

The Radio Astronomer's IT Toolkit

Tara Murphy

ATNF Synthesis Imaging School
30th September 2008



Outline

- 1 An IT Toolkit
- 2 Scripting I
- 3 Scripting II
- 4 Version control
- 5 VO Tools
- 6 Summary





Scenario: How would you solve this problem?

Your supervisor gives you a data file that they've dug out of the archives. They say that it contains Nobel prize winning data... if only you could analyse it... you take a look...





Scenario: How would you solve this problem?

Your supervisor gives you a data file that they've dug out of the archives. They say that it contains Nobel prize winning data... if only you could analyse it... you take a look...

```

010.1002587 -78.0749976 3.6 4.7 J0000M32 0.000 D
322.2776209 -64.6831876 3.1 3.6 J0000M52 0.108 C
328.0067189 -68.5874347 2.1 2.5 J0000M48 0.193 D
325.6616556 -67.1618942 1.8 2.0 J0000M48 2.539 B
314.1262894 -55.5638273 9.0 2.8 J0000M60 1.266 A
314.6473335 -56.3390663 3.3 3.5 J0000M60 2.522 C
334.7202054 -71.7342040 1.8 2.3 J0000M40 5.693 B
321.3525638 -63.9139186 2.0 2.1 J0000M52 2.080 B
325.7259722 -67.2143813 2.9 3.0 J0000M48 3.311 D
347.0515173 -75.2447016 2.3 3.4 J0000M36 4.345 C
325.5104290 -67.0742957 2.9 3.1 J0000M48 0.105 C
001.9296920 -77.4361767 5.9 6.9 J0000M32 1.530 B
307.3447442 -41.6675454 1.8 2.1 J0000M76 6.400 B

```



... and its 10 000 lines long.



What do you do next?

- 1 Run, screaming, from the room, cursing the astronomers of yesteryear. Then brew a strong coffee and prepare yourself for several days of manual adjustments.





What do you do next?

- 1 Run, screaming, from the room, cursing the astronomers of yesteryear. Then brew a strong coffee and prepare yourself for several days of manual adjustments.
- 2 Sigh and take out some old FORTRAN code your supervisor gave you. Comment out 5 lines, uncomment 10 lines, make a few random tweaks, recompile and hope that works.





What do you do next?

- 1 Run, screaming, from the room, cursing the astronomers of yesteryear. Then brew a strong coffee and prepare yourself for several days of manual adjustments.
- 2 Sigh and take out some old FORTRAN code your supervisor gave you. Comment out 5 lines, uncomment 10 lines, make a few random tweaks, recompile and hope that works.
- 3 Spend a few hours remembering Perl and writing a script to reformat it, then sit basking in your own glory, wishing there were other people around to see how brilliant you are.





What do you do next?

- 1 Run, screaming, from the room, cursing the astronomers of yesteryear. Then brew a strong coffee and prepare yourself for several days of manual adjustments.
- 2 Sigh and take out some old FORTRAN code your supervisor gave you. Comment out 5 lines, uncomment 10 lines, make a few random tweaks, recompile and hope that works.
- 3 Spend a few hours remembering Perl and writing a script to reformat it, then sit basking in your own glory, wishing there were other people around to see how brilliant you are.
- 4 Write a one line Unix script in a couple of minutes, then move on to the Nobel prize winning research. You solve this kind of problem hundreds of times a day.





The solution/s

- Python solution

```
1 types = {'A':0, 'B':0, 'C':0, 'D':0}
2 for line in open('catalogue.txt'):
3     cols = line.split()
4     dec = float(cols[1])
5     type = cols[-1]
6     if dec > -70 and dec < -60:
7         types[type] += 1
8
9 for type in types:
10    print type, types[type]
```

- Shell solution

```
1 > awk '($2>-70 && $2<-60){print $7}' catalogue.txt | sort
   | uniq -c | sort
```



The radio astronomer's IT toolkit

- What IT skills do you need to do your research effectively?
 - A data reduction/processing package
(e.g. Miriad, AIPS++)
 - A FITS visualisation package
(e.g. kvis, ds9)
 - A range of Un*x tools
(e.g. cut, paste, grep, sed, awk, for loops)
 - A scripting language
(e.g. Python, Perl)
 - A plotting package
(e.g. IDL, matplotlib)
 - Familiarity with accessing large online resources
(e.g. NED, SIMBAD, VizieR, ADS)
 - Version control software
(e.g. Subversion)





The radio astronomer's IT toolkit

- Plus sometimes you need to
 - Write software in C, C++, FORTRAN
 - Read other people's code. . .
 - Query databases using SQL (e.g. SDSS, 6dF)
 - Use a wiki for collaboration
 - Set up a website
 - Set up a website with forms and cgi scripts
 - Set up a database to share/organise your data
 - Use VO Tools for complex queries



{girls, guys} like {girls, guys} who have skills





Why use scripting?

- Advantages vs. manual processing
 - Speed
 - Reproducibility
 - Documentation
 - Collaboration

- Advantages vs. 'real programming'
 - Speed of development
 - Flexibility
 - Easier for a beginner to understand



However...





Problem 1

You have reduced your data and now have 200 FITS files sitting on your computer. You want to measure the peak flux in each image.

```
1 % foreach i (`ls *.fits`)  
2 > echo $i  
3 > fits in=$i op=xyin out=$i.xy  
4 > maxfit in=$i.xy  
5 > end
```

* These examples use `tcs` but the principle is the same in other scripting flavours (e.g. `bash`)





Problem 1

```
1 1020-5803.fits.xy
2 Fits: version 1.1 09-Feb-07
3 There were no blanked pixels in the input
4 MAXFIT: version 29-Nov-95
5
6 Peak pixel   : (65,63,1) = 4.6736E-01
7
8 Fitted pixel : (65.21,63.11,1) = 4.5010E-01
9
10 ...
11
12 Coordinate:
13   Axis 1: Fitted RA---NCP   = 10:20:15.719
14   Axis 2: Fitted DEC--NCP  = -58:03:53.13
15   Axis 3:      FREQ-LSR    = 1.86239759E+01 GHz
16   ...
```





Problem 1

You want extract the peak flux and ignore all other output.

```
1 % foreach i (`ls *.fits`)
2 > echo $i
3 > fits in=$i op=xyin out=$i.xy
4 > maxfit in=$i.xy | grep Peak
5 > end
6 1020-5803.fits
7 Peak pixel : (65,63,1) = 4.6736E-01
8 1350-6135.fits
9 Peak pixel : (64,66,1) = 8.3779E-01
10 ...
```





Problem 1

You want to fix up the annoying file names (1020-5803.fits.xy).

```
1 % foreach i (`ls *.fits | sed 's/.fits//g'`)
2 > echo $i
3 > fits in=$i.fits op=xyin out=$i.xy
4 > maxfit in=$i.xy | grep Peak
5 > end
6 1020-5803
7 Peak pixel : (65,63,1) = 4.6736E-01
8 1350-6135
9 Peak pixel : (64,66,1) = 8.3779E-01
10 ...
```





Problem 1

You want to make a table of your results.

```
1 % foreach i (`ls *.fits | sed 's/\.fits//g'`)
2 > fits in=$i.fits op=xyin out=$i.xy
3 > echo -n "$i " >> table.txt
4 > maxfit in=$i.xy | grep Peak | cut "-d " -f8 >> table.txt
5 > end
6 % cat table.txt
7 1020-5803 4.6736E-01
8 1350-6135 8.3779E-01
9 ...
```



Everybody stand back, I know regular expressions!



<http://xkcd.com>





Problem 2

You want to measure the peak flux for each of your sources and print out an annotation file for kvis with circles proportional to the peak flux.

```
1 import os
2 for filename in os.listdir('data'):
3     # run maxfit on each image
4     (rastr, decstr, peak) = maxfit('data/%s' % filename)
5
6     # convert ra and dec to decimal form
7     ra = ra2decimal(rastr)
8     dec = dec2decimal(decstr)
9
10    # print a kvis annotation file
11    print 'circle %f %f %f' % (ra, dec, 0.02*peak)
```





Problem 2

A function to wrap maxfit to run within Python

```
1 def maxfit(filename):
2     cmd = 'maxfit in=%s' % filename
3     (rastr, decstr, cols) = (None, None, None)
4     for line in os.popen(cmd):
5         cols = line.split()
6         if line.startswith('Peak'):
7             peak = float(cols[-1])
8         elif 'RA' in line:
9             rastr = cols[-1]
10        elif 'DEC' in line:
11            decstr = cols[-1]
12    return (rastr, decstr, peak)
```





Problem 2

Some functions to do coordinate conversions.

```
1 def ra2decimal(rastr):
2     r = rastr.split(':')
3     ra = (float(r[0]) + float(r[1])/60.0 + float(r[2])/3600.0)*15
4     return ra
5
6
7 def dec2decimal(decstr):
8     d = decstr.split(':')
9     if d[0].startswith('-') or float(d[0]) < 0:
10        dec = float(d[0]) - float(d[1])/60.0 - float(d[2])/3600.0
11    else:
12        dec = float(d[0]) + float(d[1])/60.0 + float(d[2])/3600.0
13    return dec
```





Problem 2: putting it altogether

```
1 import os
2 def ra2decimal(rastr):
3     ...
4
5 def dec2decimal(decstr):
6     ...
7
8 def maxfit(filename):
9     ...
10
11 for filename in os.listdir():
12     (rastr, decstr, peak) = maxfit(filename)
13
14     ra = ra2decimal(rastr)
15     dec = dec2decimal(decstr)
16
17     print 'circle %f %f %f' % (ra, dec, 0.01*peak)
```





Problem 2: results

You want to measure the peak flux for each of your sources, calculate the mean for your sample, and print out an annotation file for `kvis` with circles proportional to the peak flux.

```
1 > python peakflux.py
2 circle 155.065496 -58.064758 0.009347
3 circle 207.676467 -61.586053 0.016756
4 circle 245.046142 -50.888425 0.079280
5 circle 250.000575 -48.861683 0.065074
6 circle 260.079975 -35.912969 0.120744
```





Data reduction scripts

- I have written some scripts to go with tutorial 1
 - ① `reduce_ngc253_v1.sh`
The most basic data reduction script.
 - ② `reduce_ngc253_v2.sh`
A slightly more advanced shell script.
 - ③ `reduce_ngc253.py`
Python script with more features.
- These are not the only way to do it — they are meant as a guide to get you started
- <http://www.physics.usyd.edu.au/~tara/synthesis>
- To run them, do
 - 1 `> chmod a+x reduce_ngc253_v1.sh`
 - 2 `> ./reduce_ngc253_v1.sh`
- They assume the raw data is in a directory called `data`





Why should I use version control?

- Works as a constant backup (accessible anywhere at anytime)
- Allows syncing between laptop, work and home desktops
- Allows collaboration on source code, papers, schedule files
- Allows students and supervisors to share code/resources
- Keeps a record of who made changes and why

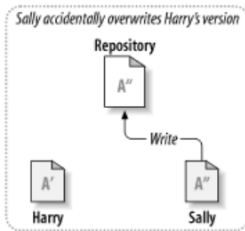
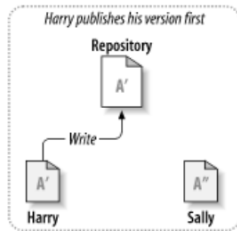
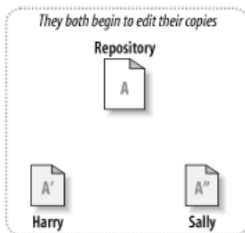
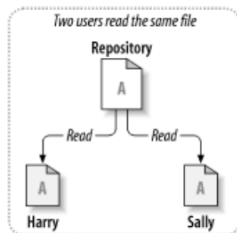
- Version control works like an e-version of a log book

- **It makes you a better coder!**





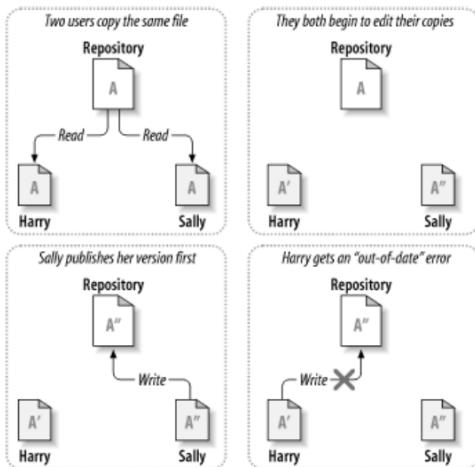
The problem to avoid



<http://svnbook.red-bean.com>



The Copy-Modify-Merge solution

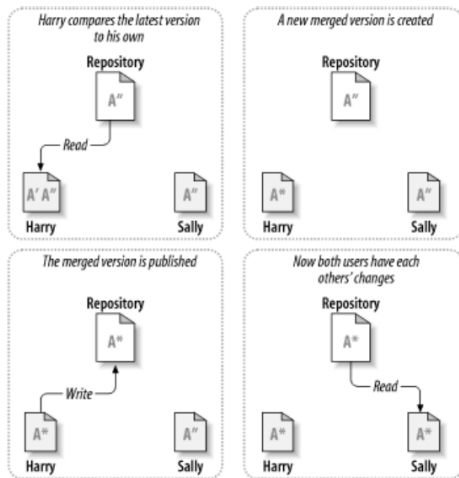


<http://svnbook.red-bean.com>





The Copy-Modify-Merge solution



<http://svnbook.red-bean.com>





What is the Virtual Observatory?

- We need a way of addressing the data management, analysis, distribution and interoperability challenges of modern astronomy
- The main drivers are
 - Data growth: volume and richness
 - Desire to work online
 - Multi-archive science
 - Large database science

The Virtual Observatory is a distributed collection of

- Data resources
- Software resources
- Computing (grid) resources
- Telescopes



What are VO tools?

NVO Open SkyQuery

Nodes

- Borist
- GALEX
- DLS
- RC3
- SDSS
- SDSSDR2
- TwoDF
- Twoqz
- USNOB
- GODDS
- HDFN
- HDFS
- UDF
- TWOMASS
- IRAS
- IRAZ
- ADFL
- FIRST
- NVSS
- NVORegistry

Query

```

SELECT o.objid, o.ra,
       o.dec, o.type, t.objid,
       t.l_m, o.z
FROM
  SDSSDR2:PhotoPrimary o, TWOMASS:PhotoPrimary t
WHERE xMATCH(o, t) < 2.5 AND
       RegionID('CIRCLE 32000 16.031 -0.891 30') AND
       (o.z - t.l_m) > 2
    
```

Spectrum Services

Spectrum Advanced Search Results

Found 447 objects. Displaying from 1 to 12

Object ID	Class	RA	DEC
1. SDSS J101549.00-002120.00	0271-51878-01	153 954180	0 338088
2. SDSS J101549.00-002020.00	0271-51878-01	153 954180	0 338088
3. SDSS J102043.82-000105.77	0271-51878-01	155 102580	0 010269
4. SDSS J102043.82-000105.77	0271-51878-01	155 102580	0 010269
5. SDSS J103432.73-002702.57	0273-51957-01	158 036330	-0 459713
6. SDSS J103432.73-002702.57	0273-51957-01	158 036330	-0 459713



An interactive sky atlas: Aladin

- Visualize digitized astronomical images
- Superimpose entries from catalogues or databases
- Interactively access online data from SIMBAD, NED, VizieR
- Fully VO aware — access other VO resources
- **See demo**
- <http://aladin.u-strasbg.fr>

- You can also write your own plug-ins (ask Keith Bannister)
- The developers are very keen to get feedback from users — they are happy to make suggested changes!



Querying online databases: SDSS

Sloan Digital Sky Survey / SkyServer

DR6 Tools

Getting Started
Famous places
Get Images
Scrolling sky
Visual Tools
Search
- Radial
- Rectangular
- SQL
- Imaging Query
- Spectro Query
Object Crossid
CasJobs

SQL Search

Please note: To be fair to other users, queries run from SkyServer search tools are restricted in how long they can run and how much output they return, by **timeouts** and **row limits**. Please see the **Query Limits help page**. To run a query that is not restricted by a timeout or number of rows returned, please use the **CasJobs batch query service**.

```
select top 10 objid,ra,dec,u,g,r,i,z
from PhotoObj
where
  u between 0 and 19.6
  and g between 0 and 20
```

Submit

To find out more

For an introduc page. In partic running much s section of the S

**select top 10 objid,ra,dec,u,g,r,i,z
from PhotoObj
where u between 0 and 19.6
and g between 0 and 20**

<http://cas.sdss.org/astrodr6/en/tools/search/sql.asp>



Querying online databases: Open Sky Query



NVO Open SkyQuery

Home Simple Query Advanced Query Import Tutorial Help

National Virtual Observatory

Nodes

- Rosat
- XMM
- GALEX
- GALEXR1
- DLS
- RC3
- GSC2
- NBCKDEDR1
- SDSS
- SDSSDR2
- SDSSDR3
- SDSSDR4
- TwoDf
- Twoqz
- TWOSLAQLRGDR
- TWOSLAQSOEDR
- USNOB
- GOODS
- HDFN
- HDFS
- UDF
- TWOMASS
- IRAS
- PSCz
- FIRST
- NVSS

Build **Edit** **Submit**

```

SELECT o.objId, o.ra,
       o.dec, o.r, o.type,
       t.objId, t.ra, t.dec
FROM
  SDSS:PhotoPrimary o, TWOMASS:PhotoPrimary t
WHERE XMATCH(o, t) < 3.5 AND
      Region('CIRCLE J2000 181.3 -0.76 6.5') AND
      o.type = 3
  
```

Welcome to the Open SkyQuery interactive query builder. You should see a parsed, clickable version of your entered query in the pane directly above this one.

If instead you see 'Query is empty', this means that builder needs a node or two to get started. You can add nodes to the builder by clicking the desired node's '+' icon in the left panel.

Once you have some sql in the above panel, you can then click on a token in that query to pull up a menu with options appropriate for that specific token. For example, one way to select an additional column from a mythical 'mytable' is to click on 'mytable' and then chose 'Add Selection', then pick the desired column from the given choices.

You can switch between 'edit' and 'build' modes at any time by using the tabs at the top of the query panel. Your changes from one will carry over to the other. Most menu options have additional mouse-over info.

Sample Queries

- XMatch/Region
- XMatch/Region 2
- Three Node Match
- Brown Dwarf Search
- MyData XMatch (upload)
- Xmatch t* (upload)
- ABELL Xmatch (upload)
- Single Node Query
- Single Node Join

<http://openskyquery.net/Sky/SkySite>



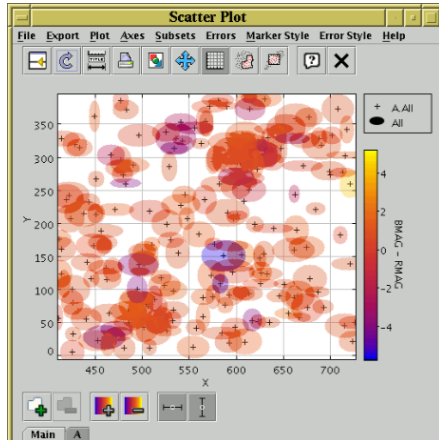
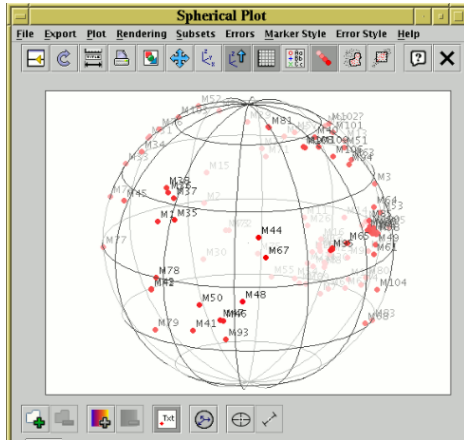


VO enabled plotting

- Many VO tools let you select sources and plot them
- All VO tools allow you to retrieve data as an XML VO table
- TOPCAT is an interactive graphical tool for analysis and manipulation of tabular data
- TOPCAT manifesto: *Does what you want with tables*
- <http://www.star.bris.ac.uk/~mbt/topcat>



VO enabled plotting: TOPCAT





Other tools worth looking at

- DataScope
heasarc.gsfc.nasa.gov/vo
- SkyView
<http://skyview.gsfc.nasa.gov>
- MyADS
<http://myads.harvard.edu>
- AstroGrid
<http://www2.astrogrid.org/science>
- Google Sky
<http://www.google.com/sky>





IT is critical in future astronomy

- IT is becoming increasingly important in 'everyday' science
- It is important to learn these skills now!

- Look out for the **Astroinformatics Summer School in 2009**
- Resources available from the Aus-VO Summer School websites
<http://www.physics.usyd.edu.au/ioa/ausvoss>
<http://www.physics.usyd.edu.au/ioa/ausvoss2007>

