



www.csiro.au

# Astroinformatics School 2009

## ATNF Spectral-line Analysis Package (ASAP)

Malte Marquarding  
Software Engineer – ASKAP/ASAP  
16 April 2009



# Introduction

- What is it's purpose?
  - process and analyse single-dish, single-pointing (radio) spectral line observations
  - handle multiple
    - spectral windows
    - beams
    - polarisations
  - full stokes/linear/circular polarisation support
- How is it developed?
  - python package on top of casacore (c++) using boost
  - ipython, numpy, matplotlib

# Introduction

- How do I run it?
  - a wrapper for ipython (interactive)  
asap

or

- in any python script  
**from asap import \***

# Development and support

- Wiki and tracking page
  - <http://svn.atnf.csiro.au/trac/asap>
  - view status
  - submit defects
  - submit enhancements
  - contribute
  - get the latest version
  - documentation

# ASAP objects

- fundamental entity is the **scantable**
  - asap representation of the data (disk or memory based)
  - contains a row for each spectrum
  - fundamental functions associated with it
    - **scantable.summary()**
  - math operators:  $sctable3 = (sctable0 + sctable1) / sctable2$
  - python iterable, e.g. if scans is a **scantable**  
for spectrum in scans:
  - export to FITS, CLASS (FITS), MS2, SDFITS, ASCII
  - access what's traditionally called header variables, through **scantable.get\_/set\_** functions
- **selector**
  - select out specific spectra, e.g.
    - **selector.set\_beams**([0,1])
    - **selector.set\_name**("Orion\*")

# ASAP object (continued)

- **asapplotter**
  - default instance plotter
  - plot any combination of scans/beams/ifs/polarisations across panels and/or stacked
- **fitter**
  - polynomial/gaussian fitting
  - can be extended to any non-linear fit
- **linecatalog**
  - representation of molecular line catalogues, e.g.
    - JPL
- **general functions**
  - operating on multiple **scantables**, e.g.
    - **merge**
    - **average\_time**
    - **quotient**

# ASAP object (continued)

- rc parameters (default values)
  - similar to matplotlib, e.g.
    - default value for frequency frame conversion
      - `scantable.freqframe: LSRK`
  - also honours matplotlib rc parameters

# Typical reduction process

- read data (scantable)
- build quotient (scantable.auto\_quotient)
- (flagging) (scantable.create\_mask/scantable.flag)
- baseline subtraction (scnatable.auto\_polybaseline)
- frame conversion/alignment (scantable.set\_freqframe)
- scaling (scantable.scale, scantable \*= value)
- averaging (scantable.average\_time/pols)
- plotting (plotter.plot)
- fitting (fitter.set\_scan, fitter.set\_function, fitter.fit)
- export (scantable.save, plotter.save)



# Help

- use `ipython help`, e.g.
  - `help(scantable)`
  - `scantable?`
  - `%pdoc`
- [wiki](#)
- [mailing list](#)
  - [asap-users@atnf.csiro.au](mailto:asap-users@atnf.csiro.au)

# MOPS broadband with line catalog overlay

