Available online at www.iseec2012.com

I-SEEC 2012

Proceeding - Science and Engineering (2013) 135–142

Proceeding
Science and Engineering

www.iseec2012.com

Science and Engineering Symposium
4th International Science, Social Science, Engineering and Energy Conference 2012

Jpowder version 2: A Java-based program for the display and examination of powder diffraction data

K. Puphaiboon[a,*], A. Markvardsen[b], M. Arjeneh[b]

[a]*Faculty of Computer and Technology, Kasem Bundit University, Bangkok, 10250, Thailand*
[b]*Data Analysis Group, STFC Rutherford Appleton Laboratory, Didcot, OX11 0QX, UK*

Abstract

The ability to display and inspect powder diffraction data quickly and efficiently is a central part of the data analysis process. Whilst many computer programs are capable of displaying powder data, their focus is typically on advanced operations such as structure solution or Rietveld refinement. Jpowder was first presented in J. Appl. Cryst. (2010). 43, 1532-1534 as a new lightweight powder diffraction visualization program with the capability of fast rendering to compare powder data sets. Based on the success and user requests, Jpowder has since been extended to display series of powder diffraction patterns in 3D and new options for creating interactive plots. With Jpowder added (e.g. XML, drag and drop, grey scale and etc) to the latter, for example, it can be used to improve visualization and create e-learning web content for Crytallography.

© 2013 The Authors. Published by Kasem Bundit University.
Selection and/or peer-review under responsibility of Faculty of Science and Technology, Kasem Bundit University, Bangkok.

Keywords: Powder diffraction data; Java; Open Source; E-learning web content

1. Introduction

We have had many user requests since the first release of Jpowder. Most of these have been requests for additional features including problems with running Jpowder on various architecture (in particular on Mac PCs) and reports of minor bugs. So the good news is that we have had a healthy interest. However, there are some reports that users have found problems with the first version of Jpowder. A number of users have found that features were missing to make this application better suit their needs. The two areas where we experienced the strongest requests were to allow Jpowder to plot series of powder datasets 3D and to make it easier to create E-learning web content. The largest task has been to extend Jpowder to plot 3D, which has been implemented together with new tools for manipulation such charts. The 2D plotting has also been modified according to user requests and the application has been extensively tested on multiple platforms.

* Corresponding author. *E-mail address:* kreecha_pu@yahoo.com

2. 3D Plotting

In the plotting area of Jpowder, two tabs have been added as highlighted in Figure (2D3Dtabs).

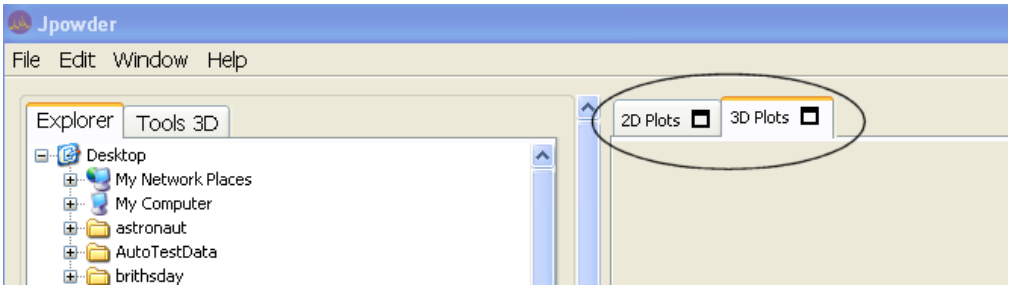

Fig. 1. 2D and 3D tabs

Clicking on the 2D Plots tab keeps the functionality for plotting data 2D as was presented with the release of Jpowder v1. Selecting the 3D Plots tab allows the user to plot data 3D. When a powder diffraction dataset is collected, additional information about this dataset may be stored in log books. This could be the temperature and pressure or the time at which data were collected. Here we refer to such information as meta data, including the name of the data file. The scientific case for allowing 3D plotting is to enable a series of powder diffraction datasets collected with different meta information to be plotted 3D such that changes in the datasets as a function of meta data changes can be visually easily identified.

To initiate the generation of a 3D plot, the user can drag and drop files from a file explorer of the operating system, or from the file explorer build into Jpowder, into the 3D plot area. This will open a window entitled “Prepare 3D Plot”, as demonstrated in Figure (prepare3Dplot), which allows the user to adjust the order datasets listed in the “Prepare 3D Plot” table, add meta data information and select the meta data to plot the datasets as a function.

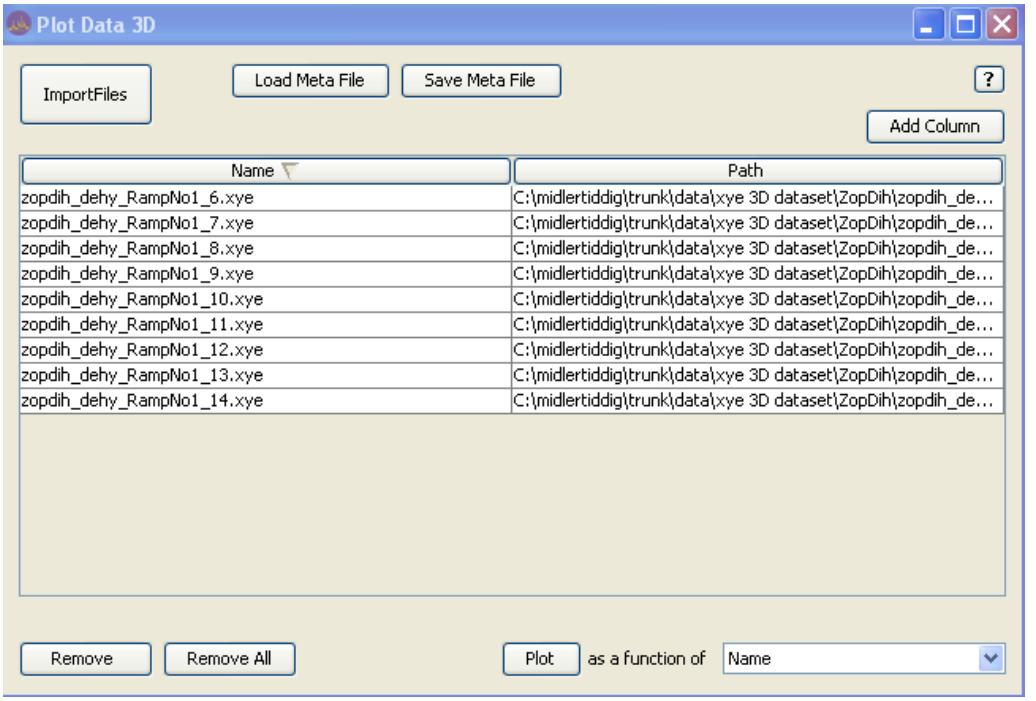

Fig. 2. Prepare 3D plot

The default option is to plot the datasets as a function of ‘name’, that is the filename of the individual datasets. This generates a 3D plot where the datasets are stacked on top of each other as demonstrated in Figure (namePlot) where the bottom horizontal grey-scale colour strip plots the first dataset listed in the “Prepare 3D Plot” table.

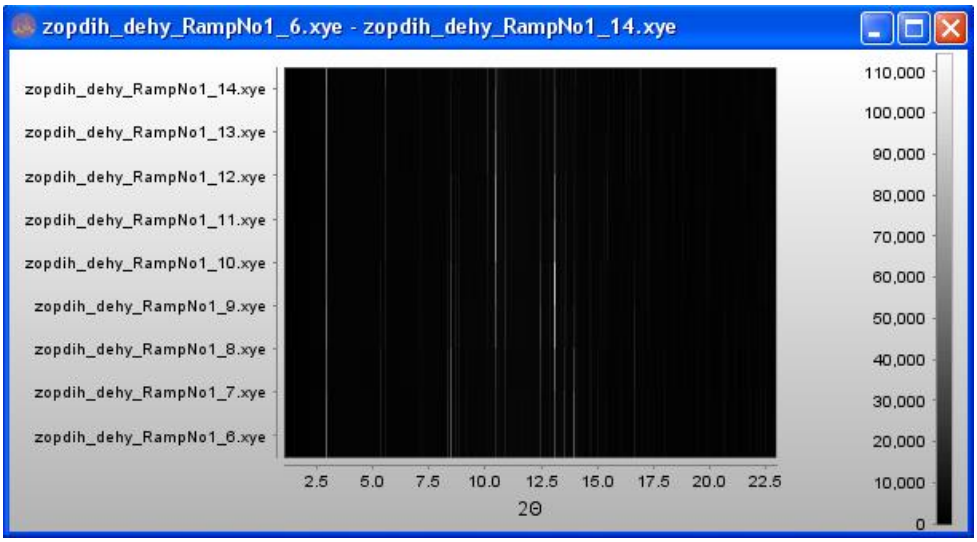

Fig. 3. Name Plot

Meta data are added to the “Prepare 3D Plot” table by clicking on the Add Column button, which pops up a small window where the user can specify the name of the meta data, for example, temperature or the date-and-time when datasets were collected, and a new column is added where either numbers or labels (strings) can be entered. Figure (prepare3DplotShowMeta) demonstrates an example where two such columns have been added.

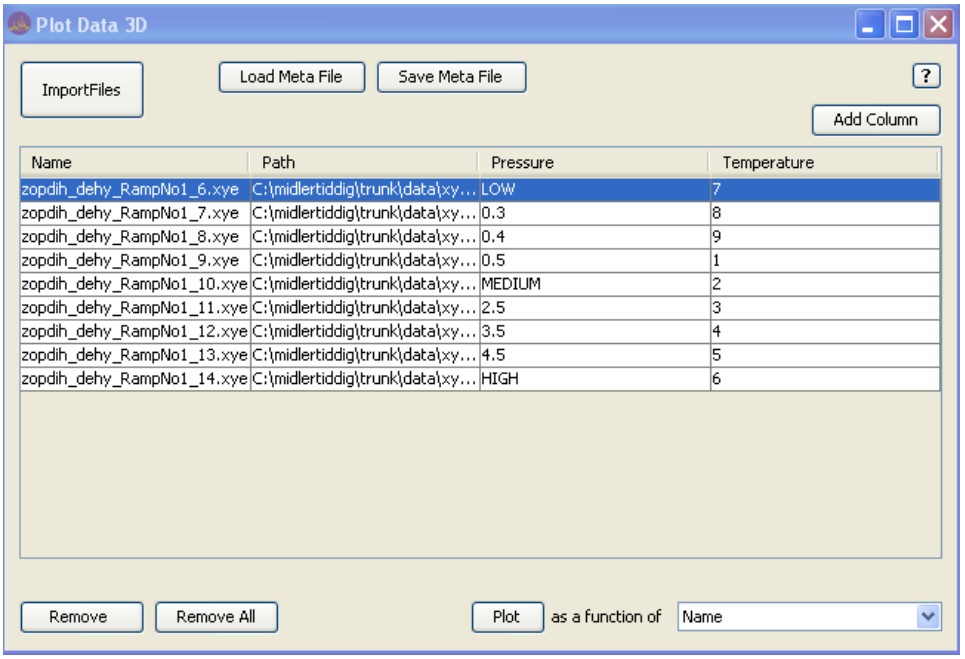

Fig. 4. Prepare 3D Plot to Show Meta

If any of the fields of a column contain a label rather than a number, as for example for the Pressure column in Figure (prepare3DplotShowMeta), then all the entries in that column are treated as labels. Only one 3D plot option is then available, which is the stack plot option demonstrated in Figure (namePlot). If all the fields of a meta column are numbers, the default plot is still a stack plot, but additional options will then be available for adjusting how the datasets are plotted as will be demonstrated in the next section.

To avoid having the user retype meta data, the "Prepare 3D Plot" window have two buttons: Save Meta Data and Load Meta Data. An example of a XML meta data file output is shown below:

```
<JpowderMetaFile>
  <dataset name="Data1.xye" path="C:\blah...">
     <meta name=”Temperature” val="10" unit=”” />
    <meta name=”Pressure” val="50" unit=”” />
  </dataset>
  <dataset name="Data2.xye" path="C:\blah...">
     <meta name=”Temperature” val="20" unit=”” />
     <meta name=”Pressure” val="40" unit=”” />
  </dataset>
  <dataset name="Data3.xye" path="C:\blah...">
     <meta name=”Temperature” val="40" unit=”” />
     <meta name=”Pressure” val="10" unit=”” />
  </dataset>
  <dataset name="Data4.xye" path="C:\blah...">
     <meta name=”Temperature” val="100" unit=”” />
     <meta name=”Pressure” val="1" unit=”” />
  </dataset>
  <default plot-as="Temperature" />
</JpowderMetaFile>
```

This is our default XML meta data format. In addition to this easily expandable XML format, a user request has been that Jpowder also supported, a XML format, more easily editable in a text editor of their choice. Hence for this reason Jpowder also accept the following meta data format:

```
<JpowderMetaFile>
Name       Temparature    Pressure        Path
Data1.xye       10             50           C:\blah...
Data2.xye       20             40           C:\blah...
Data3.xye       40             10           C:\blah...
Data4.xye       100            1            C:\blah...
<default plot-as="Temperature" />
</JpowderMetaFile>
```

One advantage of using XML is that a user may insert anywhere comments in the file using the notation <!-- comment -->. Also, Jpowder looks to see if the following XML tags are present:

```
<plot-window title="" />  <!-- This overwrite the default windows title
                              of the window framing (containing) the 3D
                              plot(chart) -->
<plot title="" />  <!-- If present will be display this title
                        above the plotted data -->
<plot description="" />  <!-- Optionaly longer description of plot which
                              may be provided -->
```

Note dragging a XML meta file directly into Jpowder will generate a 3D plot as specified in the meta file.

2.1. Jpowder 3D Tools

In association with allowing 3D plotting, Jpowder, has been fitted with a new set of tools for manipulating such plots. Note the details of how this is going to look has not been finalised yet but will be ready before the start of the I-SEE conference at Kasem Bundit University. However parts of these tools have been implemented according to the Balsamiq mock-up shown below:

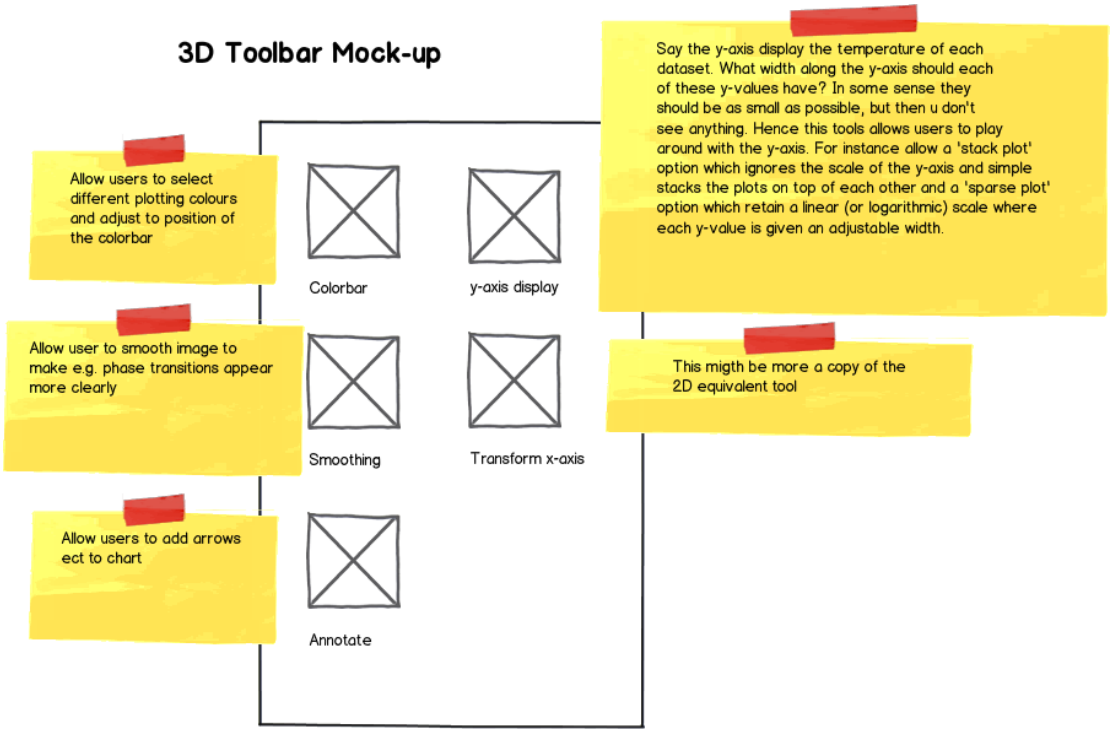

Fig. 5. 3D tools

And in more details, for example the ‘y-axis display’ 3D tool will be implemented according to:

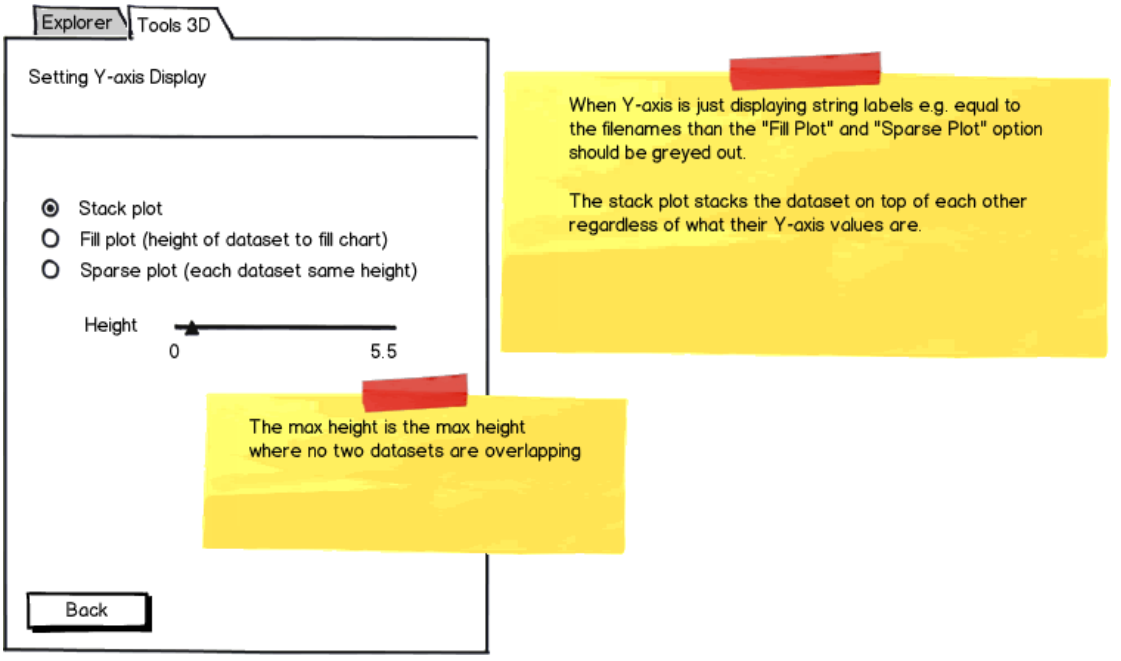

Fig. 6. Y Axis Display

2.2. Supported file formats

The file formats which are supported for this version of *Jpowder* are:

- *XY*-format (.xy): Two-column ASCII format, where the data in the 1st column are assumed to be the diffraction angle in 2θ and those in the 2nd column are X-ray or neutron counts.
- *XYE*-format (.xye): As for *XY*-format, but with an additional column listing the estimated standard deviations for the counts. This format also supports the use of a single real number, specifying a wavelength, as the first line of the file.
- *CIF*-format. The Crystallographic Information File format is described in detail at www.iucr.org/resources/cif.
- *ISIS GSAS*-format. At the STFC ISIS Facility, *GSAS* is frequently used to refine structures against time-of-flight neutron powder data. The current version of *Jpowder* requires such data to be in what the GSAS manual refers to as "RALF/FXYE format". When reading such a time-of-flight .gss file, the *x*-values are assumed to be in given in units of micro-seconds.

Jpowder will attempt to read files with extensions .gss and .cif as *ISIS GSAS*-format and *CIF*-format files respectively. For files with any other extension, *Jpowder* will attempt to read the file in *XY* or *XYE* format. All data are assumed to be constant wavelength data with the exception of data using the *ISIS GSAS*-format. The *Jpowder* code is Open Source and structured in such a way that it is straightforward for anyone used to Java programming to extend the list of supported file types.

3. Software and hardware environment

Jpowder will run on any PC that supports a Java Runtime Environment. Specifically, version 1.6 or higher is required for the version of *JPowder* described in this paper; however, if a JRE version older than 1.6 (i.e. older than Dec 11th, 2006) is installed on the local machine, the user is directed to the JRE update web page.

By default, Java Web Start applications run in a restricted environment, known as a sandbox - meaning they cannot access local files or the network. *Jpowder* provides functionality that goes beyond that allowed in the sandbox by signing the application's JAR files; this means that the user will need to accept a digitally signed certificate upon first installation, as depicted in Figure 3. This model enables Java Web Start applications to be installed by the end-user in an environment without administrative permissions, such as a university department where students have roaming accounts.

Jpowder versions can be updated for all users worldwide simply by providing an updated JAR file on the web server. On each user's computer, Java Web Start checks the web server for updates when the application runs. Resources are cached locally, for improved performance, but this also confers the advantage that they can function independently during any network/internet outages.

The code is licensed under the GNU General Public License v3.

4. Documentation and availability

Jpowder is downloadable from http://www.jpowder.org. The source code is available to download from http://code.google.com/p/jpowder and bugs can be reported via http://code.google.com/p/jpowder/issues. The *Jpowder* discussion group at http://groups.google.com/group/jpowder provides a forum for the discussion of *Jpowder*-related issues, such as feature requests.

Acknowledgements

KP gratefully acknowledges the support of Faculty of Science and Technology at Kasem Bundit University.

References

[1] Altomare, A., Camalli, M., Cuocci, C., Giacovazzo, C., Moliterni, A. and Rizzi, R. (2009). J. Appl. Cryst., 42, 1197-1202
[2] Boultif, A. and Louer, D. (2004). J. Appl. Cryst., 37, 724-731.Coelho, A. A. (2003). TOPAS User Manual, v3.1. Bruker AXS GmbH, Karlsruhe
[3] David, W. I. F., Shankland, K., van de Streek, J., Pidcock, E., Motherwell, W. D. S. & Cole, J. C. (2006). J. Appl. Cryst. 39, 910-915.
[4] Dong, C. (1999). J. Appl. Cryst. 32, 838.EVA. (2010). http://www.bruker-axs.de/eva_software.html
[5] Favre-Nicolin, V. and Cerny, R. (2002). J. Appl. Cryst. 35 734-743.
[6] Highscore. (2010). http://www.panalytical.com
[7] Jmol. (2008). Jmol: an open-source Java viewer for chemical structures in 3D,http://www.jmol.org.
[8] Kiosk (2010). http://www.pdb.org/pdb/Viewers/RCSBViewers/view.jsp?viewerType=KS
[9] Larson, A.C. and Von Dreele, R.B. (1994). Los Alamos National Laboratory Report LAUR 86-748.
[10] Lutterotti, L. and Bortolotti, M. (2003). IUCr Commission on Crystallographic Computing Newsletter, 1, 43-50.
[11] Petricek,V., Dusek, M. and Palatinus, L.(2006). Jana2006. The crystallographic computing system. Institute of Physics, Praha, Czech Republic
[12] Powder3D. (2008). http://www.fkf.mpg.de/xray/html/powder3d.html
[13] Putz, H., Schoen, J. C. and Jansen, M. (1999). J. Appl. Cryst., 32, 864-870.
[14] Rodriguez-Carvajal, J. (1993). Physica B., 192, 55-69.

Appendix A.

Table 1 A list of available keyboard shortcuts in Jpowder

Keys	Action
Ctrl+O	Open File chooser window
Ctrl+P	Print
Ctrl+Alt+P	Print for publication
Ctrl+Alt+A	Close all windows within the main plot area
Ctrl+Z	Undo closed window
Ctrl+Y	Redo closed command
Ctrl+C	Copy currently selected window to clipboard
Ctrl+E	Open the properties window
F1	Open online documentation and support
Ctrl+A	Open 'About' window
Ctrl+LeftClick	Move the plots