Australian Government

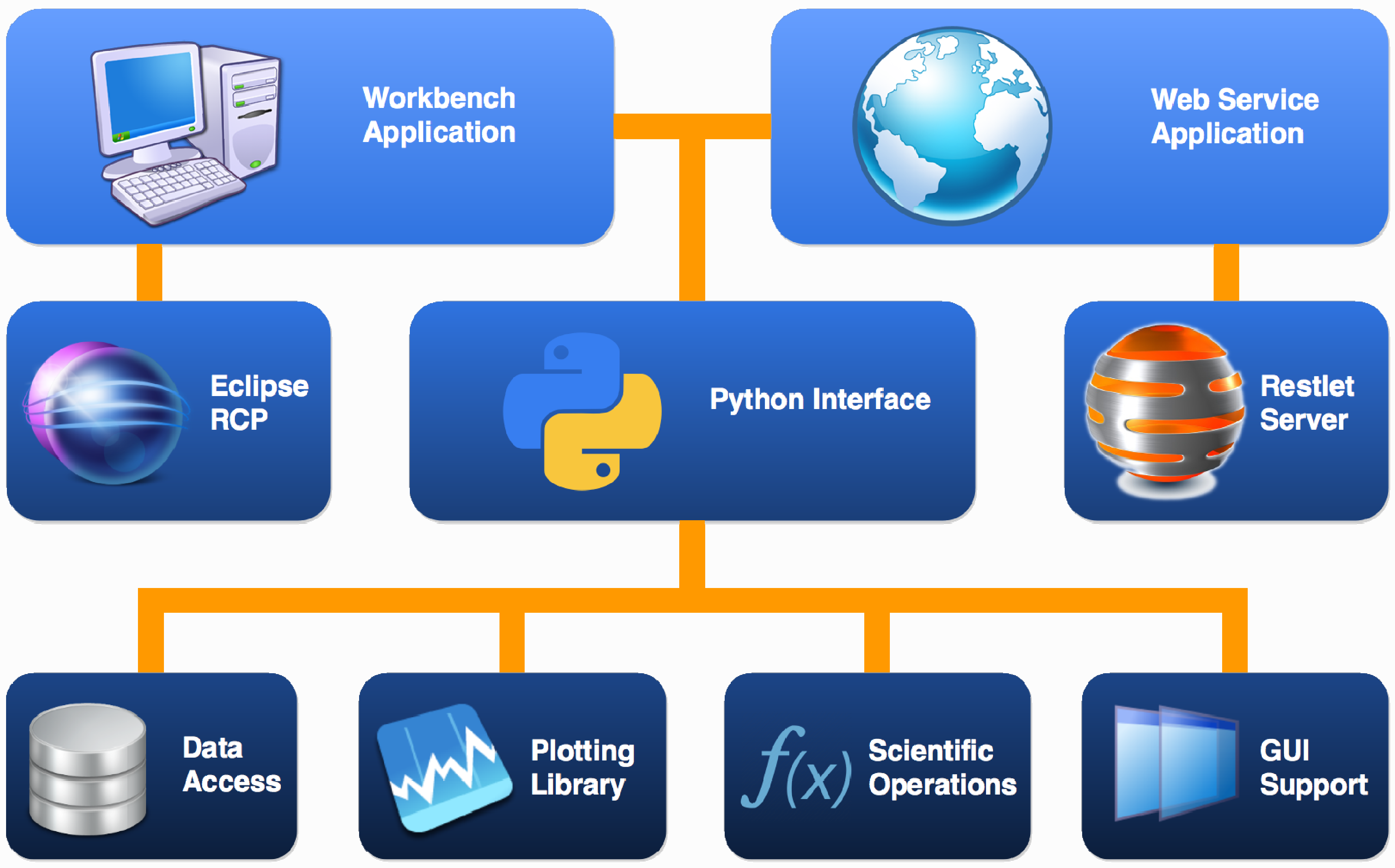
ansto

Python Scripting for Instrument Control and Online Data Treatment

N. Xiong, D. Mannicke, N. Hauser, The Bragg Institute, ANSTO
Norman.Xiong@ansto.gov.au; Locked Bag 2001, Kirrawee DC NSW 2232 AUSTRALIA

Introduction

The Gumtree scripting interface has a *numpy*-like Python library that makes data treatment easier. It also has a GUI library that automatically generates control panels for scripts. Same scripts can be loaded in both the desktop application and the web service application for online data treatment. The scripting interface benefits both the users and the developers. Users can easily make scripts to run experiments or treat the data, with a graphical interface automatically created for these scripts. Developers save a lot of time in compiling and deploying products when using this feature.



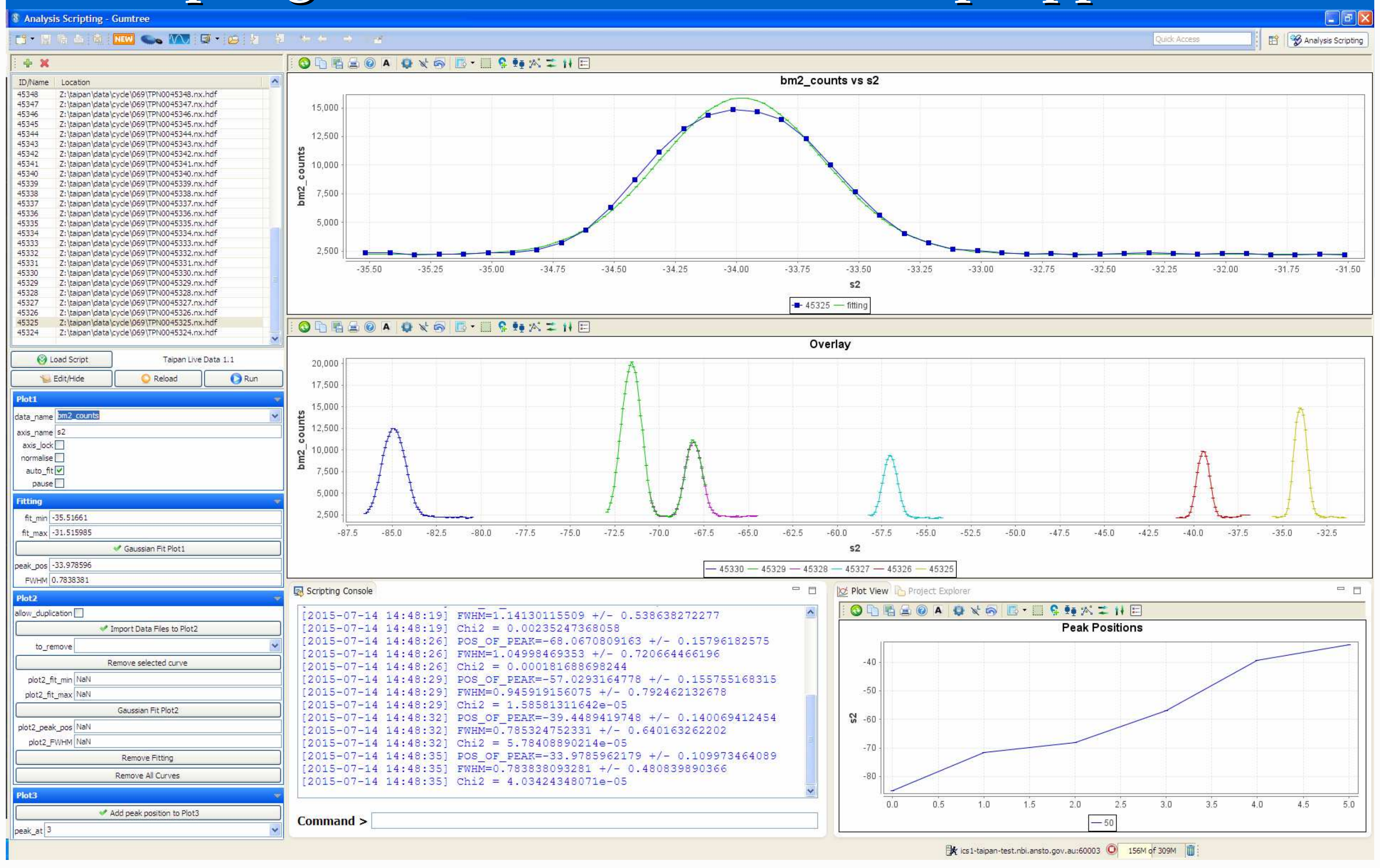
Features

- Scientific Computing Library: More than *numpy*
  - multi-dimensional array object
  - mathematical operations
  - error propagation
  - carrying axes information
- Library to Read Data from Different Sources
- Plotting Library
  - 1-D and 2-D plotting
  - real-time data plotting
  - fitting support
- Instrument Control Module
- Auto-GUI Support
  - generic GUI interface
  - desktop implementation
  - web service implementation

Scripting Can Be Used to

- Treat Experiment Data
  - preview data and send feedback to experiment
  - treat data on a local computer or on the server
- Align Instrument
- Create Quick Experiment Interfaces
- Make Highly Customisable Features

Scripting with Auto-GUI in Desktop Application



```
[2015-07-14 14:48:19] FWHM=1.14130115509 +/- 0.538638272277
[2015-07-14 14:48:19] Chi2 = 0.00235247368058
[2015-07-14 14:48:26] POS_OF_PEAK=-68.0670809163 +/- 0.15796182575
[2015-07-14 14:48:26] FWHM=1.04998469353 +/- 0.720664466196
[2015-07-14 14:48:26] Chi2 = 0.000181688698244
[2015-07-14 14:48:29] POS_OF_PEAK=-57.0293164778 +/- 0.155755168315
[2015-07-14 14:48:29] FWHM=0.945919156075 +/- 0.792462132678
[2015-07-14 14:48:29] Chi2 = 1.58581311642e-05
[2015-07-14 14:48:32] POS_OF_PEAK=-39.4489419748 +/- 0.140069412454
[2015-07-14 14:48:32] FWHM=0.785324752331 +/- 0.640163262202
[2015-07-14 14:48:32] Chi2 = 5.78408890214e-05
[2015-07-14 14:48:35] POS_OF_PEAK=-33.9785962179 +/- 0.109973464089
[2015-07-14 14:48:35] FWHM=0.783838093281 +/- 0.480839890366
[2015-07-14 14:48:35] Chi2 = 4.03424348071e-05
```

Command >

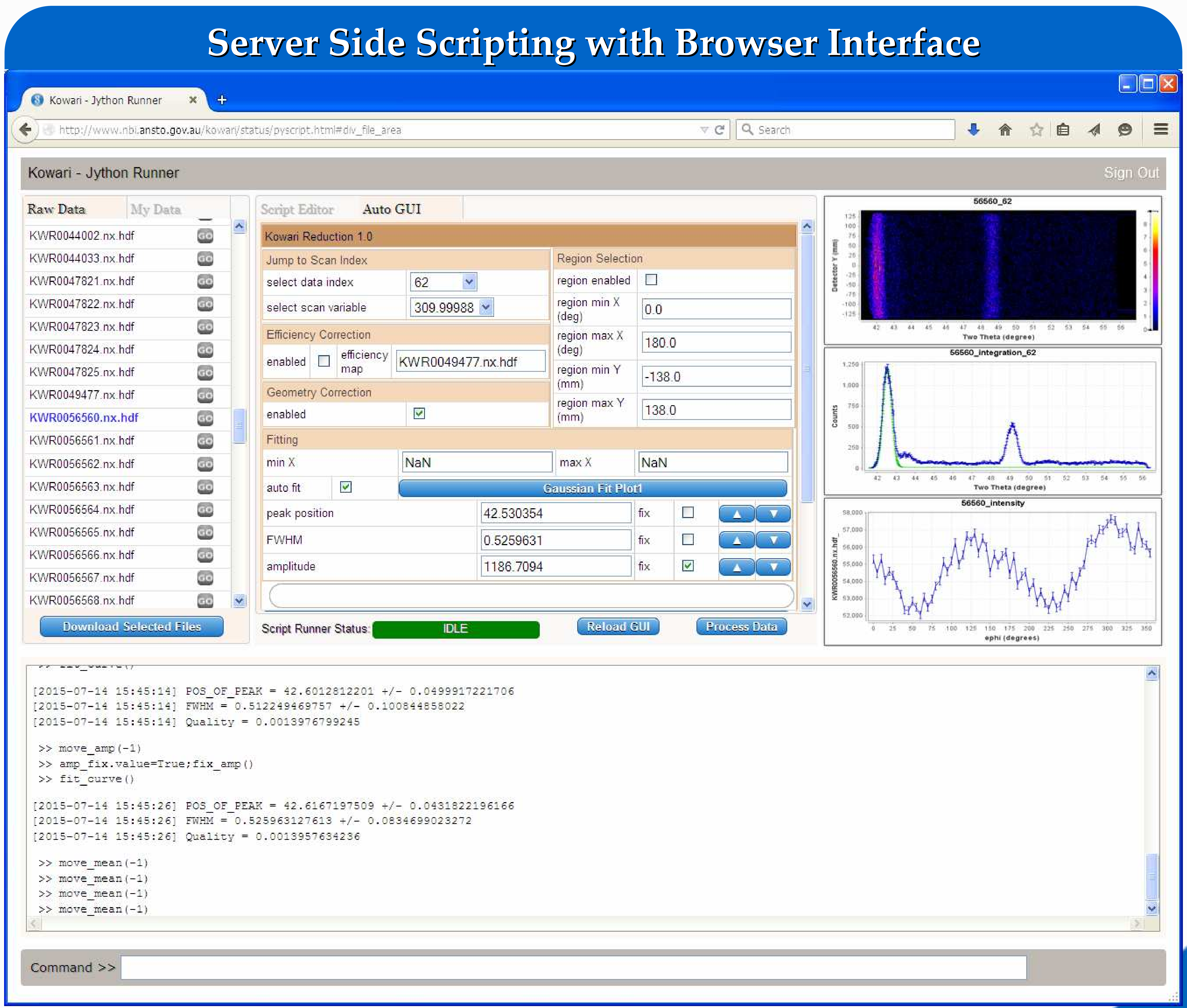
Sample Code - Arrays

```
>>> from gumpy.nexus import *
// create a 2D array, with dimension: 3x4
>>> arr = arange(0, 90.0, 90.0/12, [3, 4])
// multiply a random array
>>> arr *= rand([3, 4])
// print the array, with squares of errors and axes
>>> print arr
Dataset(Array([[0.000000, 3.578325, 13.678842, 1.894
               [17.238988, 2.261792, 13.606324, 40.2
               [27.324902, 6.201477, 3.092464, 71.38
        var=Array([[0.000000, 28.544689, 217.656676,
                   [527.075736, 84.953608, 616.39862
                   [1651.938281, 419.169446, 232.062
        axes=[SimpleData(Array([0, 1, 2])),
              SimpleData(Array([0, 1, 2, 3]))])
// do an integration along the 2nd dimension
>>> res = arr.intg(1)
>>> print res
Dataset(Array([19.151623, 73.382659, 108.005331]),
        var=Array([288.986143, 3373.792127, 8254.325
        axes=[SimpleData(Array([0, 1, 2]))])
// plot the result
>>> Plot1.set_dataset(res)
```

Sample Code – Auto GUI

```
// set UI label of the control panel
>>> __script__.title = 'Kowari Reduction'
// set the number of columns of the control panel
>>> __script__.numColumns = 2
// create a new group: a collection of parameters
>>> g_jump = Group('Jump to Scan Index')
>>> g_reg = Group('Region Selection')
// set the height of the group
>>> g_reg.rowspan = 3
>>> g_eff = Group('Efficiency Correction')
// set the number of columns in the group
>>> g_eff.numColumns = 2
// create a new parameter, giving type, initial
// value, and command to run when changing the value
>>> ind_jump = Par('int', -1, options = [],
                   command = 'jump_to_index()')
// set the UI label of the parameter
>>> ind_jump.title = 'select data index'
>>> var_jump = Par('float', float('NAN'), [],
                   command = 'jump_to_var()')
>>> var_jump.title = 'select scan variable'
// add parameters to a group
>>> g_jump.add(ind_jump, var_jump)
```

Server Side Scripting with Browser Interface



Kowari - Jython Runner

KWR0044002.nx.hdf
KWR0044033.nx.hdf
KWR0047821.nx.hdf
KWR0047822.nx.hdf
KWR0047823.nx.hdf
KWR0047824.nx.hdf
KWR0047825.nx.hdf
KWR0049477.nx.hdf
KWR0056560.nx.hdf
KWR0056561.nx.hdf
KWR0056562.nx.hdf
KWR0056563.nx.hdf
KWR0056564.nx.hdf
KWR0056565.nx.hdf
KWR0056566.nx.hdf
KWR0056567.nx.hdf
KWR0056568.nx.hdf

Kowari Reduction 1.0

Jump to Scan Index
select data index: 62
select scan variable: 309.99988

Efficiency Correction
enabled □ efficiency map: KWR0049477.nx.hdf

Geometry Correction
enabled ☑

Region Selection
region enabled □
region min X (deg): 0.0
region max X (deg): 180.0
region min Y (mm): -138.0
region max Y (mm): 138.0

Fitting
min X: NaN max X: NaN
auto fit ☑

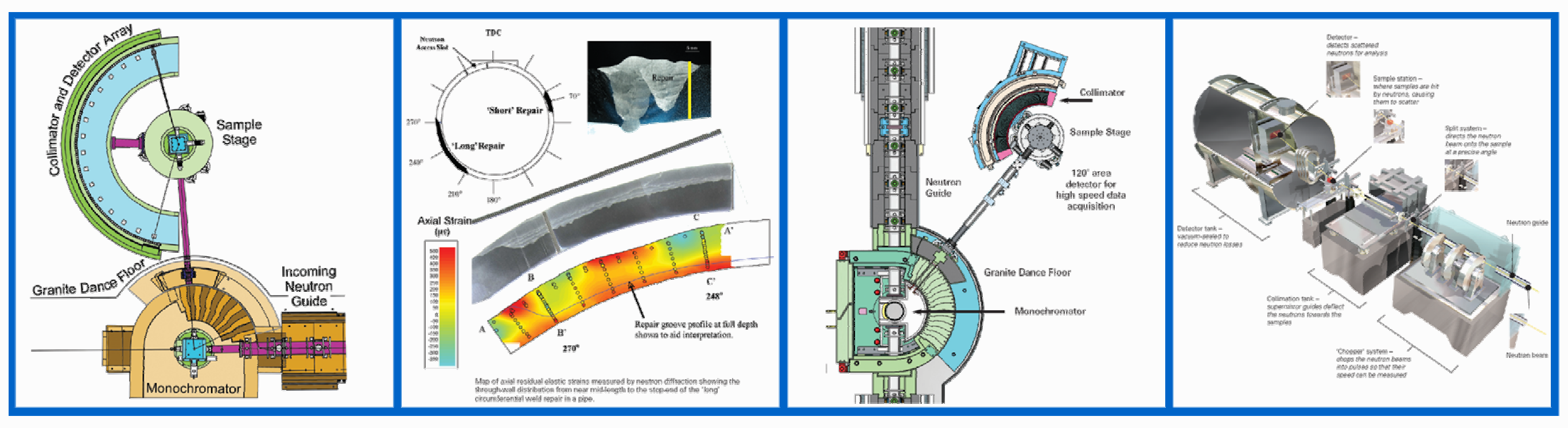
peak position	42.530354	fix	□
FWHM	0.5259631	fix	□
amplitude	1186.7094	fix	☑

Script Runner Status: IDLE

```
[2015-07-14 15:45:14] POS_OF_PEAK = 42.6012812201 +/- 0.0499917221706
[2015-07-14 15:45:14] FWHM = 0.512249469757 +/- 0.100844858022
[2015-07-14 15:45:14] Quality = 0.0013976799245

 >> move_amp(-1)
 >> amp_fix.value=True;fix_amp()
 >> fit_curve()

[2015-07-14 15:45:26] POS_OF_PEAK = 42.6167197509 +/- 0.0431822196166
[2015-07-14 15:45:26] FWHM = 0.525963127613 +/- 0.0834699023272
[2015-07-14 15:45:26] Quality = 0.0013957634236

 >> move_mean(-1)
 >> move_mean(-1)
 >> move_mean(-1)
 >> move_mean(-1)
```

Command >>

OPAL
WORLD-CLASS NEUTRON SCIENCE



Bragg Institute