THE APPLICATION OF AN ONLINE DATA VISUALIZATION TOOL, PTPLOT, IN THE WORLD DATA CENTRE (WDC) FOR SOLAR-TERRESTRIAL SCIENCE (STS) IN IPS RADIO AND SPACE SERVICES, AUSTRALIA

K Wang[1*] and C Yuile[2]

IPS Radio and Space Services, Bureau of Meteorology,
Level 15, Tower C, 300 Elizabeth Street, Surry Hills, NSW, Australia 2010
*[*1]Email: mkw@ips.gov.au*
[2]Email: colin@ips.gov.au

ABSTRACT

Ptplot is a set of two dimensional signal plotters components written in Java with multiple properties, such as being embeddable in applets or applications, utilizing automatic or manual tick marks, logarithmic axes, infinite zooming and much more. The World Data Centre of IPS applies Ptplot as a multiple function online data plot tool by converting various text format data files into Ptplot recognizable XML files with the AWK language. At present, Ptplot has allowed eight archived solar-terrestrial science data sets to be easily plotted, viewed and downloaded from the IPS web site.

Keywords: Ptplot, Ptolemy Project, Plot, Data Visualization, Infographics, World Data Centre, IPS, Solar-Terrestrial Science, Magnetometer

1 INTRODUCTION

The rapid development of sensor, storage and networking technology has resulted in a huge increase in the retrieval and archival of scientific data. Data analysis and data mining is becoming increasingly important and challenging. Along with the rapid increase in data volumes, more powerful and efficient data processing technology and tools are being developed. Data plotting is a key aspect of data visualization and infographics technology and a useful method for a data analyst to locate interesting data.

In data processing, "computer users spend a lot of time doing simple, mechanical data manipulation – changing the format of data, checking its validity, finding items with some property, adding up numbers, printing reports, and the like. AWK is a programming language that makes it possible to handle such tasks with very short programs, often only one or two lines long" (Hao, Kernighan & Weinberger, 1988).

Ptplot (http://ptolemy.eecs.berkeley.edu/java/ptplot/) is a 2D data plotter and histogram tool implemented in Java. Ptplot can be used as a standalone applet or application, or it can be embedded in your own applet or application. It has properties: Embeddable in applets or applications; Auto-ranging; Automatic or manual labeling of axes; Automatic or manual tick marks; Logarithmic axes; Live, animated plots; Infinite zooming; Various plot styles: connected lines, scatter plot, bars, etc; Various point styles: none, dots, points, and unique marks; Multiple data sets and a legend; Color or black and white plotting; Error bars; Editable plots; PlotML, and XML language for specifying plots; Compatibility with pxgraph, an older plotting program. There are many interesting demonstrations, download links, and other useful information in its website.

Ptplot is a part of Ptolemy II (http://ptolemy.eecs.berkeley.edu/index.htm), but Ptplot is also available as a separate download. The latest version of Ptplot is Ptplot 5.8. The older versions, patches and extensions also can be downloaded from its web site. The Ptolemy II project is developed by the Electrical Engineering and Computer Science, College of Engineering, University of California Berkeley, USA.
IPS applies Ptplot as an online data plot tool to visualize various archived digital text files. Before a text file can be plotted with Ptplot, an AWK program is used to convert the text file into a temporary Ptplot recognizable XML file.

2 THE APPLICATION OF PTPLOT

IPS has made available eight different digital data archived in plain text format, as well as several other binary and image datasets. These eight data sets are able to be downloaded, or plotted and viewed online with Ptplot. This paper will demonstrate the process of converting and displaying magnetometer data using AWK and Ptplot.

2.1 The first webpage of the magnetometer online display

Figure 1 shows the main part of the first web page of the magnetometer data online display and download (http://www.ips.gov.au/World_Data_Centre/1/2). IPS has listed 14 magnetometers on its World Data Centre section web page. A magnetometer installed at Casey station in Antarctica will be used as an example to demonstrate the process displaying magnetometer data online.

After selecting a station, year, month, and day, a magnetometer plot consisting of x and y components will be displayed on clicking "Plot Graph" as shown in Figure 1. If the user is interested in a daily magnetometer file, it is able to be downloaded by clicking the "Download Data" button on the same page.

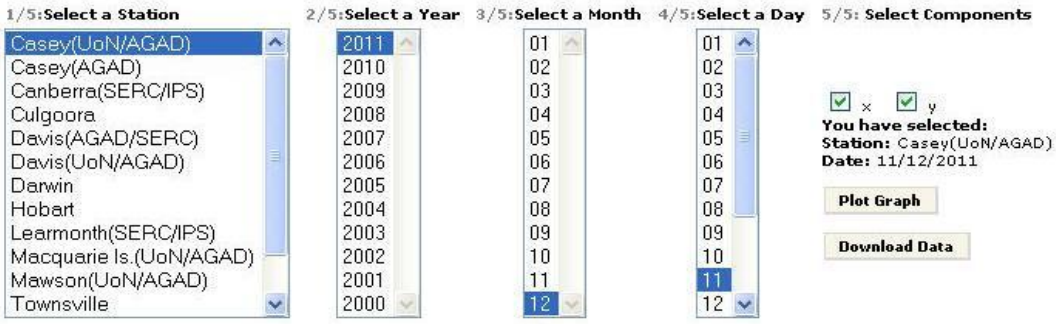

Figure 1. The first web page of the magnetometer online display and data download

2.2 The data format of the magnetometer text data file

Before the plot page is introduced, let us to start from the daily original magnetometer data file. The magnetometer data file was originally recorded at Casey in Antarctica and transferred to Sydney in near real-time. This data is automatically processed and archived as a daily text file. Figure 2 is a small part of the daily file recorded on 11th December of 2011 at Casey Station of Antarctica. The left three columns are Universal Time of hour, minute, and second respectively and the right two columns are x and y components of the magnetometer records respectively. There are two records every second. So there are about 172800 records in a daily file.

```
11 59 58.00   -69.00  -129.00
11 59 58.50   -79.00  -124.00
11 59 59.00   -84.00  -125.00
11 59 59.50   -84.00  -124.00
12  0  0.00   -78.00  -128.00
12  0  0.50   -73.00  -123.00
12  0  1.00   -70.00  -135.00
12  0  1.50   -72.00  -137.00
```

Figure 2. A small part of the original magnetometer text file recorded at Casey Station of Antarctica on 11th December 2011

2.3 Convert text file to XML file with an AWK program

Behind the front page of Figure 1, on the server side there is a process to convert the original text magnetometer data file to Ptplot recognizable XML file. Figure 3 is the main part of an AWK program that is used to dynamically make the conversion once a date is selected and the "Plot Graph" button is clicked on Figure 1.

```
BEGIN { i = 0
while (getline <"header.plt" > 0)
      print $0
}
{
if ($3 == "0.00") {
  h[i] = $1
  m[i] = $2
  x[1,i] = $4
  x[2,i] = $5
  i++
}
}
END {
print "<title>"b"</title>"
split(a,p,",")
for (n in p) {
   if (p[n] == "x") {
     print "<dataset name=\""p[n]"\">"
     for (j = 0; j < i; j++)
       print "   <p x=\""h[j]*60+m[j]".0\" y=\""x[1,j]"\"/>"
     print "</dataset>"
   }
   if (p[n] == "y") {
     print "<dataset name=\""p[n]"\">"
     for (j = 0; j < i; j++)
       print "   <p x=\""h[j]*60+m[j]".0\" y=\""x[2,j]"\"/>"
     print "</dataset>"
   }
}
print "</plot>"
}
```

Figure 3. The main part of an AWK program used to convert text magnetometer files into XML files

Because a daily file has about 172800 records, to read, convert and plot it would take a long time. In order to reduce the server side process time, and deliver a quick online plotting, one record in every minute, that is only 1/120 records will be used to plot. So the condition If ($3 == "0.00") in Figure 3 is used to selected records. When the third column (second) value of Figure 2 equals 0.00, its fourth (x component) and fifth (y component) column values are read and converted into XML file for further plotting.
Figure 4 is a small part of the XML file, which is saved in server side as a temporary file and is readable by the Ptplot Java applet. Each XML file consists of a head part and a data part. Each head part consists of lines of Title, xLabel, yLabel, xRange, yRange, default mark type and a series of xTicks, each xTick line consists of an x label value and a position value. The upper part of the Figure 4 shows the last two lines of the xTicks. Each data part consists of two datasets: x and y components. The lower part of Figure 4 shows the first four lines of the x component dataset of the XML file. The middle of the Figure 4 is the Title line, the title "Casey (UoN.AGAD) Magnetogram (nT) on 11/12/2011" will be shown in Figure 5.

```
    <tick label="23:58" position="1438.0"/>
    <tick label="23:59" position="1439.0"/>
</xTicks>
<title>Casey(UoN/AGAD) Magnetogram (nT) on 11/12/2011</title>
<dataset name="x">
    <p x="0.0" y="-24.00"/>
    <p x="1.0" y="-7.00"/>
    <p x="2.0" y="-140.00"/>
    <p x="3.0" y="-117.00"/>
```

Figure 4. A small part of the converted XML file, readable by the Ptplot Java applet

2.4 The properties and functions of Ptplot

On clicking the "Plot Graph" button (Figure 1), The PHP program will find the data file according to the selected Station, Year, Month and Day in Figure 1 and convert the text data file into a XML file with the AWK program shown in Figure 3 and an XML Head file. Ptplot reads the XML file and plots the data in a new web page. Figure 5 shows the new web page displaying the plot and other buttons above the plot. The plots of 'previous day' and 'next day' can be easily viewed by clicking the respective buttons (Figure 5). The "Download Data" button in the new webpage also can used to directly download the full data file.

The Ptplot provides four small buttons on its upper right corner. The left most button is used to print the plot. The plot is infinitely zoomed with the mouse and the second button is used to recover the plot after zooming. A

"Set plot format" window will pop up when the third button is clicked, allowing properties of the plot to be edited. The right most button rescales the plot to fit the data.

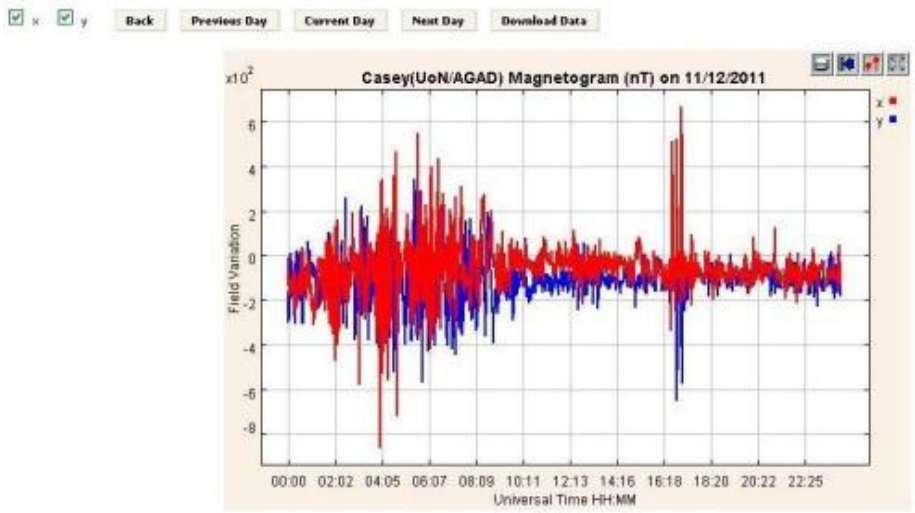

Figure 5. The plot web page of magnetometer data with other function buttons

3 CONCLUSION

The Ptplot Java applet is a powerful, efficient and versatile data visualization tool. Ptplot enables a user to make data plots without extensive programming experience and expensive commercial software. The only required programming is to write a script or a program to convert data files into a standard Ptplot recognizable XML file. IPS has applied Ptplot on eight data sets for public use. Over twelve thousand online plots are created each year by customers around the world who visit the website.

4 ACKNOWLEDGEMENTS

We are thankful to the Electrical Engineering and Computer Science, College of Engineering, University of California Berkeley, for the use of Ptolemy II.

5 REFERENCES

Hao, A.V., Kernighan, B.W. & Weinberger, P.J. (1988) The AWK Programming Language, New York: Addison-Wesley.