Single-Dish Tools and Tasks

Takeshi Nakazato, Kanako Sugimoto, Wataru Kawasaki, George Kosugi (NAOJ), and Takahiro Tsutsumi, Nick Elias (NRAO)
Outline

• Overview
  – Single-Dish Analysis in CASA
  – Organization

• Architecture
  – Overview
  – Tasks
  – Tools

• Recent Developments

• Future Plans
  – Short Term
  – Long Term
• Overview
  – Single-Dish Analysis in CASA
  – Organization
• Architecture
  – Overview
  – Tasks
  – Tools
• Recent Developments
• Future Plans
  – Short Term
  – Long Term
Single-Dish Analysis in CASA

• ASAP (ATNF Spectral Analysis Package)
  – originally, stand-alone software that depends only on casacore
  – depends on both casa and casacore in our version
  – Python interface
• Execute asap_init() to access SD functions
  – import ASAP as sd tool
    (all ASAP functions can be accessed as sd.*)
  – import SD tasks
• Scantable
  – Data format for SD data
  – CASA table
• Overview
  – Single-Dish Analysis in CASA
  – Organization

• Architecture
  – Overview
  – Tasks
  – Tools

• Recent Developments

• Future Plans
  – Short Term
  – Long Term
Architectural View of CASA

CLI (IPython)

Tasks

Tools

Python

GUI

Qt

ASAP Python (sd)

CASA C++ library

ASAP C++ library

CASACORE C++ library

depends on libatnf (data filler)

depends on tb tool and casalogger

user

depends on libatnf (data filler)

aatm (ATM library)

GUI matplotlib
SD Tasks

• `sdaverage`: calibration and averaging
• `sdbaseline`: baseline fitting
• `sdcal`: `sdaverage`+`sdbaseline`+`sdsmooth`
• `sdcoadd`: coadd multiple scantables into one
• `sdfit`: spectral line fitting
• `sdflag`: data flagging
• `sdimaging`: imaging spectral/total power data
• `sdimprocess`: noise reduction on SD OTF image (experimental)
• `sdlist`: display data summary
SD Tasks

- `sdmath`: simple arithmetics on SD data (cf. `immath`)
- `sdplot`: plot SD data
- `sdsave`: save data, data conversion
- `sdscale`: scaling spectral data (and Tsys if desired)
- `sdsmooth`: smoothing spectral data
- `sdstat`: compute statistics of spectral data
- `sdtpimaging`: imaging total power data, simple sky subtraction
- `sdsim`: SD simulation (cf. `simdata`)
## Data Reduction and SD Tasks

<table>
<thead>
<tr>
<th>Procedure</th>
<th>Task Name</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Flagging</td>
<td>sdflag</td>
<td></td>
</tr>
<tr>
<td>Calibration</td>
<td>sdaverage</td>
<td>sdcal</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Intensity scale</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Sky subtraction</td>
</tr>
<tr>
<td>Averaging</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Smoothing</td>
<td>sdsmooth</td>
<td></td>
</tr>
<tr>
<td>Baseline Fitting</td>
<td>sdbaseline</td>
<td>interactive mask</td>
</tr>
<tr>
<td>Line Fitting</td>
<td>sdfit</td>
<td>interactive mask</td>
</tr>
<tr>
<td>Imaging</td>
<td>sdtpimaging, sdimaging</td>
<td></td>
</tr>
<tr>
<td>Image Process</td>
<td>sdimprocess</td>
<td>Basket-Weaving (FFT) or Press</td>
</tr>
<tr>
<td>Statistics</td>
<td>sdstat</td>
<td>interactive mask</td>
</tr>
<tr>
<td>Other Utility Tasks</td>
<td>sdcoadd, sdlst, sdmath, sdplot, sdsave, sdscale, sdsim</td>
<td></td>
</tr>
</tbody>
</table>
SD Tools

• Originally, user interface of ASAP
  – ASAP functions are imported as sd tool
  – all ASAP functions are accessible just putting sd. at the head (e.g. sd.scantable to access scantable class)

• Tool classes are written in Python
  – in contrast to CASA tools (*_cmpt.cc/h)

• Each C++ class have Python interface
  – asap/src/python_*_.cpp
  – Boost.Python

• C++ classes are packed one module (_asap)
  – You have direct access to several C++ classes (sd._asap.xxx)
SD Tools

C++

- Scantable
  - Data structure
  - MAIN table
  - CASA Table
  - Header
  - Keyword
  - Subtable (CASA table)

- STSelector
  - Data selector
  - IF
  - Scan No.
  - Polarization
  - Beam
  - etc.

- STFiller
  - Data filler (reader)
  - Scantable
  - SDFITS
  - MS (ASDM)
  - NRO data

- STWriter
  - Data writer
  - Scantable
  - SDFITS
  - MS

Python (sd._asap.xxx)

- Log
- Scantable
- selector
- stfiller

Python (sd.xxx)

- asaplog
- scantable
- selector
- reader

CASA Logger

Boost.Python

2010/5/10-14 CASA Developers Meeting 12
SD Tools

Python (sd._asap.xxx)

Python (sd.xxx)

C++

STMath
Math utilities
• Calibration
• Averaging
• etc.

STLineFinder
Line finder
• Automatic Spectral line detection

STFitter
Fitting functions
• Baseline fit (polynomial)
• Line fit (Gaussian)

STFitEntry
Fit parameters
• Function
• Parameters
• Components
• Frame info

LineCatalog
Spectral Line catalog

STLineFinder
Line finder
• Automatic Spectral line detection

linecatalog
linecatalog

fitentry
fitentry

asapfit
asapfit

fitter
fitter

asapmath
asapmath

Math utilities

Math utilities

asaplinefind
asaplinefind

Line finder

Line finder

Boo st. P y t h o n
SD Tools

GUI classes (Python)

Third Party

- matplotlib
- Tk

ASAP (sd.xxx)

- asaplotbase
- asaplot
- asaplotgui
- asapplotter
- interactivemask
- casatoolbar (after 3.0.2)
Data Import and Export

• Import
  – Scantable
  – ASDM (via MS)
  – MS
  – SDFITS (limited use for GBT)
  – RPFITS
  – NRO FITS and NRO OTF

• Export
  – Scantable
  – MS
  – SDFITS
• Overview
  – Single-Dish Analysis in CASA
  – Organization
• Architecture
  – Overview
  – Tasks
  – Tools
• Recent Developments
• Future Plans
  – Short Term
  – Long Term
Developments: 3.0

- ASAP upgraded to 2.3.1
- Use CASA Logger
- new task: imaging spectral data (sdimaging)
- new task: image processing (sdimprocess)
- new task: SD simulation (sdsim)
- sdaverage improvement
  - averaging OTF data
  - new calibration scheme (Chopper-Wheel, freq. switch)
- row-based flagging in sdflag
- verification parameter in some sd tasks
- interactive masking extended to sdfit and sdsstat
- data selection by row number in sdsave
- fortran style formatting parameter in sdsstat
Developments: 3.0.1

• calibration scheme for ALMA (position switch) implemented in sdaverage
• y-axis clipping implemented in sdflag (similar to clipminmax for flagdata)
• Lorentzian fitting implemented in sdfit
• vector scaling factor available in sdscale
• sophisticated arithmetic in sdmath (similar to immath)
• parameter modification in sdsim (to align with simdata task)
• some improvements and bug fixes in sdsim
• help text for SD part revised
Developments: 3.0.2

• new task: compute moments from SD spectral data in MS format (msmoments)
• plotter improvements
  – more header informations printed on the panel
  – plotter control parameters added in sdplot
• averaging weight parameter in sdcoadd
• Overview
  – Single-Dish Analysis in CASA
  – Organization
• Architecture
  – Overview
  – Tasks
  – Tools
• Recent Developments
• Future Plans
  – Short Term
  – Long Term
Short Term Plan

• JIRA (CASA): 36 unresolved tickets (after 3.0.2)
  – data filler: 3 tickets
  – imaging: 5 tickets
  – flagging: 6 tickets
  – baseline fitting: 4 tickets
  – spectral analysis: 4 tickets
  – plotter/GUI: 10 tickets
  – regression: 4 tickets

• Upcoming JIRA tickets

• Other
  – Regression using ALMA SD data
  – Merge sdimaging and sdtpimaging tasks
  – Merge sdsim into simdata
  – Documentation
Long Term Plan

• asap_init()
  – Is it needed?

• Data Format
  – Use scantable? or switch to MS?
  – Will have discussion session on May 13

• Speed up
  – Refactoring existing codes (Python and C++)
  – Rewrite Python layer (sd tool) in C++?

• PIPELINE SD Heuristics functions
  – Rewrite unavailable functions (Python codes) in C++ and import