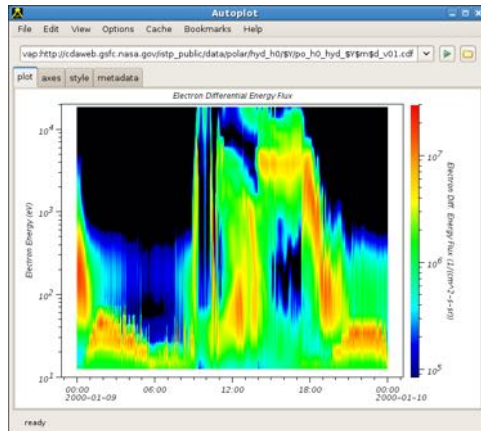


# Autoplot IHDEA 2019

Jeremy Faden

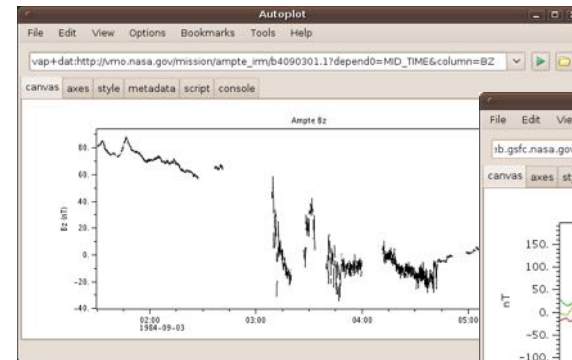
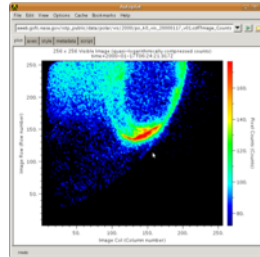
Cottage Systems, Iowa City, Iowa

U. of Iowa, Radio and Plasma Wave Group

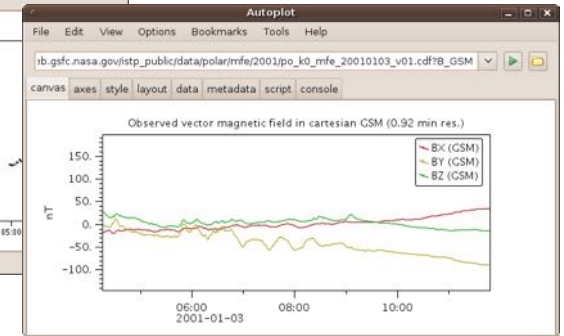


Spectral Time Series  
Flux(Time,En) from  
CDF file

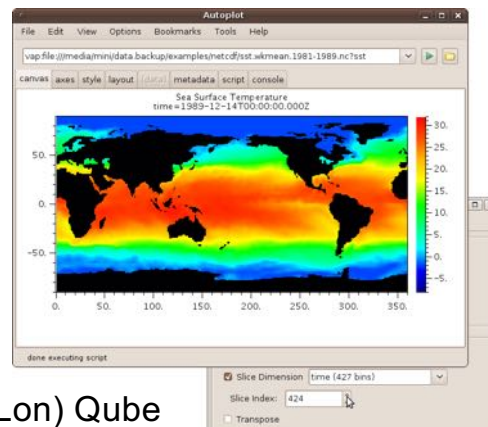
Image from CDF File



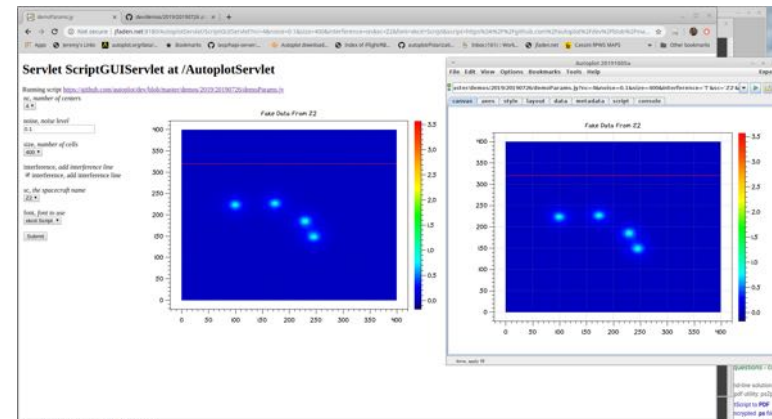
Scalar Time Series  
Bz(Time)  
from ASCII File



Vector Time Series  
from CDF File

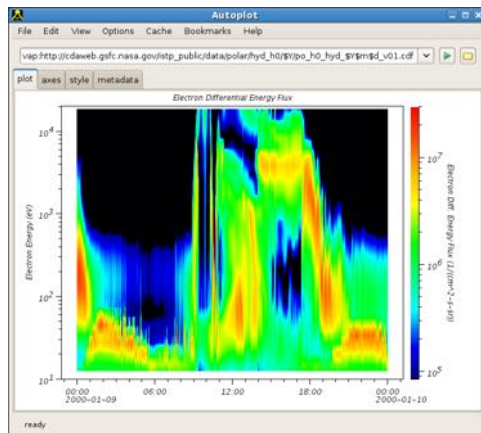


SST(Time,Lat,Lon) Qube  
from NetCDF File



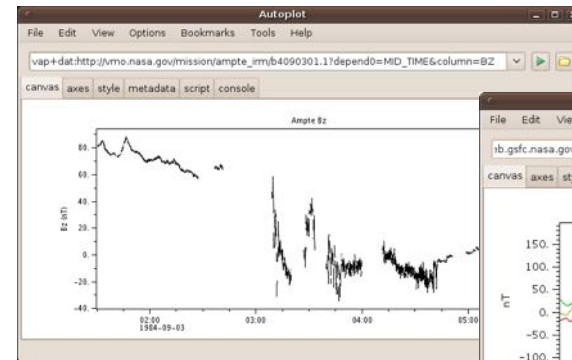
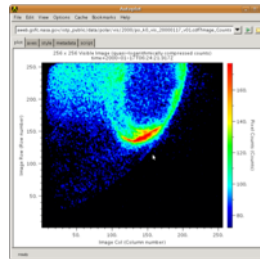
On-the-fly graphics  
from web servers

Autoplot is plotting software that plots data in local and remote files and from data servers.

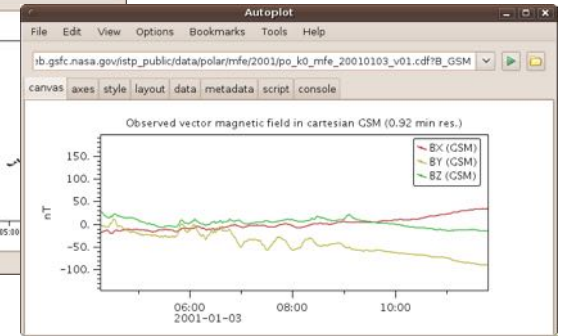


Spectral Time Series  
Flux(Time,En) from  
CDF file

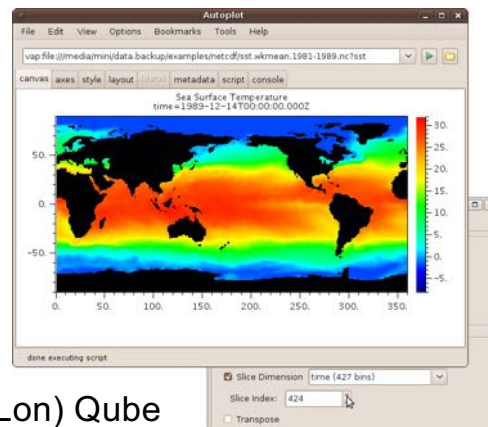
Image from CDF File



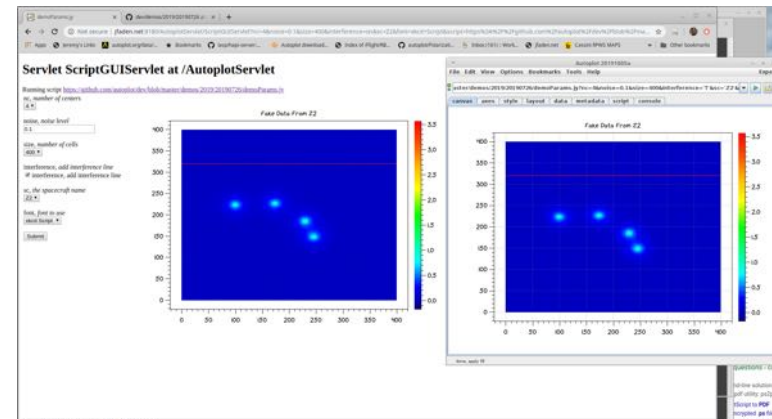
Scalar Time Series  
Bz(Time)  
from ASCII File



Vector Time Series  
from CDF File

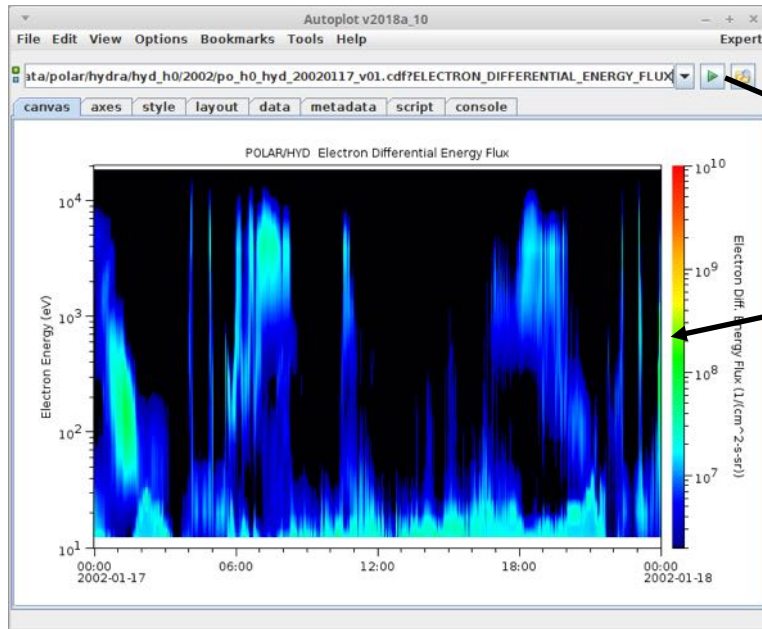


SST(Time,Lat,Lon) Qube  
from NetCDF File



On-the-fly graphics  
from web servers

# Using Autoplot



All data in Autoplot have a data address, or URI, identifying them. The address is entered in the address bar, and the data is loaded and displayed. Typically URIs are the names of data files and additional parameters needed to access.

The axes and plot are interactive: you can adjust settings like you would with Google Maps: the mouse wheel zooms and you can do box zoom, etc. There are horizontal and vertical slices as well.

(DEMO of pointing Autoplot to an ASCII file with spectrogram:  
<https://youtu.be/T6XMrwVuwUU>)

# Many Data Forms Supported

## File Formats

ASCII Tables  
CSV  
Binary Tables  
CDF  
NetCDF  
HDF

Wav Audio Files  
Excel Spreadsheets  
PNG images  
Export to IDLSav  
FITS  
Das2Streams  
QStreams

## Data Servers

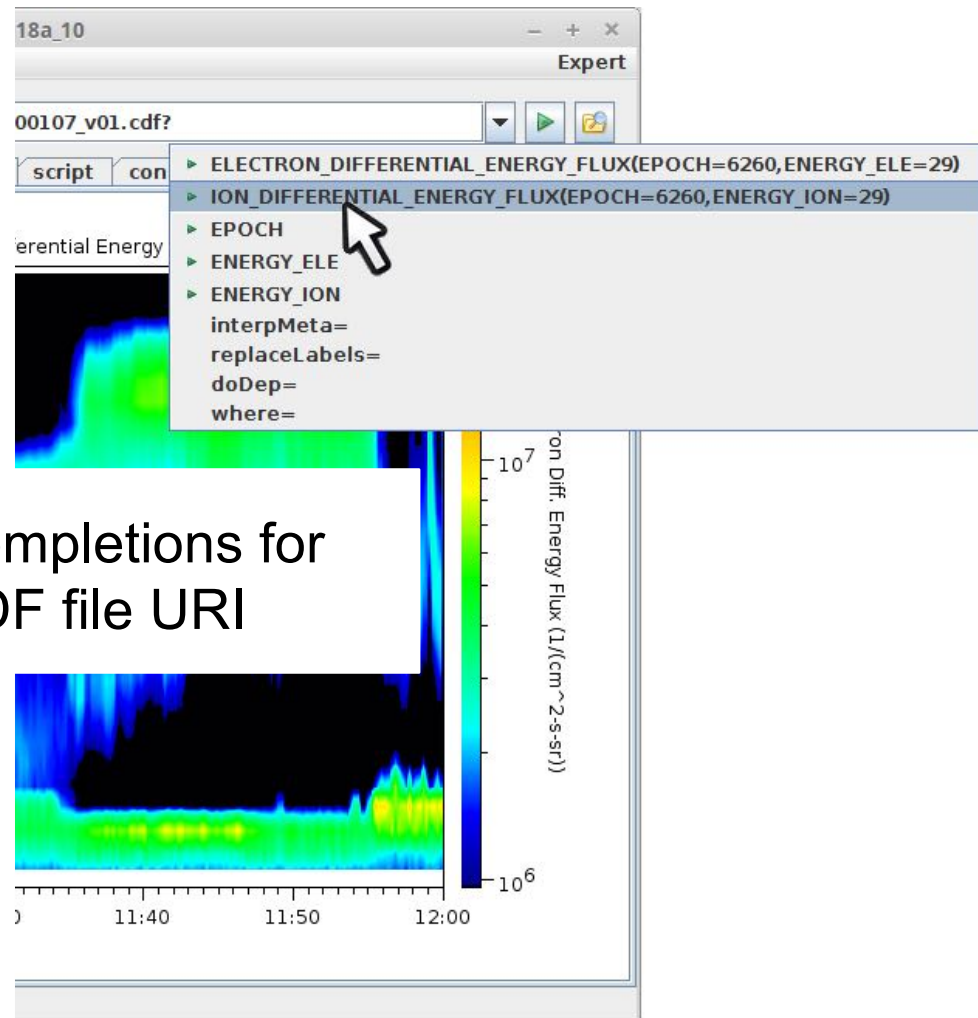
CDAWeb  
Das2  
Servers  
PDS-PPI  
HAPI

URIs can contain the URL of a file on a web site. Two collaborators can type in a URL of their data and see the same thing.

# Developing URIs

Typically URIs would be created by data providers, but scientists create them as well.

Data source plug-ins provide completions and a GUI to show what's available.

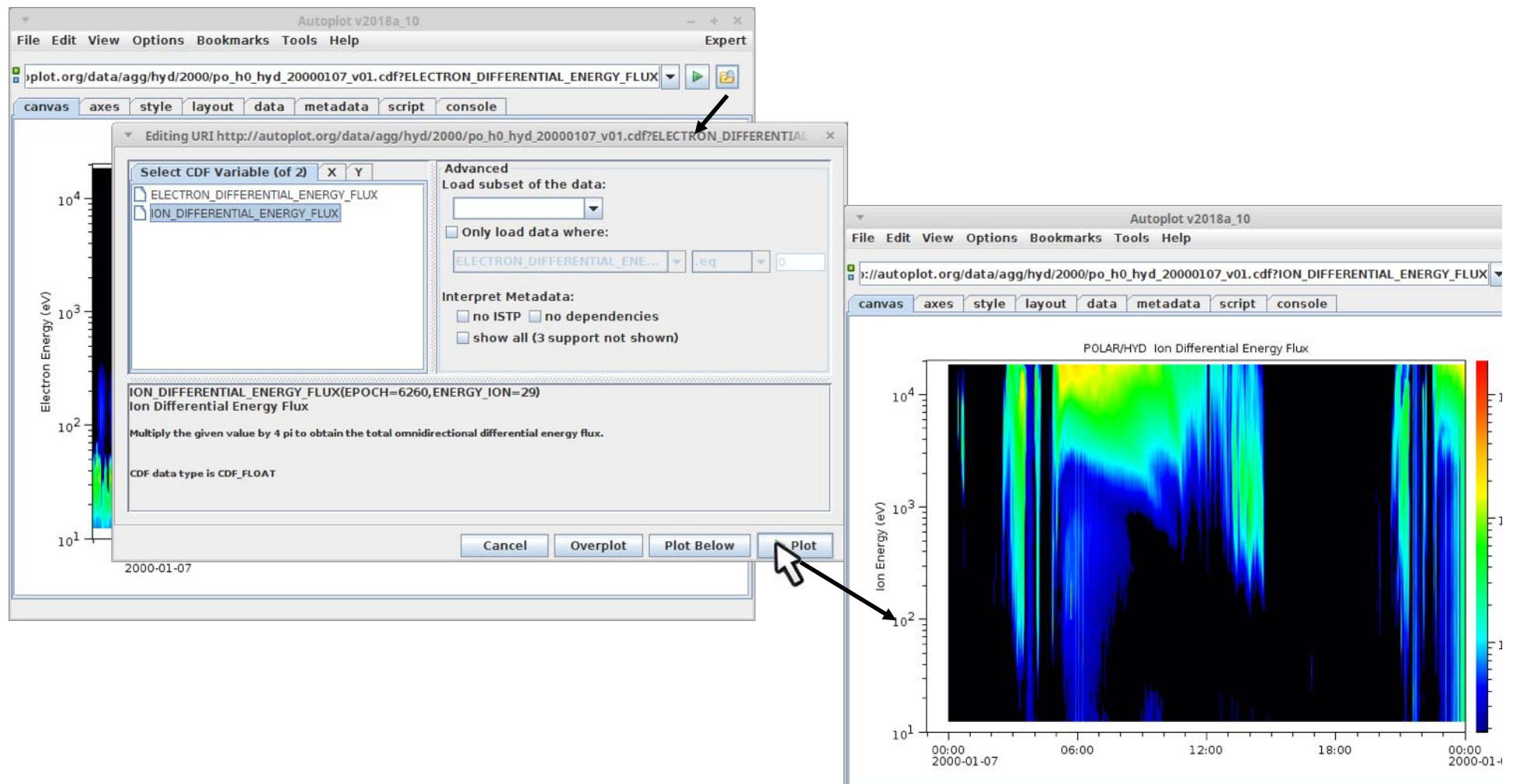


Completions for  
CDF file URI

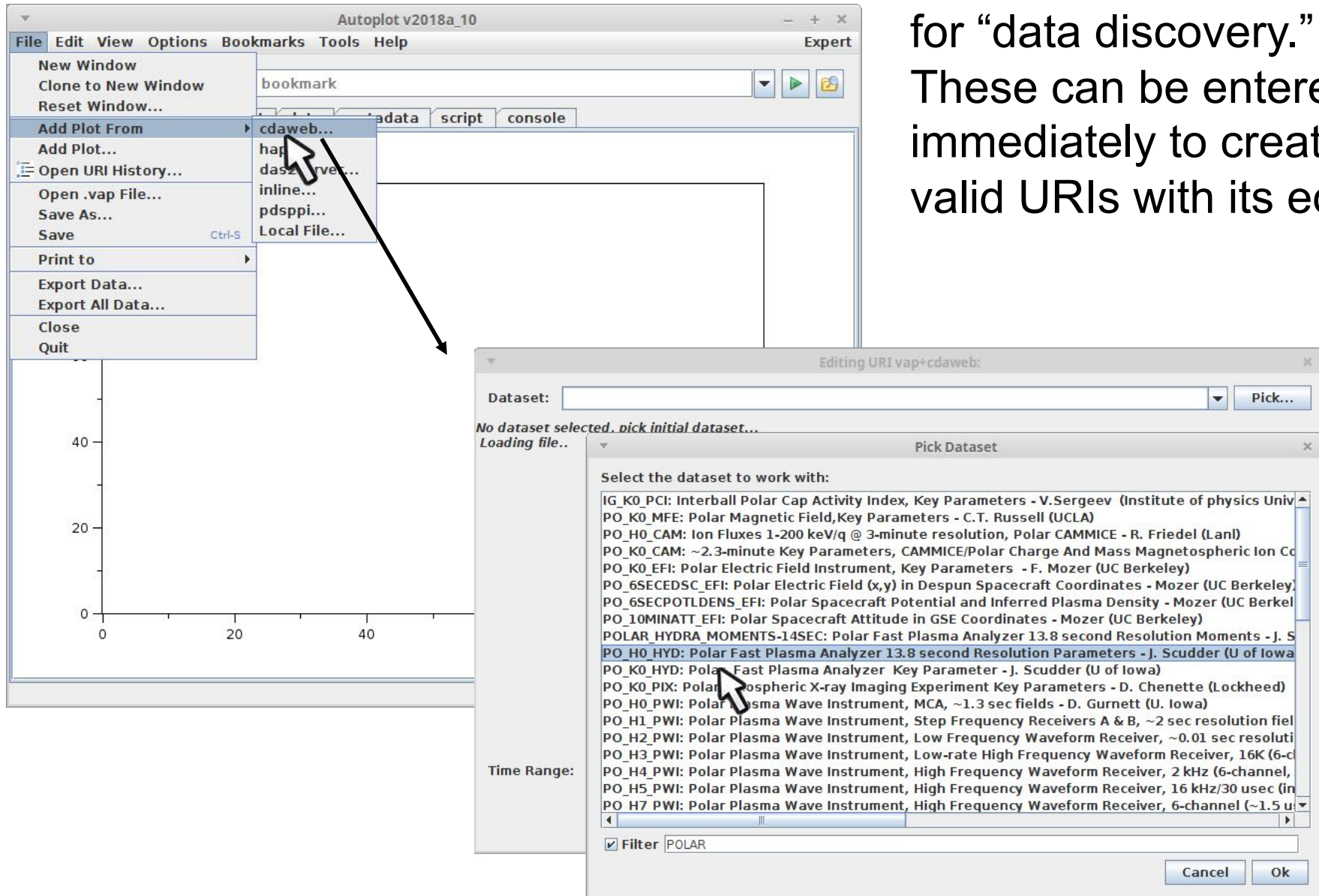


# Data Source Editors

Some URIs are complex, so editors make it so scientists don't have to think about them. The folder icon next to the URI bar enters the editor.

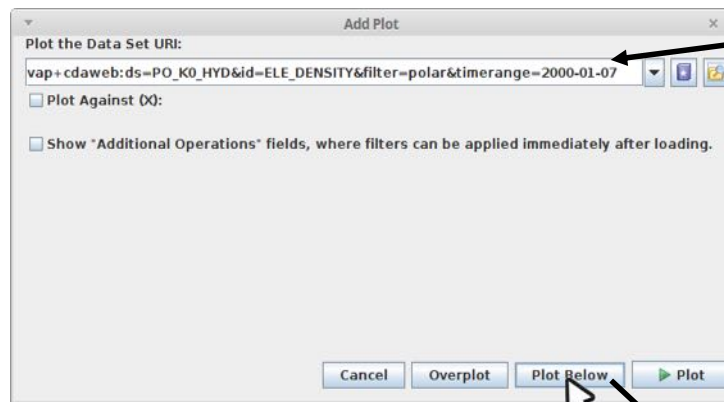


Some data sources allow for “data discovery.” These can be entered immediately to create valid URIs with its editor.

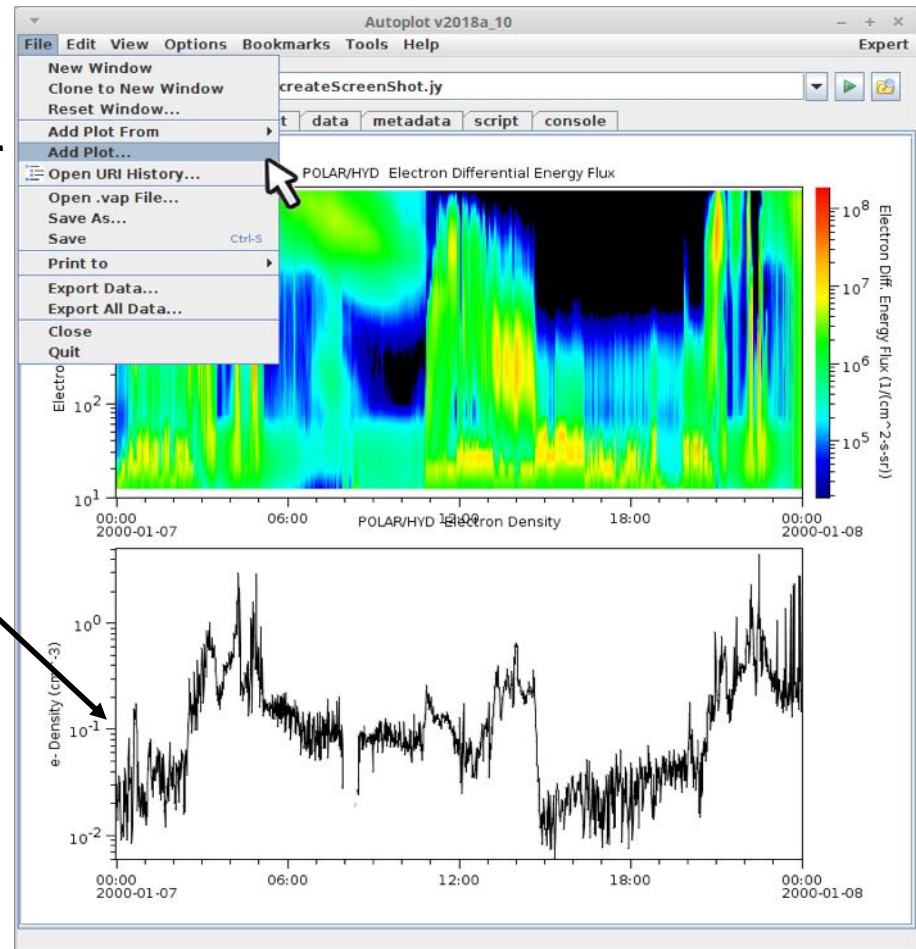


# More complex layouts

File→'Add Plot...' dialog is used to add additional plots.



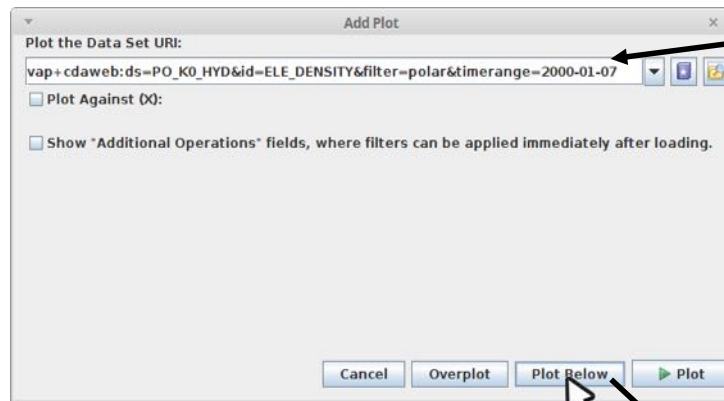
Time axes are automatically bound together, but they can be disconnected.





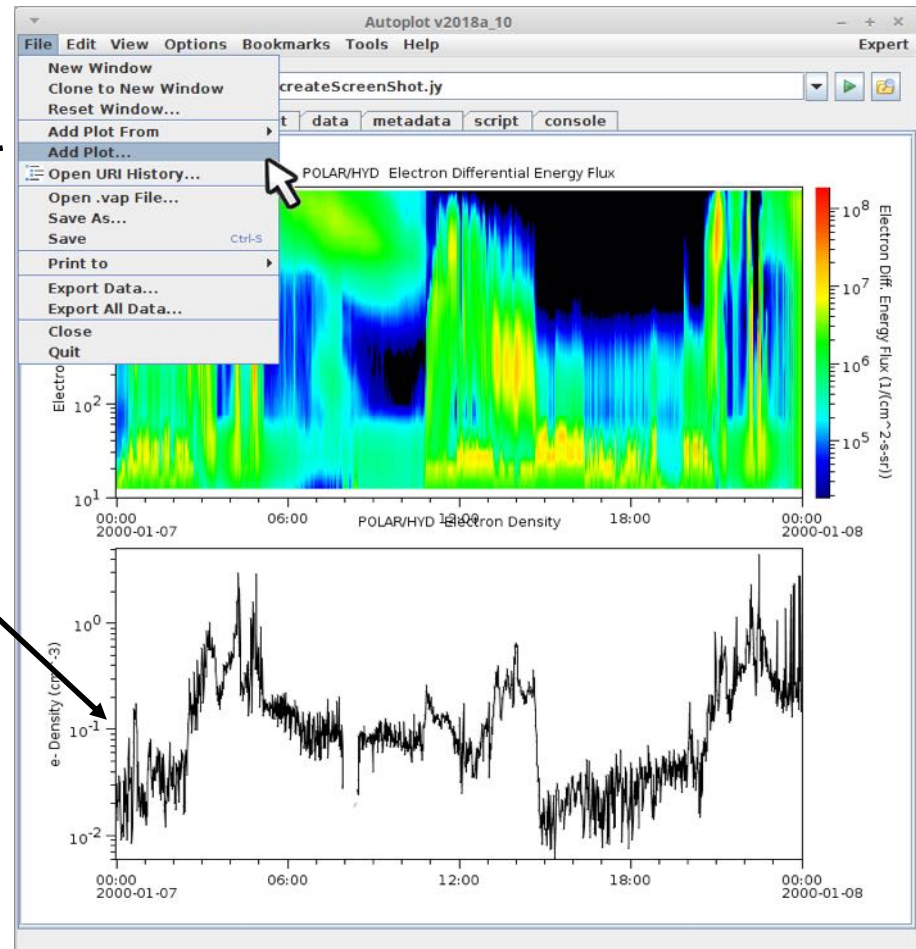
# More complex layouts

File→'Add Plot...' dialog is used to add additional plots.



Time axes are automatically bound together, but they can be disconnected.

Entire layout can be saved as a “.vap” file, an XML file describing the layout.



# File Aggregation

```
~/autoplot_data/fscache/https/cdaweb.sci.gsfc.nasa.gov/pub/data/omni/low_res_omni
spot5> ls
omni2_1965.dat  omni2_1968.dat  omni2_1971.dat  omni2_1974.dat  omni2_1977.dat
omni2_1966.dat  omni2_1969.dat  omni2_1972.dat  omni2_1975.dat  omni2_1978.dat
omni2_1967.dat  omni2_1970.dat  omni2_1973.dat  omni2_1976.dat  omni2_1979.dat
```

Autoplot works fine with just a pile of files. If there's a file on the web, you can read it or point a colleague to it. We can consume data from old web servers created in the 1990's, without ingesting it into a special server.

Often the data files form a long time series, and aggregation allows us to view the files as one data set.

.../omni2\_1972.dat is one file, but  
.../omni2\_1972.dat?timerange=1970-1980 is part of a time series.

Autoplot lists the remote directory, brings over the data it needs, then combines each granule into a long time series.

# Knowledge Containers

Autoplot URIs and .vap files are containers.

URIs are containers for a data set. (“Data Set” is a plottable thing in Autoplot)

They are compact strings which can be emailed, bookmarked, logged etc.

Autoplot .vaps are containers for plot configuration. (Stack of plots, which URIs go where)

# Sampling of Changes this past year

CDF

HAPI

GitHub Filesystem

Das2 (Graphics Library) changes

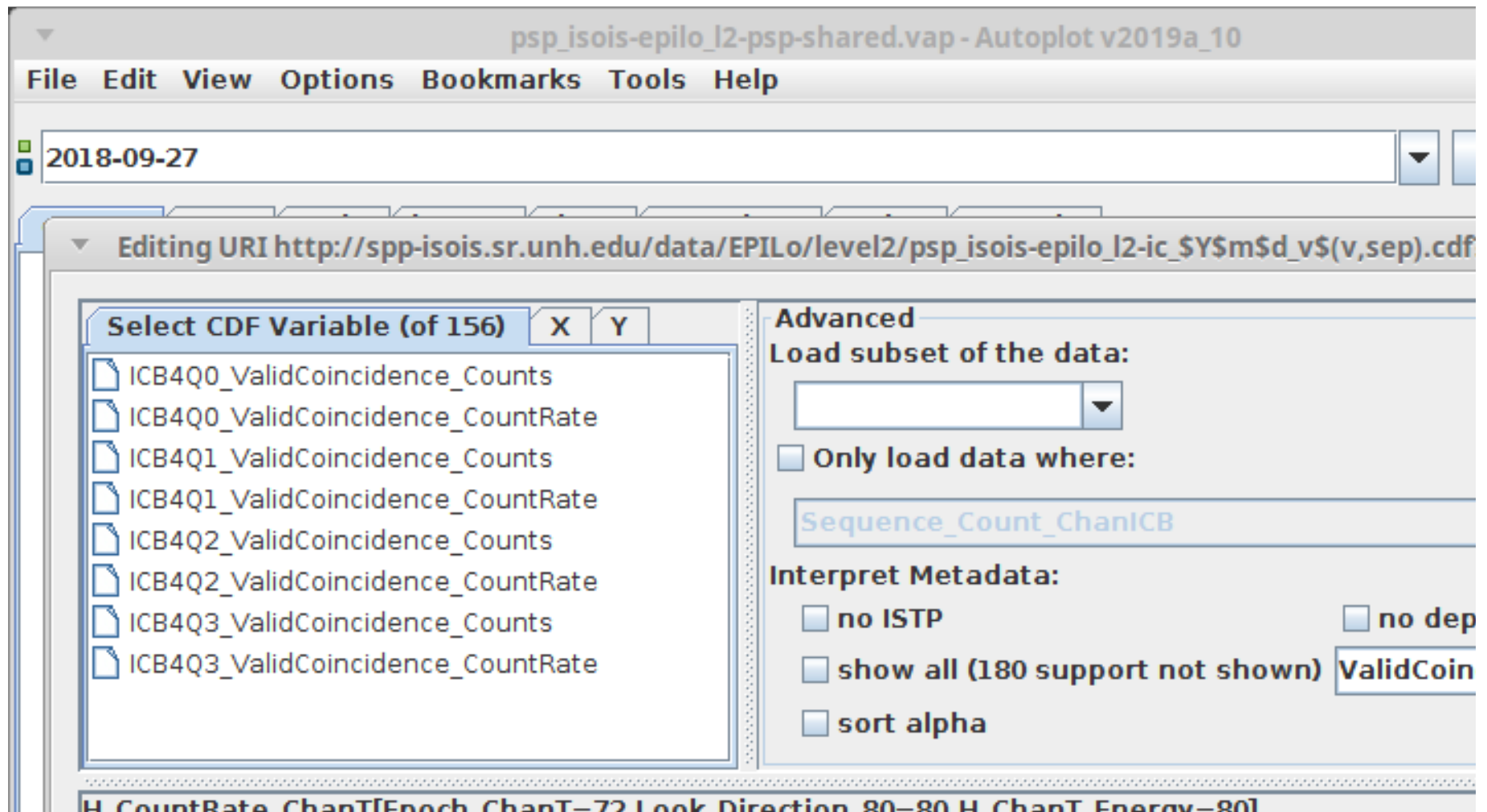
# New CDF Features

- Improve support for NetCDF files. Support should be as good as with CDF files. Gold mission.
- Many variable limitations. Solar Probe team has CDF files with ~1800 variables. Significant time (30s) to open prevented long time series plots.
- Discovered that aggregation's reduce wasn't reducing data with multiple dimensions (DEPEND\_2), preventing long time series plots.



# New CDF Features

- Filter on Parameters and metadata

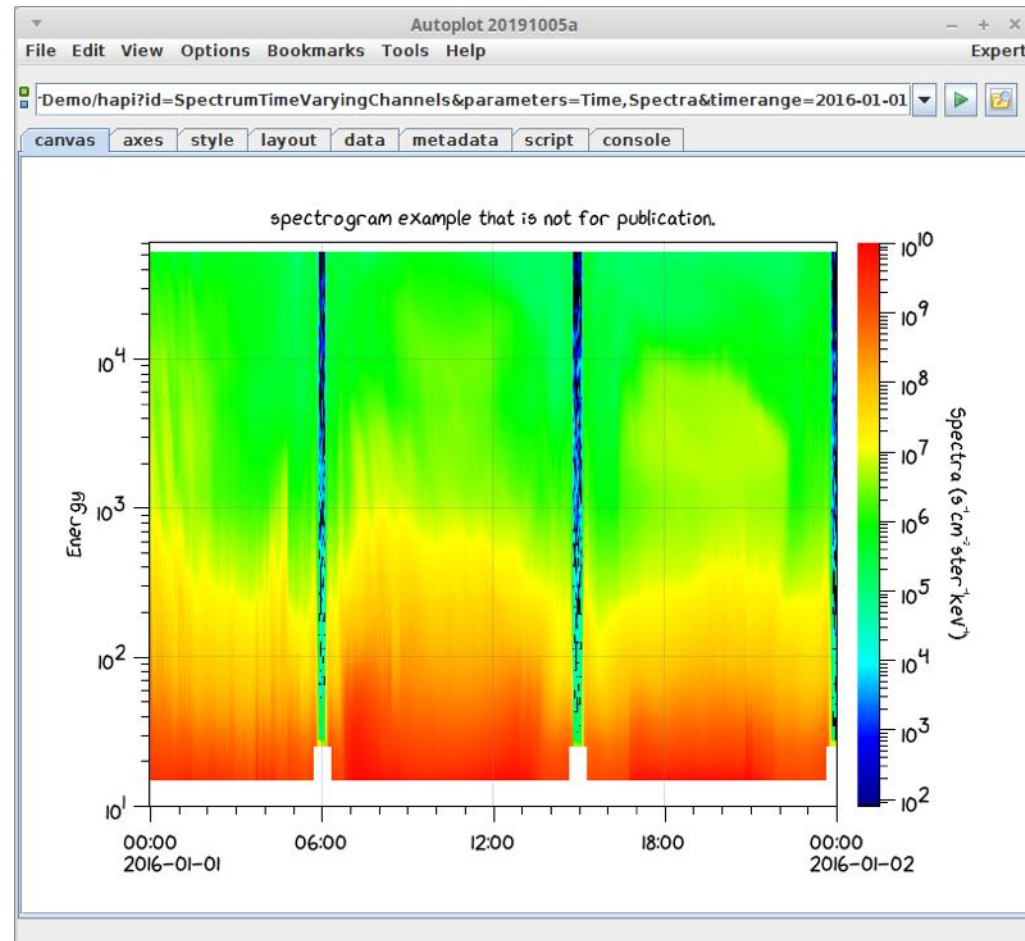


# New HAPI Features

- HAPI 2.1 Support
  - TimeStampLocation—timetags need not be centered in aliasing interval.
  - Tightened up responses
- Client-Side Caching
  - Binary responses are cached
  - Python client will can share cached data
  - All still experimental—caching is disabled by default

# New HAPI Features

- Prototype HAPI 3.0 Support
  - Time-Varying ytags
  - References in info response



# New GitHub Filesystem

- With SVN, Autoplot could grab vap files directly from the SVN server (with HTTP Filesystem type)
- GitHub is attractive because anyone can set up a repository (e.g. students and science teams)
- GitHub requires special support, because of strange filenames and time stamps.
- Support is actually for GitLabs, so teams can set up a GitLabs instance and use it with Autoplot.

# New Das2 Features

- (Das2 is the graphics and science analysis library Autoplot uses.)
- Das2streams are read directly into QDataSet (Autoplot's internal model), easing debugging and maybe helping performance.
- Chris Piker has been working on his Federated Catalog (<https://das2.org/browse/uiowa/juno/wav/survey>) for data products from Das2Servers (and HAPI servers and Piles of Files.)
- Refactoring of plotting codes to ease maintenance, performance
- manual tick locations matured ('+20s' means tick every 20 seconds, for example.) Contours renderer.
- Bugfixes to support older versions of Illustrator. Points far outside the plot bounds would not clip properly by Illustrator, so we do this manually now.
- Probably coming this year: proper units handling, "cm" / "s" -> "cm/s". HAPI 3.0 servers can identify units. (Cluster CDF units conventions are already supported, but never used in production codes.)



# Scripting Introduction

Last year at this meeting there seemed to be some interest in scripting.

Scripting in Autoplot started as I was given an IDL code to maintain, and with Jython I added enough commands that I could support the task in Autoplot.

Jython is Python implemented in Java. Python is a “glue” language which provides an easy way to access native C codes. Jython is Python, but provides access to Java codes instead of C codes.

The Java codes are Autoplot codes and Das2

# Scripting Introduction

Note that SciPy is trying to do the same thing, make an environment for science programming, so we've found that NumPy codes and Autoplot Jython codes are very similar.

We had a SciPy data digitizing script which was converted to an Autoplot script with very little code needing modification.

# Scripting Introduction

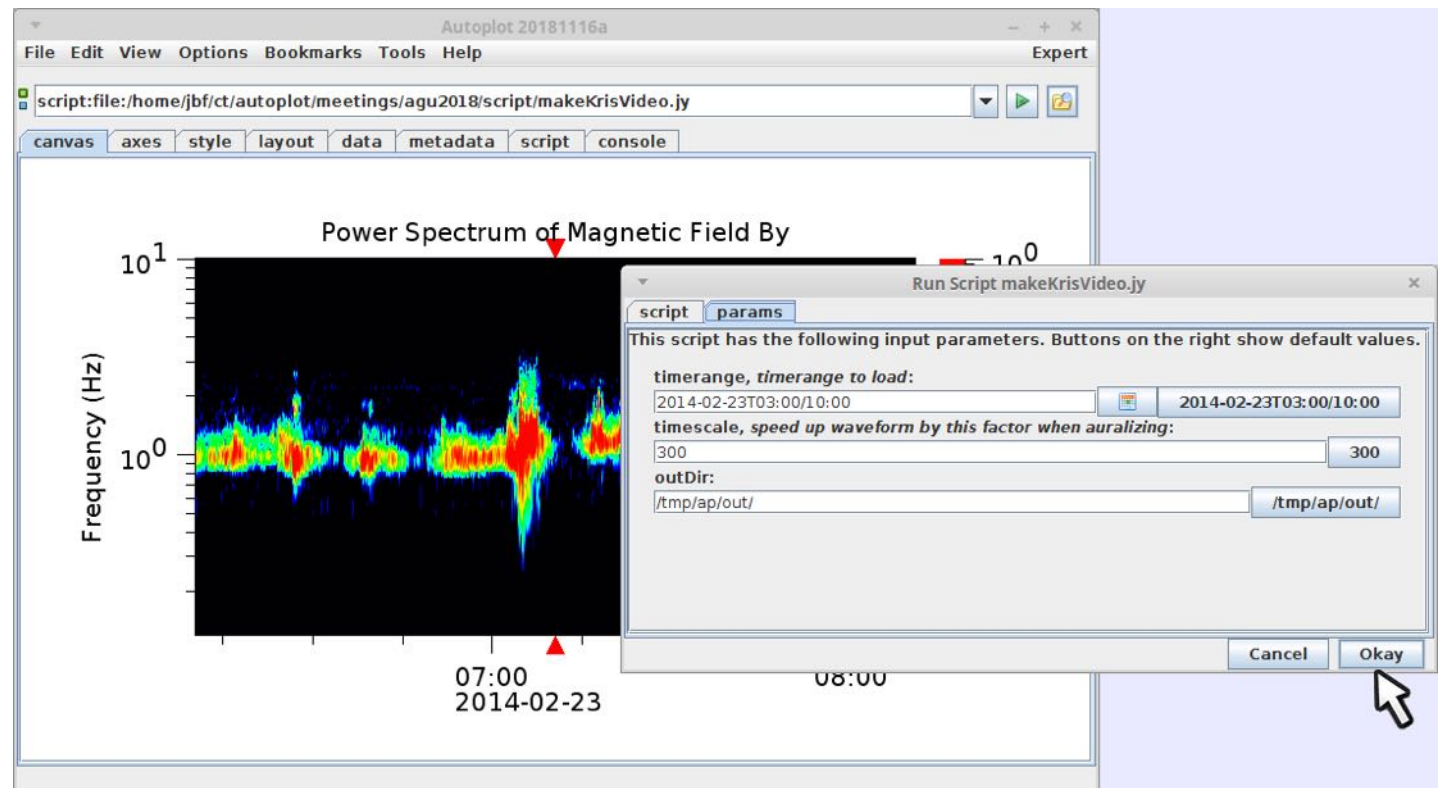
But the main goals and benefits:

- Provide an environment for science tasks and workflows
  - Using Autoplot URIs as a resource, provide means to combine data and create new products for analysis
  - Zero-installation: given one script, which can be on a web site, I can deliver functionality to colleagues. They run Autoplot, type in the script URI and the script runs. (Security warnings)
-

# Scripting Introduction

Note that like URIs and .vap files, scripts are another “knowledge container”

I can't tell you how a script I wrote last year works (where it gets its data, etc), but I know how to use it and I can go right in there and remind myself when necessary.



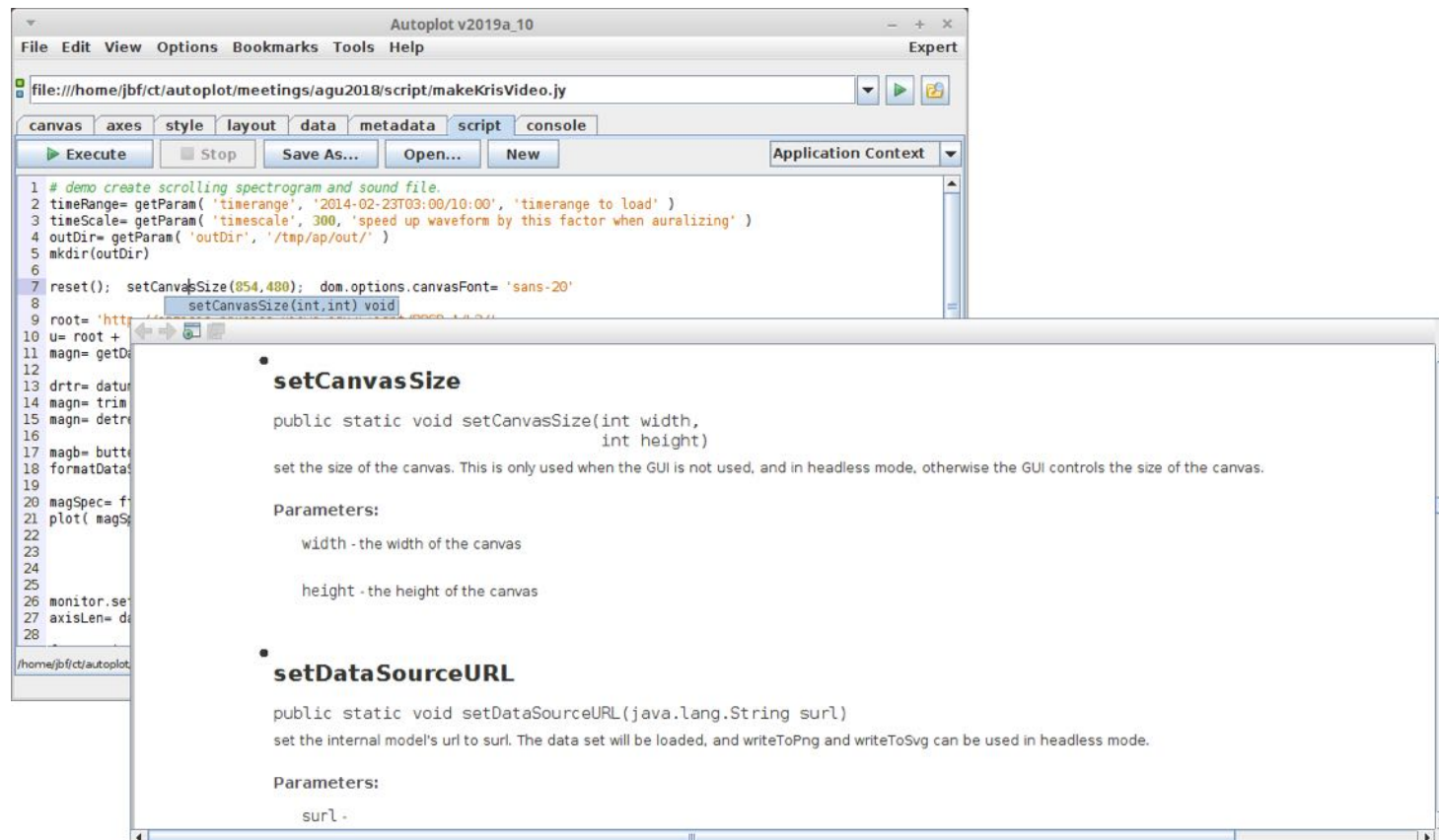
# Scripting Introduction

Documentation is still a mess. Internally, routines are well-documented using Java's documentation system. I will be rewriting pages that try to thoroughly document scripting.



# Scripting

The script editor links over to this documentation to provide a reference popup



# Scripting GitHub autoplot/dev

The github filesystem support is intended to enhance scripting.

In addition to easily distributing applications to colleagues, the dev area is intended to contain many small demonstrations and bug demos.

<https://github.com/autoplot/dev>

Autoplot can be pointed to github to run a script, and it will check for updates, keeping everybody updated and on the “same page.”

This will soon be tested automatically with a nightly test that runs through all codes and

# Scripting w/GitHub support

Note that it's actually GitLabs which is supported, which is freely available. Three GitLabs instances are used—GitHub and also U.Iowa's GitLabs, and U. Iowa Radio and Plasma Wave group GitLabs.

Let me know if you are interested in setting one up. Supposedly it's easy but I haven't succeeded. Also Autoplot doesn't automatically detect GitLabs, so I have to code it in.

Last, only public projects are accessible. Surely someone will want support for private (team) projects. and I hope to support this with

Chrome File Edit View History Bookmarks People Window Help

u/kris/2019/20190531/stereoBField20151222.jy

git.uiowa.edu/jbf/rbsp/blob/master/u/kris/2019/20190531/stereoBField20151222.jy

GitLab Projects Groups Activity Milestones Snippets Search or jump to...

R rbsp

Project

Repository

Files

Commits

Branches

Tags

Contributors

Graph

Compare

Charts

Issues 2

<< Collapse sidebar

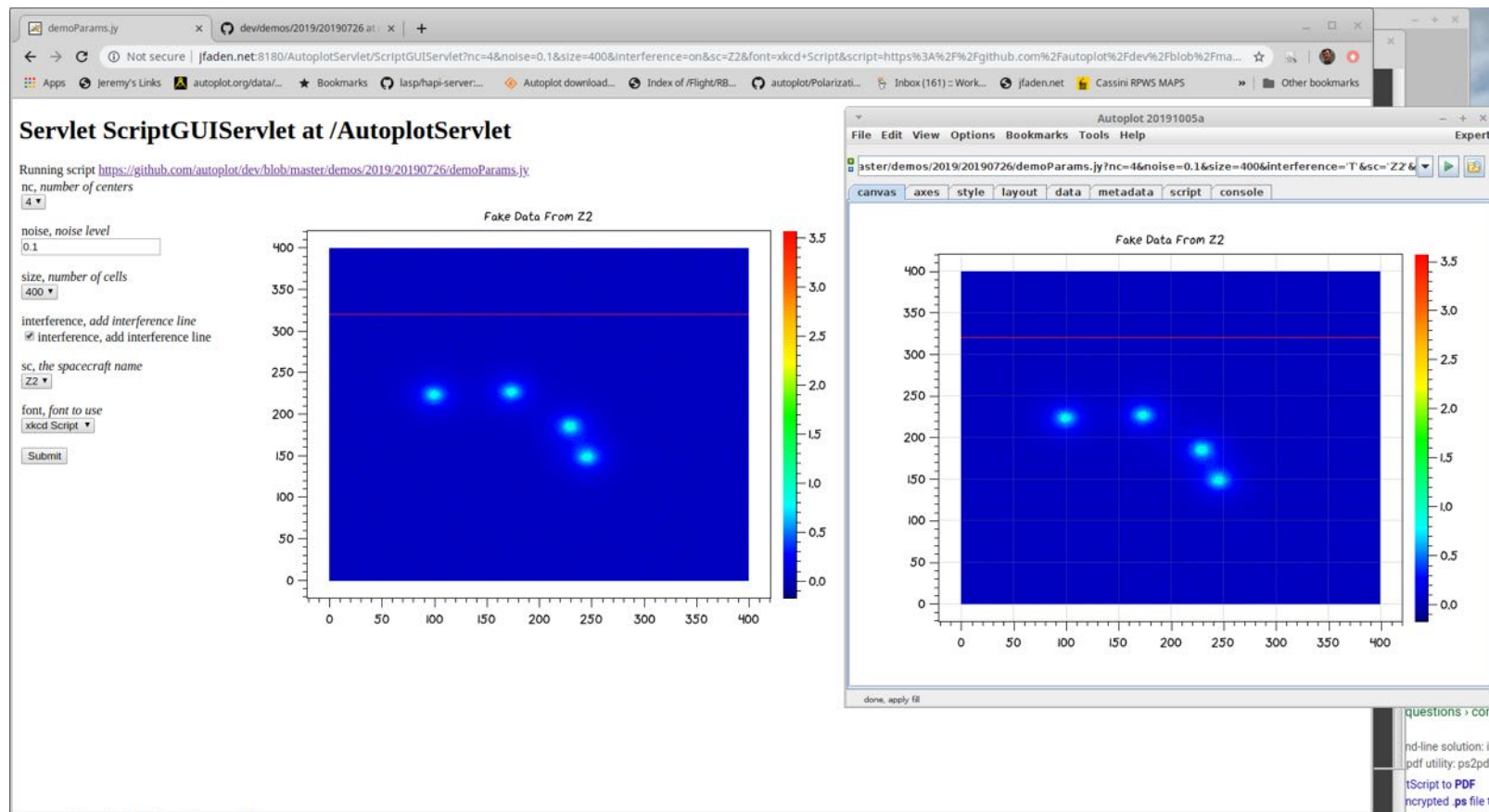
stereoBField20151222.jy 906 Bytes

```
1 maga= getDataSet('https://emfisis.physics.uiowa.edu/Flight/RBSP-A/L3/$Y/$m/$d/rbsp-a_magnetometer_hires-sm_emfisis-L3_$Y$m$d_v$
2 magb= getDataSet('https://emfisis.physics.uiowa.edu/Flight/RBSP-B/L3/$Y/$m/$d/rbsp-b_magnetometer_hires-sm_emfisis-L3_$Y$m$d_v$
3
4 maga= detrend( maga,101 )
5 magb= detrend( magb,101 )
6
7 maga= butterworth(maga,2,datum('5Hz'),True)
8 magb= butterworth(magb,2,datum('5Hz'),True)
9
10 dep0= maga.property(QDataSet.DEPEND_0)
11 r= where( within( dep0, '2015-12-22T06:20/7:40' ) )
12 maga= maga[r]
13
14 dep0= magb.property(QDataSet.DEPEND_0)
15 r= where( within( dep0, '2015-12-22T06:20/7:40' ) )
16 magb= magb[r]
17
18 stereo= bundle( maga, magb )
19 stereo.putProperty( QDataSet.DEPEND_0, maga.property( QDataSet.DEPEND_0 ) )
20
21 formatDataSet( stereo, '/tmp/stereo.20190604a.wav?timeScale=120' )
```

New Folder 2019-07-03 Zach's col...

# New Servlet Type—GitHub script

I've recently added a new servlet type—where a Jython script is used to build a GUI, and the result is shown beside.

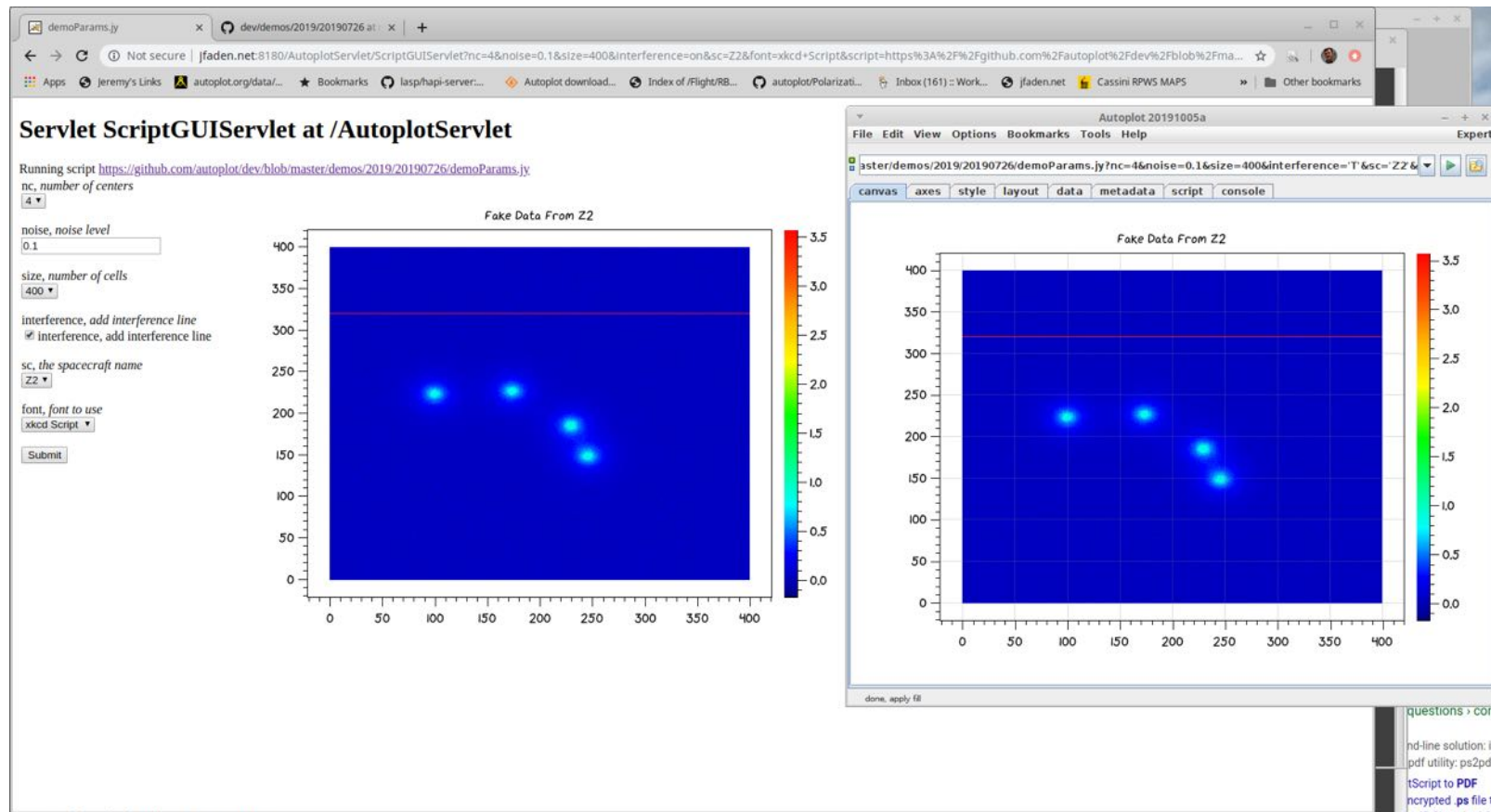




# New Servlet Type—GitHub script

Script runs on the Apache/Tomcat server, presumably on the server with the spacecraft data, so no data is transferred.

This allows development of servers without special knowledge.



(end)

# Matlab/IDL/Python interface

Autoplot's data access libraries can be used to read data into IDL and Matlab. The design goal is if you can see it in Autoplot, you should be able to read the digital data into these data environments.

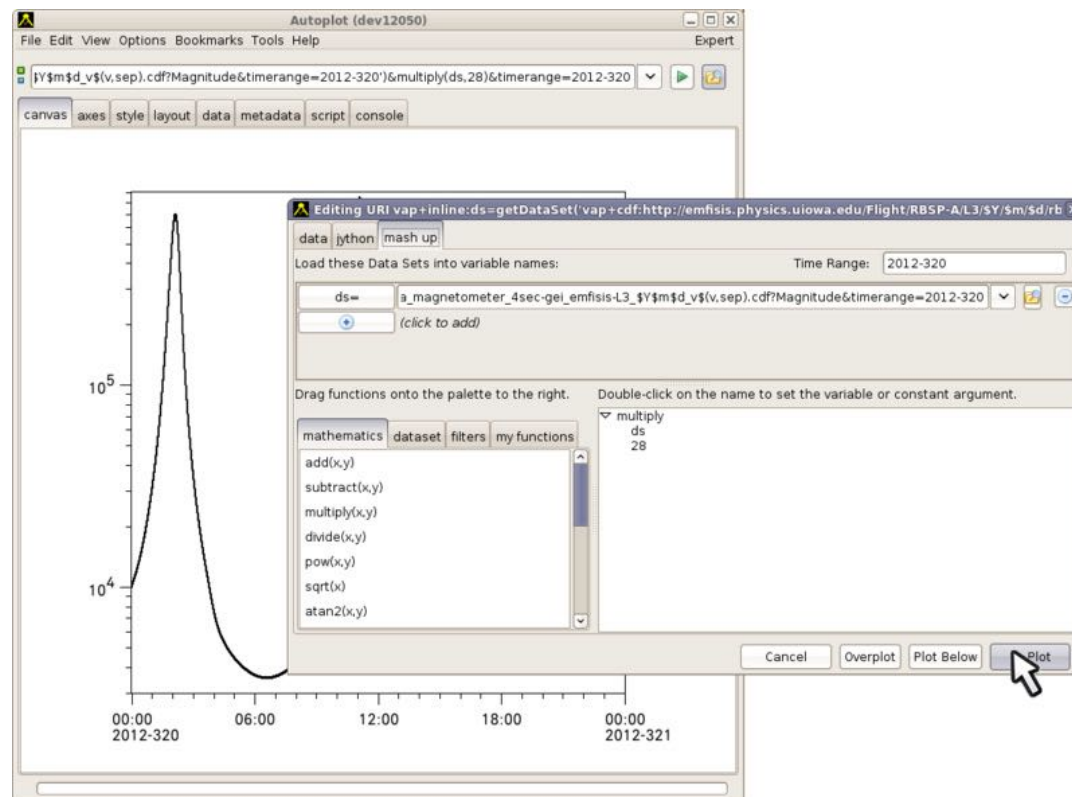
```
Unix% wget -N http://autoplot.org/jnlp/latest/autoplot.jar
IDL> setenv, 'CLASSPATH=./autoplot.jar'
IDL> qds= OBJ_NEW('IDLjavaObject$APDataSet', 'org.autoplot.idlsupport.APDataSet')
% QDataSetBridge v1.6.2
% Java version 1.7.0._25
IDL> qds.setDataSetUri, $
    'http://autoplot.org/data/swe-np.xls?column=data&depend0=dep0'
IDL> qds.doGetDataSet
IDL> print, qds.toString()
http://autoplot.org/data/swe-np.xls?column=data&depend0=dep0
data: data[dep0=287] (dimensionless)
dep0: dep0[287] (t1970) (DEPEND_0)
IDL> plot, qds.values('dep0'), qds.values()
IDL> qds.setPreferredUnits, 'hours since 2007-017T00:00'
IDL> plot, qds.values('dep0'), qds.values()
```

Use with Matlab is similar, just a different way to link in the software and syntactic differences. Use in Python is supported with Jpye. See <http://autoplot.org/python>, and <http://autoplot.org/matlab>.

# Data Mashup Tool

The Data Mashup Tool provides a way to load multiple data URIs, synchronize them to the same timetags, and then combine them in some way.

The mash-up is itself a URI, and can be bookmarked or sent to colleagues.



Here the scientist plots the plasma frequency by multiplying the B-field magnitude by 28.

# Summary

Autoplot has been an effective, general-purpose tool for looking at data

It's file-based support encourages use of standard file types like CDF, and patterns for open storage like a web site with files which can be aggregated.

It supports server types like HAPI and CDAWeb, encouraging use of existing standards.

It provides a free and capable analysis environment with scripting. Mash-up tool provides graphical scripting.