solvent, Anisotropic scaling, Twinning parameters refinement

Ordered solvent (add / remove)

Target weights calculation

Coordinate refinement
(rigid body, individual) (minimization or Simulated Annealing)

ADP refinement
(TLS, group, individual iso / aniso)

Occupancy refinement (individual, group)

Repeated several times

Output: Refined model, various maps, structure factors, complete statistics, ready for deposition PDB file

Files for
COOT, O,
PyMol

Automatic Water Picking

- **Built into refinement:**

  Loop over refinement macro-cycles:

  - bulk-solvent and anisotropic scale
  - water picking
  - refinement (XYZ, ADP, occupancies,…)

- **Water picking steps:**

  - remove "dead" water:

    2mFo-DFc, distances: water-other, water-water, Bmax/Bmin, anisotropy, occupancy max/min

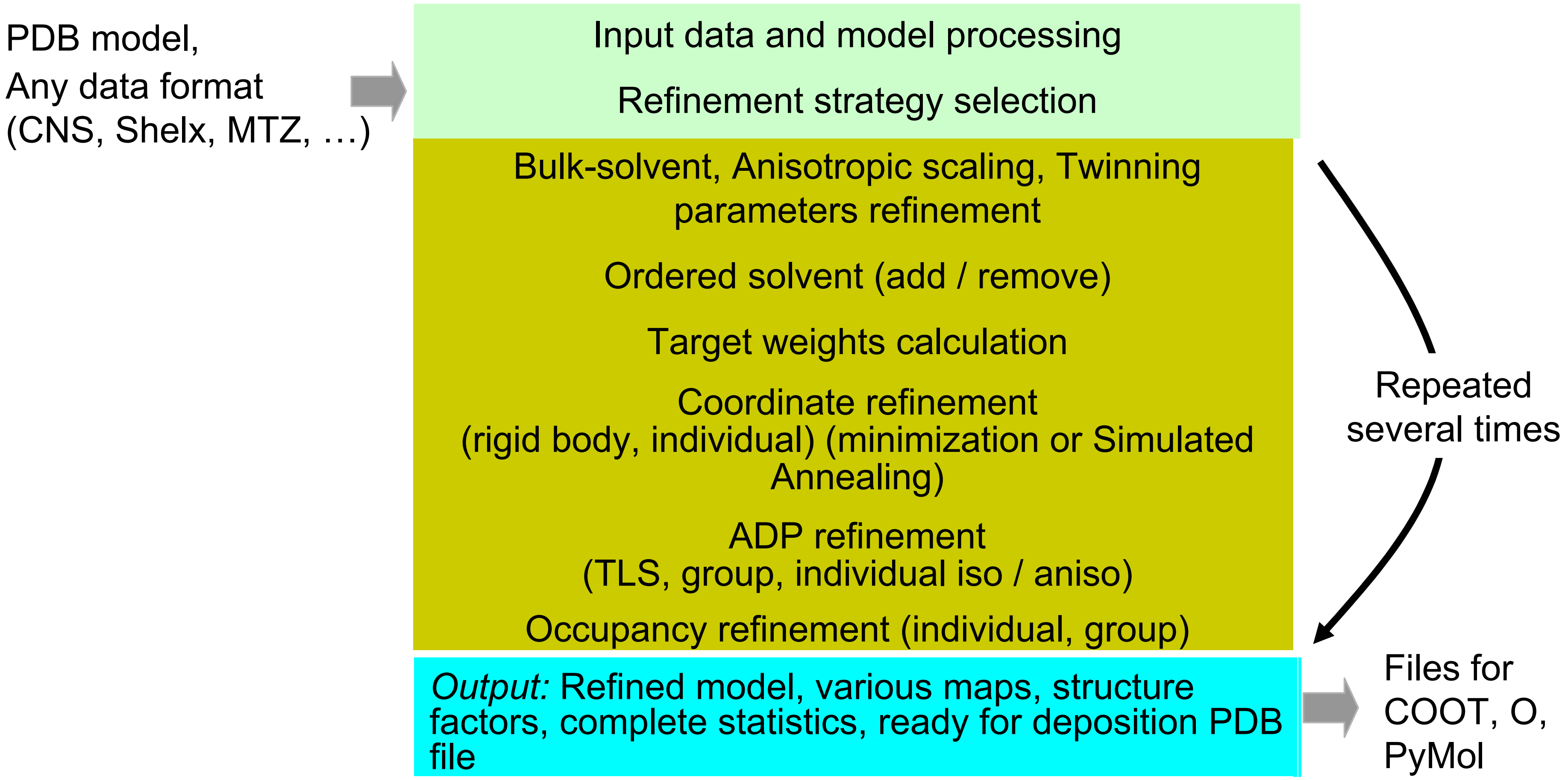
  - add new: mFo-DFc, distances: water-other, water-water
  - refine ADP (always) and occupancy (optional) for water only
  - remove "dead" water:

    2mFo-DFc, distances: water-other, water-water, Bmax/Bmin, anisotropy, occupancy max/min

- **Very flexible:** there are ~39 parameters available to adjust (if really wanted)

- **Limitation:** no peak sphericity or connectivity analysis (ligand density can be filled)

Refinement flowchart



PDB model,
Any data format
(CNS, Shelx, MTZ, …)

Input data and model processing

Refinement strategy selection

Bulk-solvent, Anisotropic scaling, Twinning parameters refinement

Ordered solvent (add / remove)

Target weights calculation

Coordinate refinement
(rigid body, individual) (minimization or Simulated Annealing)

ADP refinement
(TLS, group, individual iso / aniso)

Occupancy refinement (individual, group)

Repeated several times

Output: Refined model, various maps, structure factors, complete statistics, ready for deposition PDB file

Files for COOT, O, PyMol

Atomic Displacement Parameters (ADP or “B-factors”)

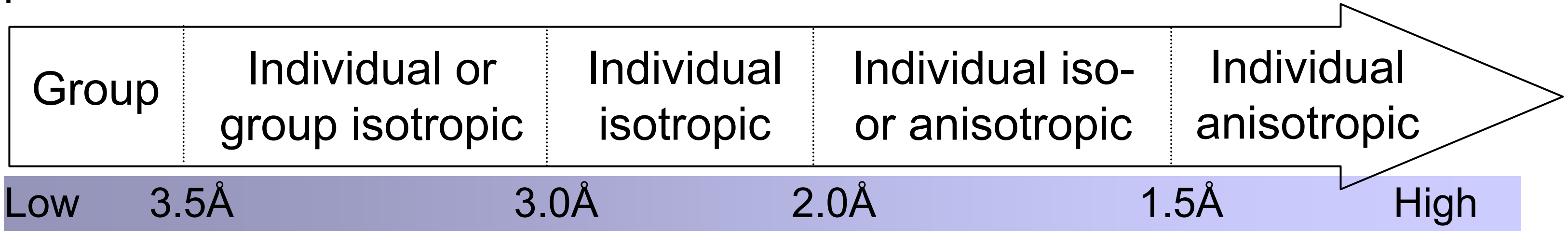
- **Total atomic ADP** $\mathbf{U_{TOTAL} = U_{CRYSTAL} + U_{TLS} + U_{INTERNAL} + U_{ATOM}}$

  - $\mathbf{U_{CRYSTAL}}$ - overall anisotropic scale w.r.t. cell axes (6 parameters).

  - $\mathbf{U_{TLS}}$ - rigid body displacements of molecules, domains, secondary structure elements. $U_{TLS} = T + ALA^t + AS + S^tA^t$ (20 TLS parameters per group).

  - $\mathbf{U_{INTERNAL}}$ - arising from normal modes of vibration (not modeled in current refinement software packages).

  - $\mathbf{U_{ATOM}}$ - vibration of individual atoms. Should obey Hirshfeld’s rigid bond postulate.

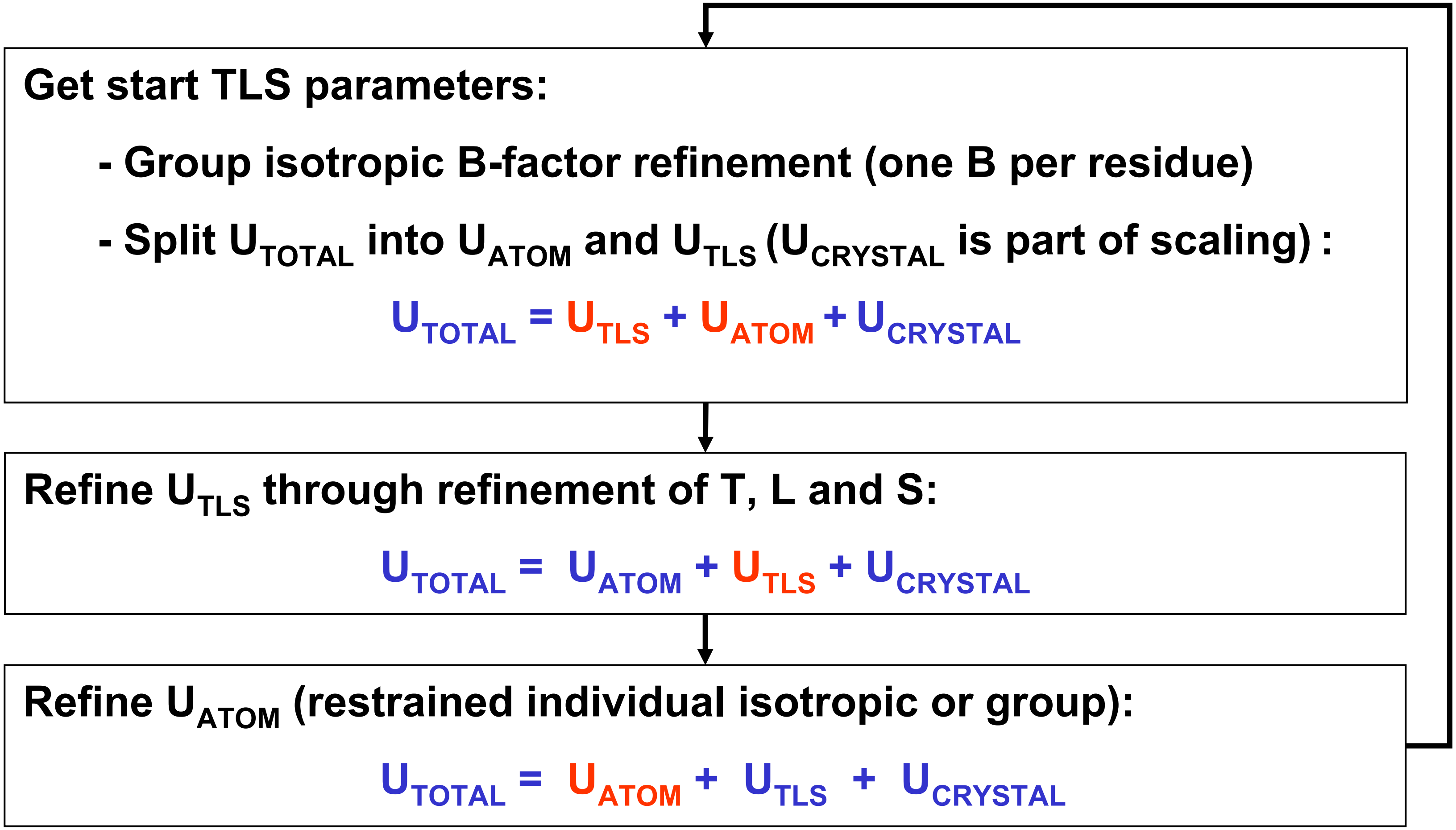


Group | Individual or group isotropic | Individual isotropic | Individual iso- or anisotropic | Individual anisotropic

Low 3.5Å 3.0Å 2.0Å 1.5Å High

TLS refinement in PHENIX: robust and efficient

$U_{TOTAL} = U_{CRYSTAL} + U_{TLS} + U_{ATOM}$



Get start TLS parameters:

- Group isotropic B-factor refinement (one B per residue)
- Split U_{TOTAL} into U_{ATOM} and U_{TLS} ($U_{CRYSTAL}$ is part of scaling) :

$$U_{TOTAL} = U_{TLS} + U_{ATOM} + U_{CRYSTAL}$$

Refine U_{TLS} through refinement of T, L and S:

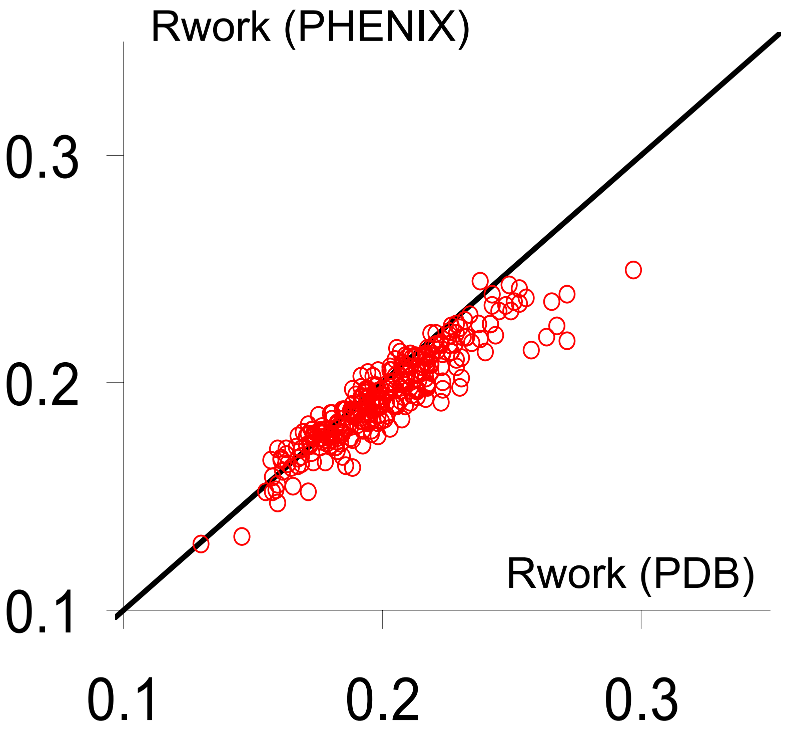
$$U_{TOTAL} = U_{ATOM} + U_{TLS} + U_{CRYSTAL}$$

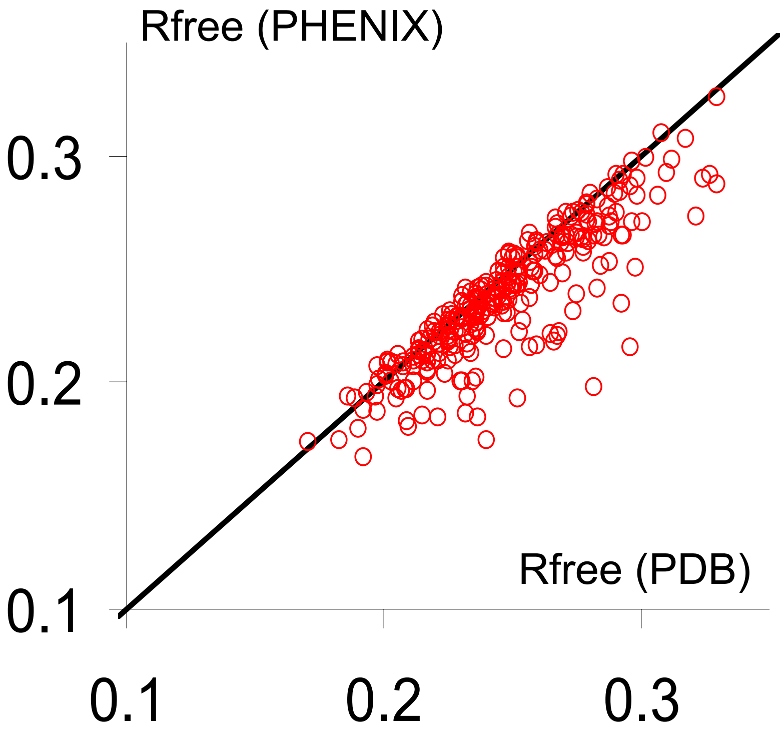
Refine U_{ATOM} (restrained individual isotropic or group):

$$U_{TOTAL} = U_{ATOM} + U_{TLS} + U_{CRYSTAL}$$

TLS refinement in PHENIX: robust and efficient

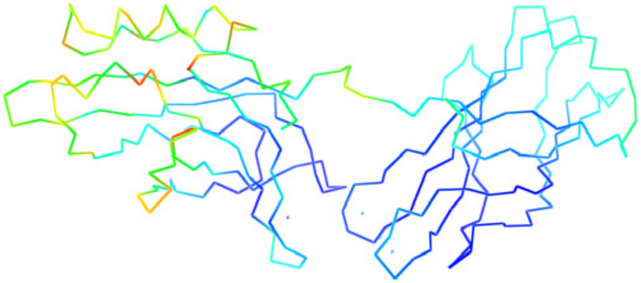
- Highly optimized algorithm based on systematic re-refinement of ~350 PDB models
- In most of cases phenix.refine produces better R-factors compared to published
- Never crashed or got “unstable”



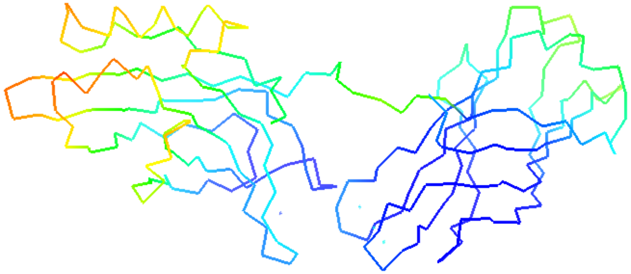


ADP refinement : from group B and TLS to individual anisotropic

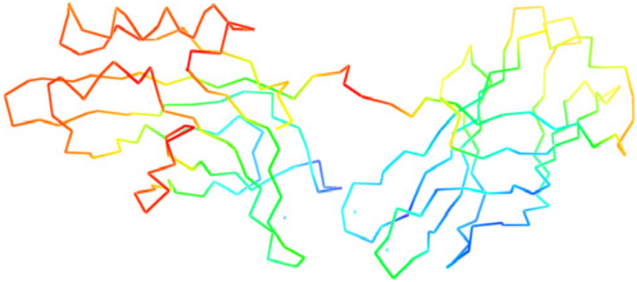
Synaptotagmin refinement at 3.2 Å



CNS
R-free = *34.%*
R = *29.%*



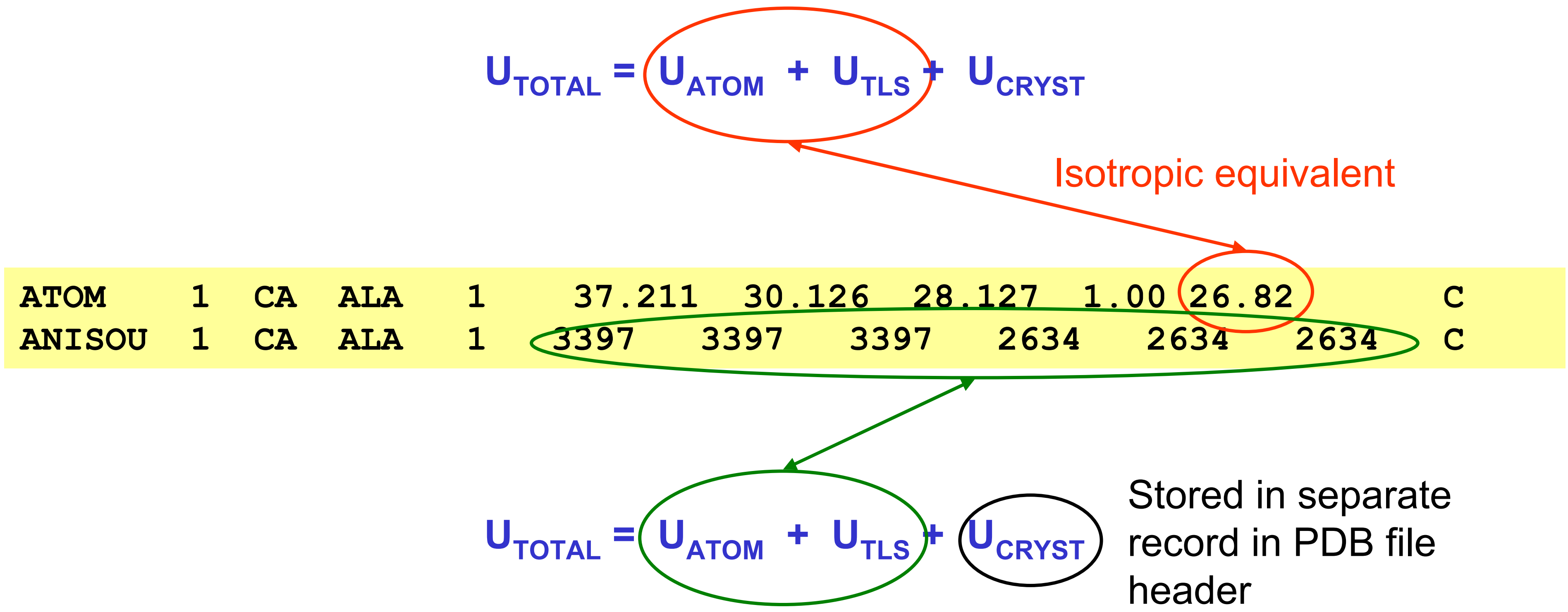
PHENIX – Isotropic restrained ADP
R-free = *27.7%*
R = *24.6%*



PHENIX – TLS + Isotropic ADP
R-free = *24.4%*
R = *20.7%*

ADP refinement: what goes to PDB

phenix.refine outputs TOTAL B-factor (iso- and anisotropic):



$$U_{TOTAL} = U_{ATOM} + U_{TLS} + U_{CRYST}$$

Isotropic equivalent

```
ATOM     1  CA  ALA    1     37.211  30.126  28.127  1.00 26.82          C
ANISOU   1  CA  ALA    1   3397   3397   3397   2634   2634   2634       C
```

$$U_{TOTAL} = U_{ATOM} + U_{TLS} + U_{CRYST}$$

Stored in separate record in PDB file header

Atom records are self-consistent:

✓ Straightforward visualization (color by B-factors, or anisotropic ellipsoids)

✓ Straightforward computation of other statistics (R-factors, etc.) – no need to use external helper programs for any conversions.

Occupancy refinement

- Automatic constraints for occupancies:

```
ATOM    549  HA3 GLY A  34     -23.064   7.146 -23.942  1.00 15.44           H
ATOM    550  H   AGLY A  34     -24.447   7.644 -21.715  0.15  8.34           H
ATOM    551  D   BGLY A  34     -24.413   7.658 -21.713  0.85  7.65           D
ATOM    552  N    GLU A  35     -22.459   9.801 -22.791  1.00  8.54           N


ATOM      1  N   AGLY A 192      -5.782  17.932  11.414  0.72  8.38           N
ATOM      2  CA  AGLY A 192      -6.979  17.425  10.929  0.72 10.12           C
ATOM      3  C   AGLY A 192      -6.762  16.088  10.271  0.72  7.90           C
ATOM      4  O   AGLY A 192      -5.920  15.288  10.688  0.72  7.86           O
ATOM      7  N   BGLY A 192     -11.719  17.007   9.061  0.28  9.89           N
ATOM      8  CA  BGLY A 192     -10.495  17.679   9.569  0.28 11.66           C
ATOM      9  C   BGLY A 192      -9.259  17.590   8.718  0.28 12.76           C
ATOM     10  O   BGLY A 192      -9.508  17.810   7.396  0.28 14.04           O
```

- Any user defined selections for individual and/or group occupancy refinement can be added on top of automatic selection.

Restraints and novel ligands in phenix.refine

- When running:

```
% phenix.refine model.pdb data.hkl
```

each item in model.pdb is matched against the CCP4 Monomer Library to extract the topology and parameters and to automatically build corresponding restraints.

- If model.pdb contains an item not available in CCP4 Monomer Library, e.g. a novel ligand, use eLBOW to generate topology and parameter definitions for refinement:

```
% phenix.elbow model.pdb --residue=LIG
Or
% phenix.elbow model.pdb --do-all
```

This will produce the file LIG.cif which can be used for refinement:

```
% phenix.refine model.pdb data.hkl LIG.cif
```

Neutron and joint X-ray/Neutron refinement

Macromolecular Neutron Crystallography Consortium (MNC)

Los Alamos National Laboratory
Paul Langan, Marat Mustyakimov, Benno Schoenborn

Lawrence Berkeley National Lab (LBNL)
Paul Adams, Pavel Afonine

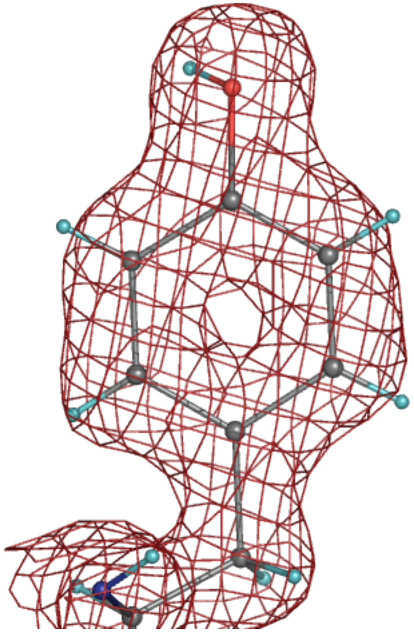
http://mnc.lanl.gov/

Maps: X-ray and neutron

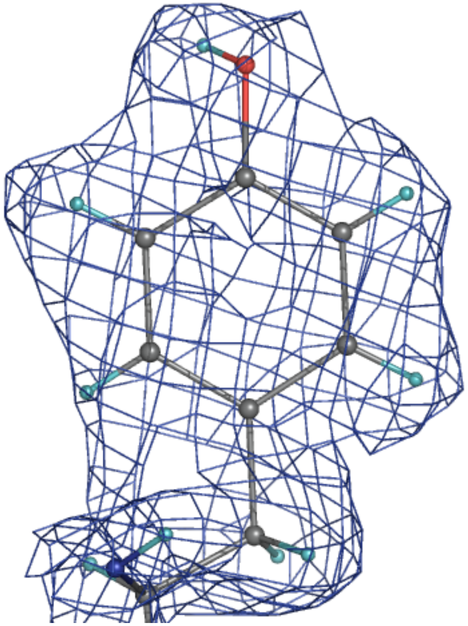
- Different techniques – different information

2mFo-DFc maps (Aldose Reductase)

X-ray (1.8 Å)



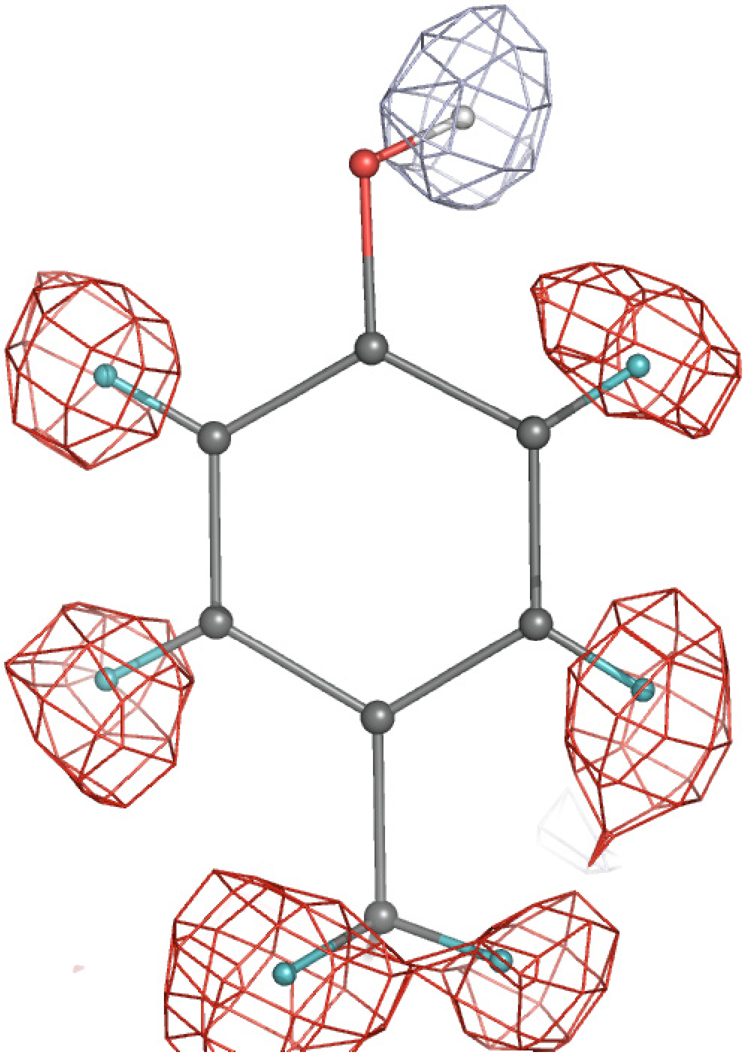
Neutron (2.2 Å)



Quantum model of catalysis based on a mobile proton revealed by subatomic x-ray and neutron diffraction studies of h-aldose reductase
PNAS, 2008; 105(6): 1844 - 1848.

Maps: X-ray and neutron

- Different techniques – different information (Automatic determination of H/D state)



PDB: 1iu6 and 1iu5 (resolution ~1.6A)

joint XN refinement

Fo-Fc map, (H and D omitted), neutron data

positive (blue, 2.6σ, D atoms)

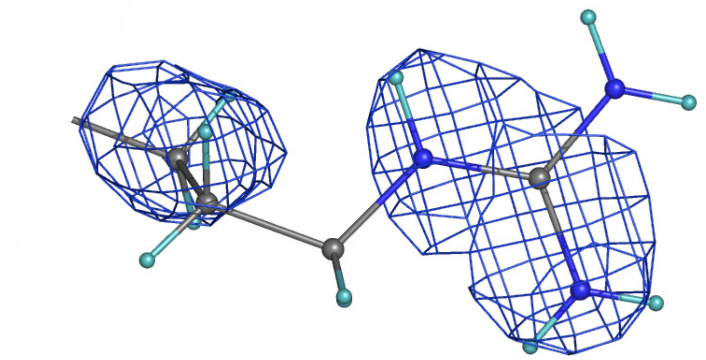
negative (red, -2.9σ, H atoms)

Individual neutron and joint X+N refinement

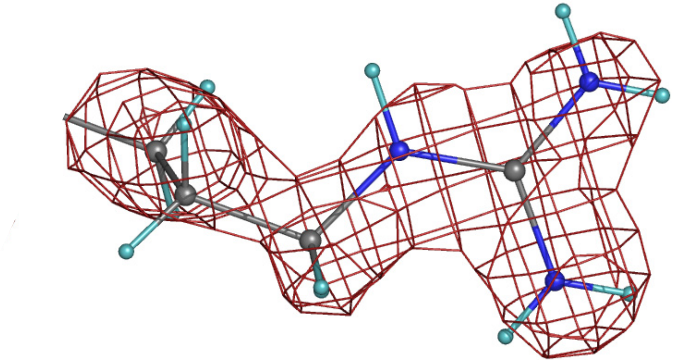
- Maps are improved after joint refinement compared to refinement with neutron data only:

2mFo-DFc, neutron data, 2σ, 2.2 Å resolution (Aldose Reductase)

Refinement (neutron data only)



Refinement (X-ray and neutron data)



- Target used for joint X-ray + neutron refinement:

$$\text{Target}_{\text{JOINT}} = E_{\text{XRAY}} * w_{\text{XC}} + E_{\text{NEUTRON}} * w_{\text{NC}} * w_{\text{XN}} + E_{\text{GEOM}}$$

- Running joint X-ray + neutron refinement in PHENIX

```
% phenix.refine model.pdb data_xray.hkl neutron_data.file_name=data_neutron.hkl
input.xray_data.labels=FOBSx input.neutron_data.labels=FOBSn
```

Hydrogen atoms in refinement

- phenix.refine offers various options for handling H atoms:
  - Riding model (low-high resolution)
  - Individual atoms (ultrahigh resolution or neutron data)
  - Account for scattering contribution or just use to improve the geometry
- Expected benefits from using the H atoms in refinement:
  - Improve R-factors
  - Improve model geometry (remove bad clashes)
  - Model residual density at high resolution or in neutron maps
- Example from automatic re-refinement of 1000 PDB models with and without H:

pdb	**resolution**	**Rfree(no H) – Rfree(with H)**
1akg	1.1	1.9
1byp	1.75	1.41
1dkp	2.3	0.93
1rgv	2.9	0.50

- Build hydrogens:

```
%phenix.reduce model.pdb > model_H.pdb
```

or

```
%phenix.build_hydrogens model.pdb
```

Refinement with twinned data

- Two steps to perform twin refinement:

  - run phenix.xtriage to get twin operator (twin law):

```
% phenix.xtriage data.mtz
```

  - run phenix.refine:

```
% phenix.refine model.pdb data.mtz twin_law="-h-k,k,-l"
```

- Taking twinning into account makes difference:

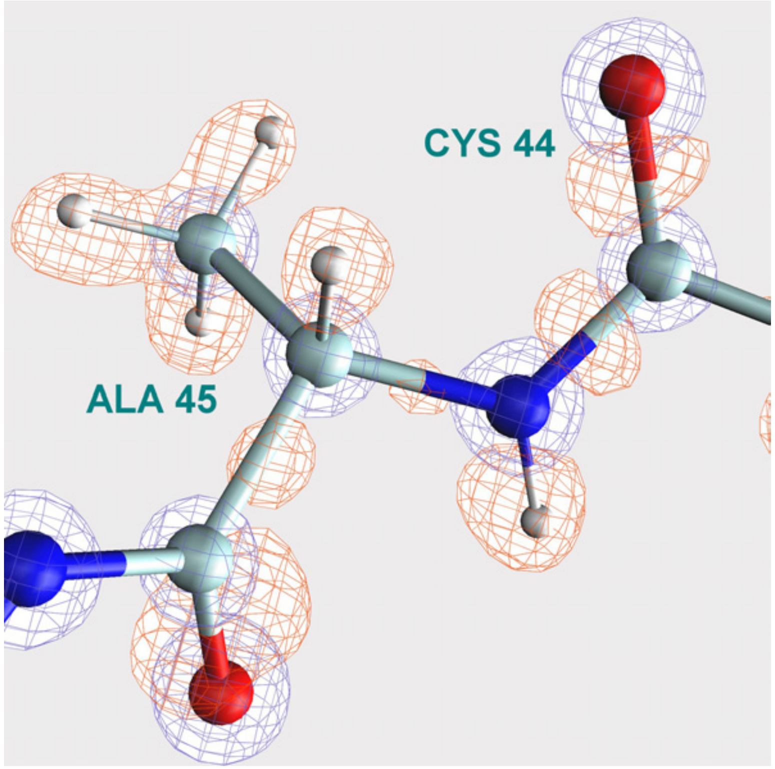
Interleukin mutant (PDB code: 1l2h)

	R/R-free (%)
PHENIX (no twinning):	24.9 / 27.4
PHENIX (twin refinement):	15.3 / 19.2

Refinement at subatomic resolution

- Subatomic resolution (higher than ~ 0.9 Å): bond densities and H atoms

Aldose Reductase (0.66 Å resolution)



Fo-Fc (orange) 2Fo-Fc (blue)

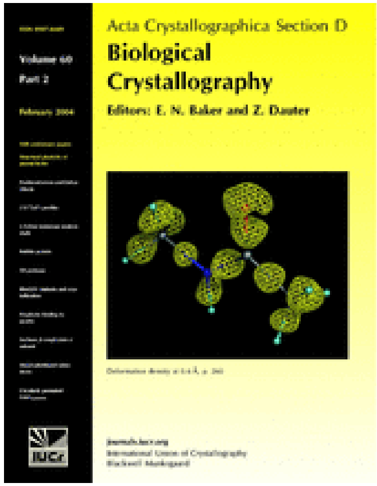
Modeling at subatomic resolution: IAS model

- Basics of IAS model:

  Afonine et al, Acta Cryst. D60 (2004)

- First practical examples of implementation and use in PHENIX:

  Afonine et al, Acta Cryst. D63, 1194-1197 (2007)

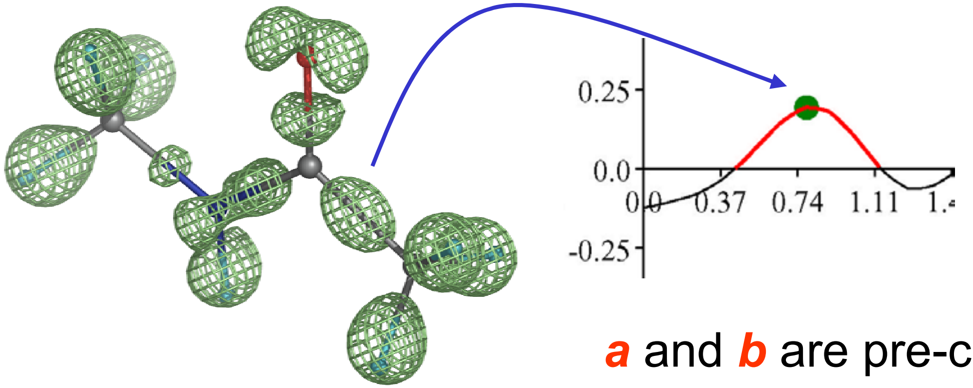


Acta Crystallographica Section D

Biological Crystallography

Editors: E. N. Baker and Z. Dauter

IAS modeling in PHENIX



Simple Gaussian is good enough:

$$f_{bond\_scatterer}(\mathbf{s}) = a \exp(b\,\mathbf{s}^2)$$

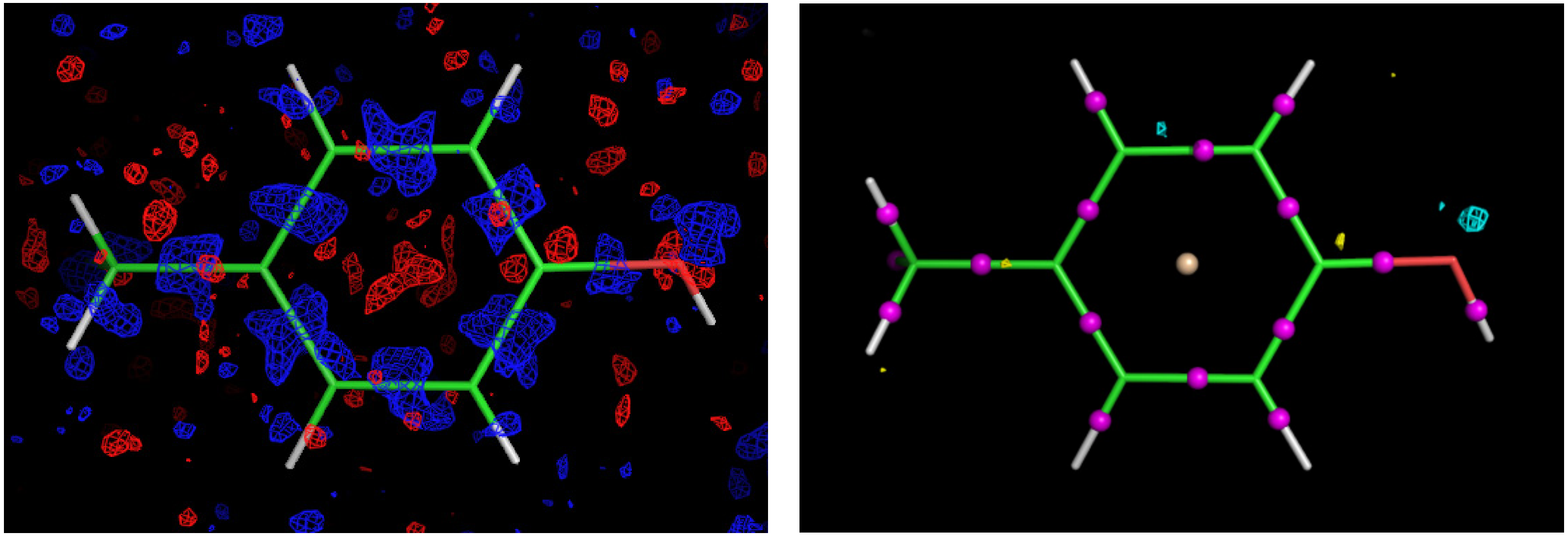
a and *b* are pre-computed library for most bond types

- Compared to Multipolar model that is commonly used at ultra-high resolutions, the new IAS model features:

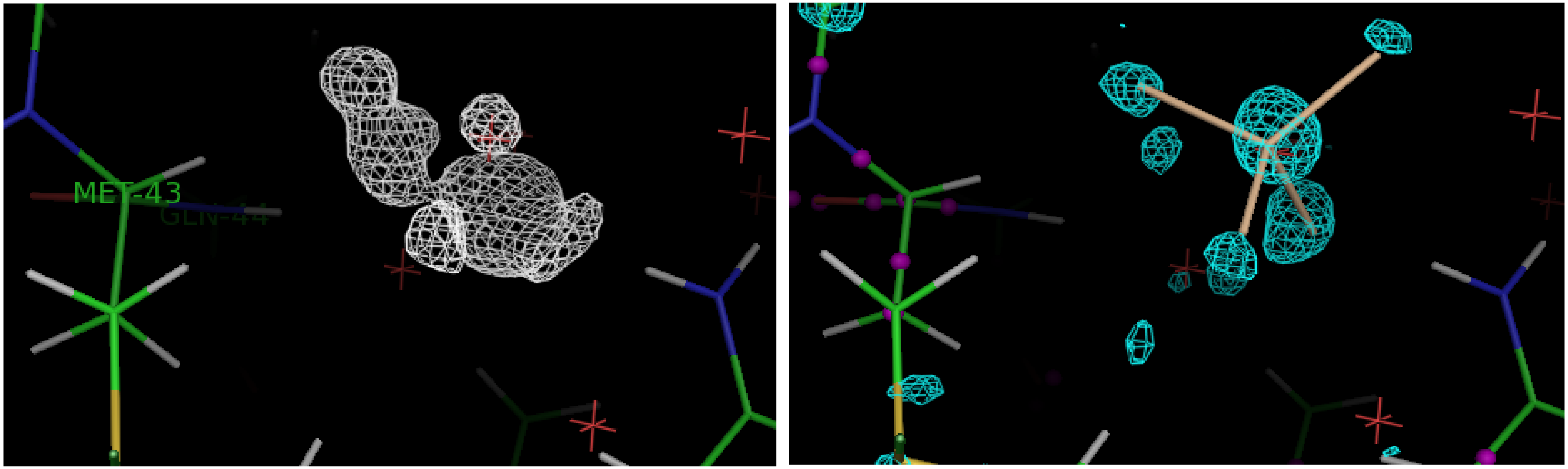
  - faster and much simpler computations,
  - less or no risk of overfitting,
  - similar results as Multipolar model (R-factors, ADP, maps)

IAS modeling: benefits

- Improve maps: reduce noise. Before (left) and after (right) adding of IAS.

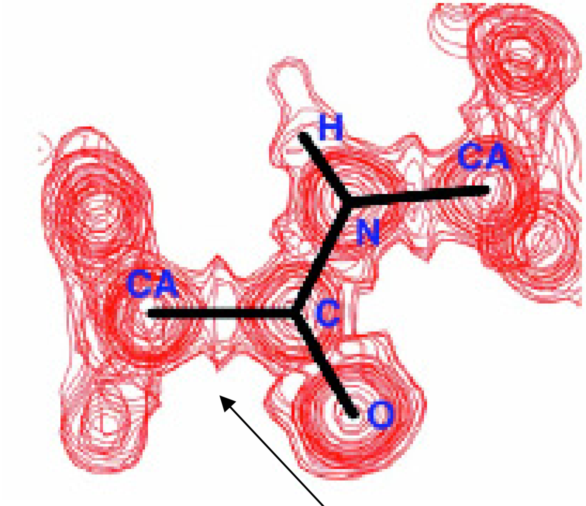


- Find new features: originally wrong water (left) replaced with SO4 ion (right) clearly suggested by improved map after adding IAS

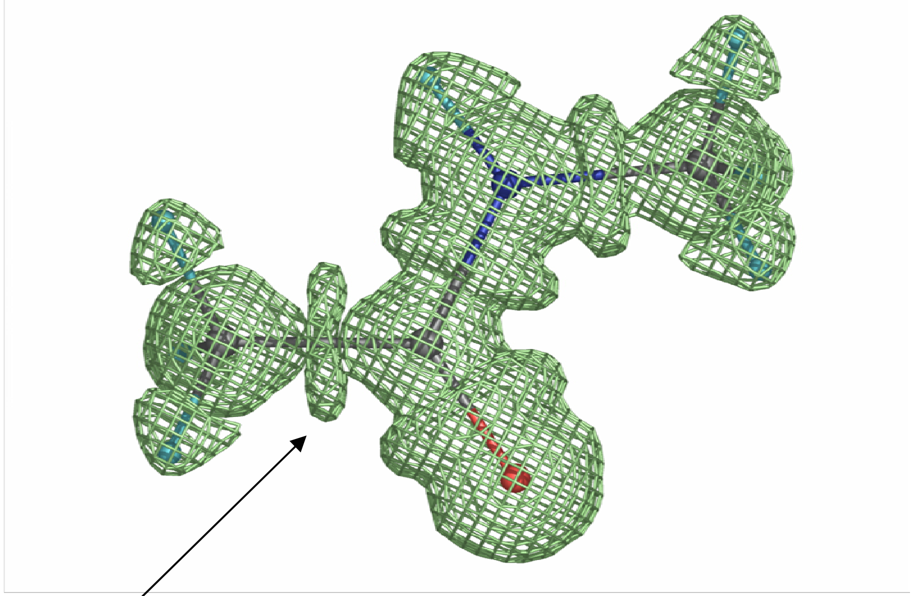


Maps at subatomic resolutions: dangers

- “Experimental Observation of Bonding Electrons in Proteins”, JBC, 1999, Vol. 274.



- $(F_{CALC}, \varphi_{CALC})$ synthesis at 0.6 Å :



This is not bonding electrons! This is Fourier series truncation ripples !

Shocking examples (or why automation is important…)

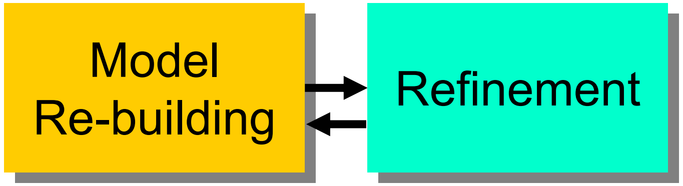
- Structure from PDB: **1eic** (resolution = 1.4Å)

PUBLISHED: Rwork = 20% Rfree = 25%

- Clear problems:
  - No H atoms;
  - All atoms isotropic;
- Potential problems
  - Inoptimal weights, refinement is not converged, incomplete solvent model
- Fixing the model with PHENIX:
  - Add and refine H as riding model
  - Update ordered solvent
  - Refine all atoms as anisotropic (except H and water)
  - Optimize Xray/Restraints weights

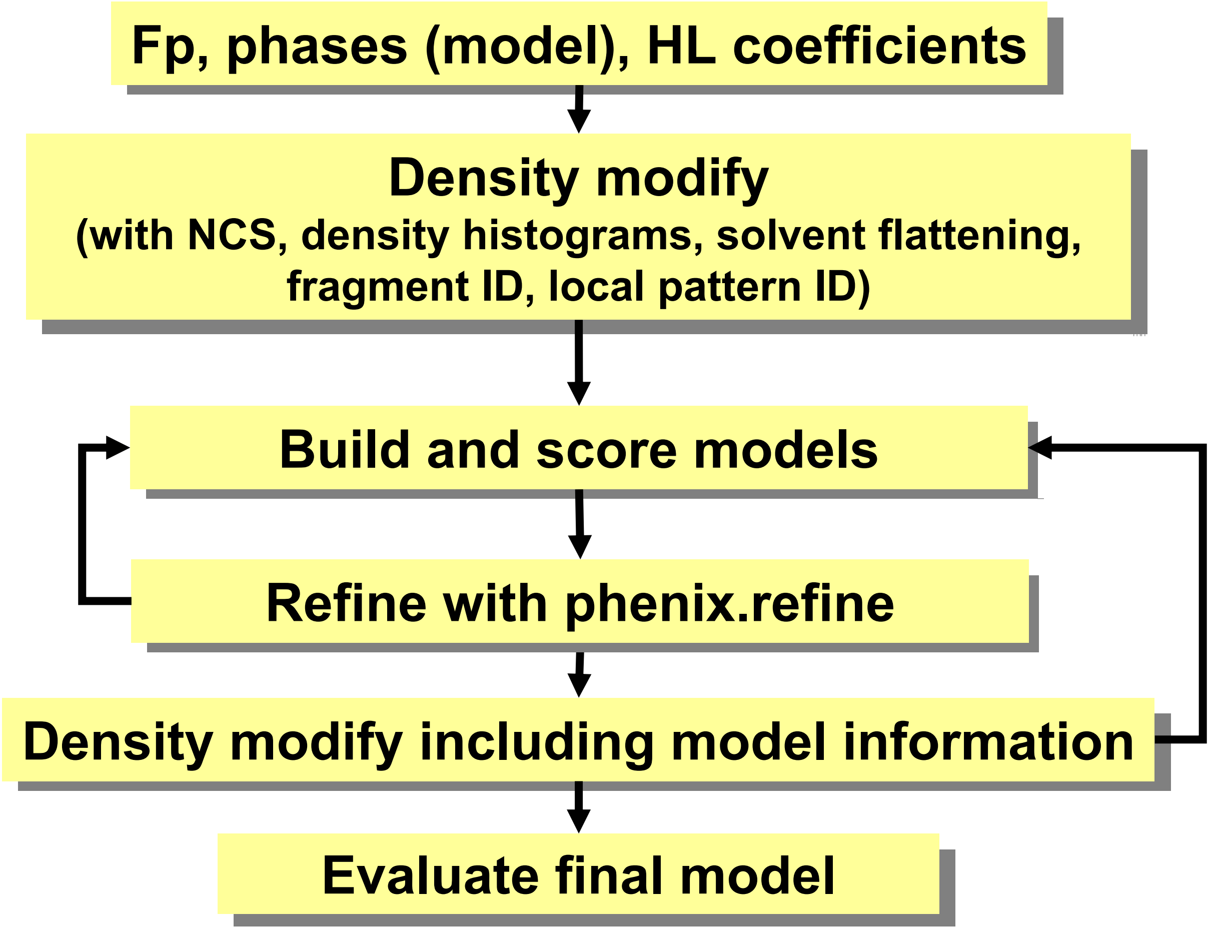
FINAL MODEL: Rwork = 14% Rfree = 17%

Autobuild wizard in PHENIX: phenix.refine + (SOLVE & RESOLVE)



Model Re-building

Refinement



Fp, phases (model), HL coefficients

Density modify
(with NCS, density histograms, solvent flattening, fragment ID, local pattern ID)

Build and score models

Refine with phenix.refine

Density modify including model information

Evaluate final model

phenix.pdbtools

- phenix.pdbtools – set of tools for PDB file manipulations

- For any selected model part:
  - shake coordinates, ADP, occupancies
  - rotation-translation shift of coordinates
  - shift, scale, set ADP (add, multiply, assign a constant)
  - converting to isotropic / anisotropic
  - removing selected part of a model

- Easy to run:

```
% phenix.pdbtools model.pdb rotate="10 20 30" selection="chain A"
```

- Also:
  - complete model statistics (geometry, B-factors)
  - geometry regularization
  - output MTZ with Fcalc (or Fmodel) computed as:

```
Fmodel = scale * exp(-h * bcart * ht) * (
       Fcalc_atoms + ksol * exp(-bsol * s^2) * Fmask)
```

phenix.superpose_pdbs

- Usage:

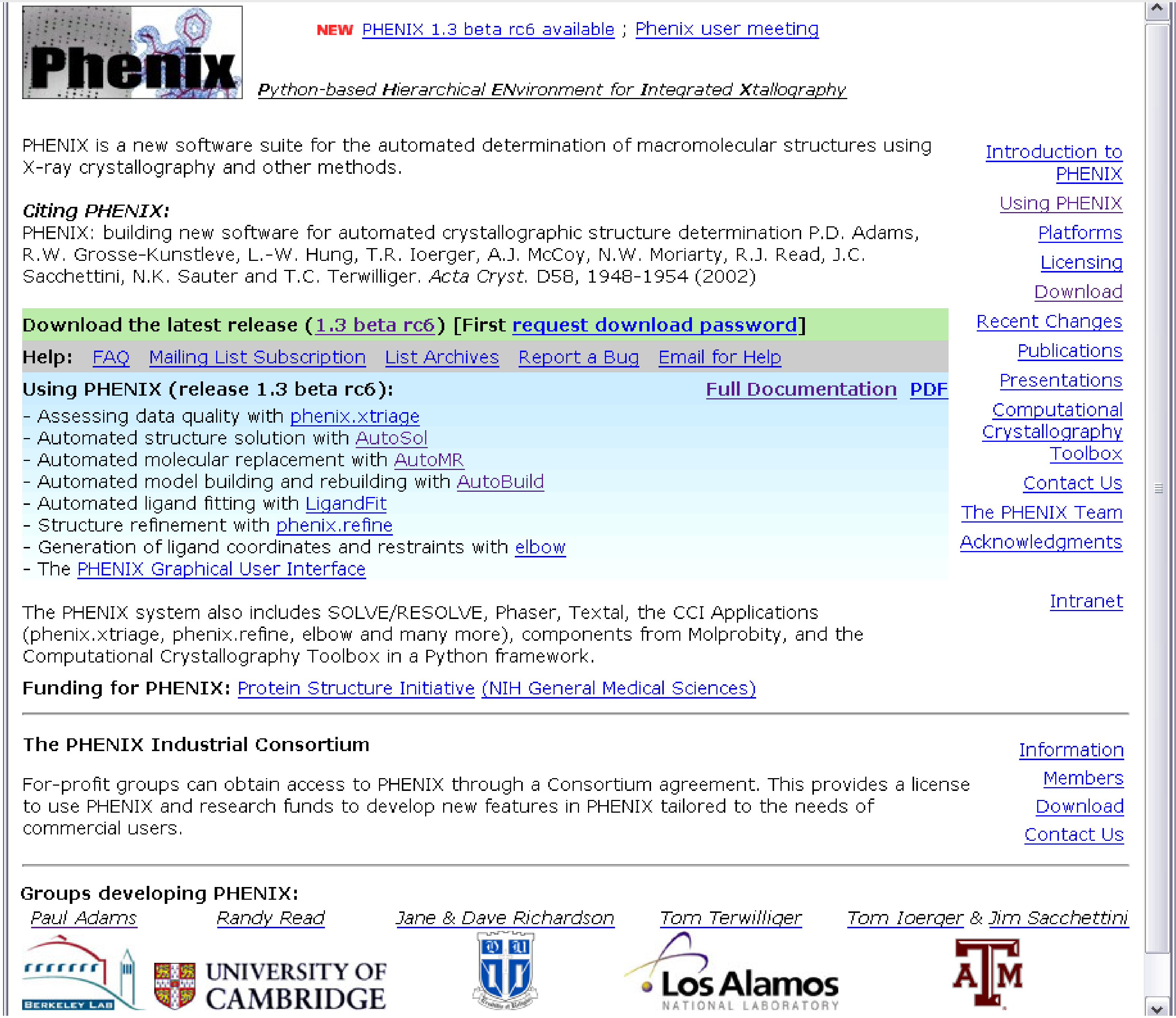
- uses alignment if atoms not 100% matching:

```
% phenix.superpose_pdbs fixed.pdb moving.pdb
```

- superpose using selected parts (must exactly match):

```
% phenix.superpose_pdbs fixed.pdb moving.pdb \
selection_fixed="chain A and name CA" \
selection_moving="chain B and name CA"
```

Documentation: www.phenix-online.org



NEW PHENIX 1.3 beta rc6 available ; Phenix user meeting

Python-based Hierarchical ENvironment for Integrated Xtallography

PHENIX is a new software suite for the automated determination of macromolecular structures using X-ray crystallography and other methods.

Citing PHENIX:
PHENIX: building new software for automated crystallographic structure determination P.D. Adams, R.W. Grosse-Kunstleve, L.-W. Hung, T.R. Ioerger, A.J. McCoy, N.W. Moriarty, R.J. Read, J.C. Sacchettini, N.K. Sauter and T.C. Terwilliger. *Acta Cryst.* D58, 1948-1954 (2002)

Download the latest release (1.3 beta rc6) [First request download password]

Help: FAQ Mailing List Subscription List Archives Report a Bug Email for Help

Using PHENIX (release 1.3 beta rc6): **Full Documentation** **PDF**

- Assessing data quality with phenix.xtriage
- Automated structure solution with AutoSol
- Automated molecular replacement with AutoMR
- Automated model building and rebuilding with AutoBuild
- Automated ligand fitting with LigandFit
- Structure refinement with phenix.refine
- Generation of ligand coordinates and restraints with elbow
- The PHENIX Graphical User Interface

The PHENIX system also includes SOLVE/RESOLVE, Phaser, Textal, the CCI Applications (phenix.xtriage, phenix.refine, elbow and many more), components from Molprobity, and the Computational Crystallography Toolbox in a Python framework.

Funding for PHENIX: Protein Structure Initiative (NIH General Medical Sciences)

Introduction to PHENIX
Using PHENIX
Platforms
Licensing
Download
Recent Changes
Publications
Presentations
Computational Crystallography Toolbox
Contact Us
The PHENIX Team
Acknowledgments

Intranet

The PHENIX Industrial Consortium

For-profit groups can obtain access to PHENIX through a Consortium agreement. This provides a license to use PHENIX and research funds to develop new features in PHENIX tailored to the needs of commercial users.

Information
Members
Download
Contact Us

Groups developing PHENIX:

Paul Adams *Randy Read* *Jane & Dave Richardson* *Tom Terwilliger* *Tom Ioerger & Jim Sacchettini*

Reporting bugs, problems, asking questions

- **Something didn’t work as expected?... program crashed?... missing feature?...**

  - **Bad:** silently give up and run away looking for alternative software.

  - **Good:** report us a problem, ask a question, request a feature (explain why it’s good to have), ask for help (send data).

- **Reporting a bug / problem:**

  - **Bad:** “Hi! phenix.refine crashed and I don’t know why and what to do.”

  - **Good:** “Hi! phenix.refine crashed. Here are:
    1) PHENIX version;
    2) The exact command I used;
    3) Input and output files (at least logs).”

PHENIX: www.phenix-online.org

- Computational Crystallography Initiative
    - Paul Adams
    - Nigel Moriarty
    - Nick Sauter
    - Peter Zwart
    - Ralf Grosse-Kunstleve
- Los Alamos National Laboratory
    - Tom Terwilliger
    - Li-Wei Hung
- Cambridge University
    - Randy Read
    - Airlie McCoy
    - Laurent Storoni
- Texas A&M University
    - Tom Ioerger
    - Jim Sacchettini
    - Erik McKee

- Others
    - Axel Brunger
    - David Abrahams
    - CCP4 developers
    - Alexei Vagin & Garib Murshudov
    - Kevin Cowtan
    - Sasha Urzhumtsev
    - Vladimir Lunin
- Duke University
    - Jane and David Richardson
    - Ian Davis
    - Vincent Chen
    - Bob Immormino

Funding:

NIH / NIGMS [*P01GM063210, R01GM071939*, *P01GM064692*]
LBNL [*DE-AC03-76SF00098*]
PHENIX industrial consortium