Madagascar (formerly known as RSF)

Open-source software package for geophysical data processing and reproducible research

Gilles Hennenfent
PhD student
Department of Earth & Ocean Sciences
The University of British Columbia

Email: ghennenfent@eos.ubc.ca
Homepage: http://wigner.eos.ubc.ca/~hegilles

Madagascar*

- motivation
  - bringing reproducibility and peer review to the field of computational geophysics
  - providing an efficient technology transfer tool

- 2-level architecture
  - low-level (main programs)
    - typically developed in C/C++ (fortran) programming language
  - high-level (processing flows and documentation)
    - typically developed in Python (processing flows) and LaTeX (documentation)
    - machinery provided by SCons

* from http://rsf.sourceforge.net

Main features

- new package started in 2003 by Dr. S. Fomel (BEG - University of Texas at Austin)
  - >300 main programs
  - >3000 tests, reproductible reports and papers
  - capitalizes on previous experience in geophysical open packages (SEPlib and SU)
- released in June 2006 during EAGE workshop
  - available for download on ***http://rsf.sourceforge.net***
- distributed under the standard GPL open-source license
- uses simple, flexible, and universal data format
- follows modern software engineering practices
  - module encapsulation
  - test-driven development
- managed by developers from different research groups around the world (including SLIM @ UBC!)

Python

- what is Python?
  - dynamic object-oriented programming language

- main features
  - multi-paradigm language
    - object orientation, structured programming, functional programming, and aspect-oriented programming supported
  - uses automatic memory management
    - garbage collection
  - very clear, readable syntax
  - strong support for integration with other languages and tools
    - extension modules can be written in C/C++ and wrapped with SWIG
  - comes with extensive standard libraries
  - cross-platform
    - Windows, Linux/Unix, Mac OS X, and OS/2
  - distributed under an OSI-approved open source license

* from http://www.python.org/

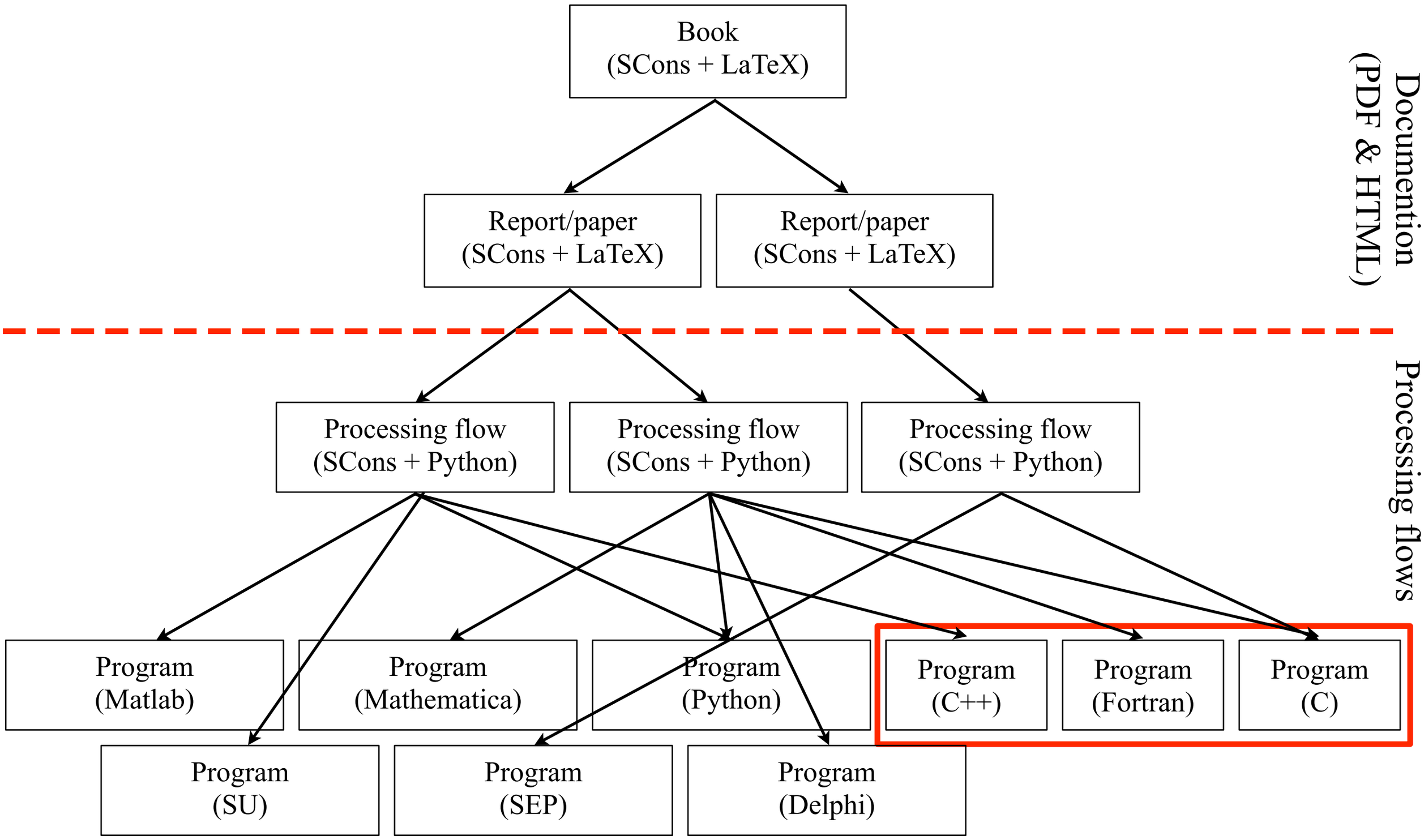
SCons*

- what is SCons?
  - next-generation, cross-platform, **build tool**
  - alternative to Make
- main features
  - configuration files are Python scripts
  - reliable, automatic dependency analysis
    - no need of *make depend* step
  - built-in support for multiple languages
    - C, C++, Java, Fortran, Qt, SWIG, (La)TeX among others...
  - cross-platform
    - Linux, POSIX systems, Windows (NT, 2000, XP), Mac OS X, and OS/2
  - reliable detection of file changes using MD5 signatures (and/or time stamp)
  - global view of dependencies
  - parallel builds
  - released under MIT license

* from http://scons.tigris.org

Command line

3 levels of usage



Madagascar file format

- borrowed from the SEPlib data format
  - data values stored in a binary format
    - contiguous 1D array
    - conceptual data model is a multidimensional hypercube
  - data attributes stored in text files that can be read by humans and processed with universal text-processing utilities

```
xom.rsf:
    in="/tmp/xom.rsf@"
    esize=4 type=float form=native
    n1=100          d1=.004         o1=0
    n2=10           d2=5            o2=0
        1000 elements 4000 bytes
```

  - pack header and data together (optional)

- compatible with other file formats
  - SEPlib (directly)
  - SU, SEGY (using conversion tools)

Madagascar functions

- *"sf"* prefix
  - e.g. sfin, sffft1, sffdct

- file-in, file-out, command line arguments
  - sfthr <in.rsf >out.rsf fthr=file1.rsf thr=4 mode=soft
  - sfmath x=file1.rsf y=file2.rsf power=file3.rsf output='sin((x+2*y)^power)' >out.rsf

- self-documention
  - list of all functions
    - http://egl.beg.utexas.edu/RSF/
    - included in *doc* folder of Madagascar software
  - function doc. obtained by typing its name on the command line

```
NAME
      sfthr
SYNOPSIS
      sfthr < in.rsf > out.rsf fthr=fthr.rsf thr= fthr= mode=
COMMENTS
      Threshold float/complex inputs given a constant/varying
      threshold level.

      Methods available:
      - soft
      - hard
      - non-negative Garrote (nng)

      Written by: Gilles Hennenfent & Colin Russell, UBC
      Created: February 2006

PARAMETERS
      string  fthr=   varying threshold level (>0)
      string  mode=   'soft', 'hard', 'nng' (default: soft)
      float   thr=    threshold level (>0)
USED IN
      slim/rsf/sfthr
SOURCE
      user/slim/thr.c
```

Demo

```
sfspike n1=1000 k1=300 | sfbandpass fhi=2 phase=1 >junk.rsf
sfin junk.rsf
sfattr <junk.rsf
sfrm junk.rsf
sfspike n1=1000 k1=300 | sfbandpass fhi=2 phase=1 | sfwiggle | tube
sfmath n1=300 n2=1 output="cos(2*3.141*x1/150)" | sfnoise seed=1 var=0.1 | sfgraph | tube
```

Write your own Madagascar funtions

- C/C++

```c
#include <rsf.h>

int main(int argc, char* argv[])
 {
    int n1, n2, i1, i2;
    float clip, *trace;
    sf_file in, out; /* Input and output files */

    /* Initialize RSF */
     sf_init(argc,argv);
    /* standard input */
    in = sf_input("in");
    /* standard output */
    out = sf_output("out");
    /* check that the input is float */
    if (SF_FLOAT != sf_gettype(in))
            sf_error("Need float input");
    /* n1 is the fastest dimension (trace length) */
    if (!sf_histint(in,"n1",&n1))
            sf_error("No n1= in input");
    /* leftsize gets n2*n3*n4*... (the number of traces) */
     n2 = sf_leftsize(in,1);
    /* parameter from the command line (i.e. clip=1.5 ) */
    if (!sf_getfloat("clip",&clip)) sf_error("Need clip=");
    /* allocate floating point array */
     trace = sf_floatalloc (n1);
    /* loop over traces */
    for (i2=0; i2 < n2; i2++) {
            /*read a trace */
            sf_floatread(trace,n1,in);
            /* loop over samples */
            for (i1=0; i1 < n1; i1++) {
                if      (trace[i1] >  clip) trace[i1]= clip;
                else if (trace[i1] < -clip) trace[i1]=-clip;
            }
            /* write a trace */
            sf_floatwrite(trace,n1,out);
     }
     exit(0);
}
```

Write your own Madagascar funtions

- Matlab

```
function clip(in,out,clip)
dims = rsf_dim(in);
n1 = dims(1);            % trace length
n2 = prod(dims(2:end)); % number of traces
trace = 1:n1;            % allocate trace
rsf_create(out,in)       % create an output file
for i2 = 1:n2            % loop over traces
    rsf_read(trace,in,'same');
    trace(trace >   clip) =  clip;
    trace(trace < - clip) = -clip;
    rsf_write(trace,out,'same');
end
```

- Python

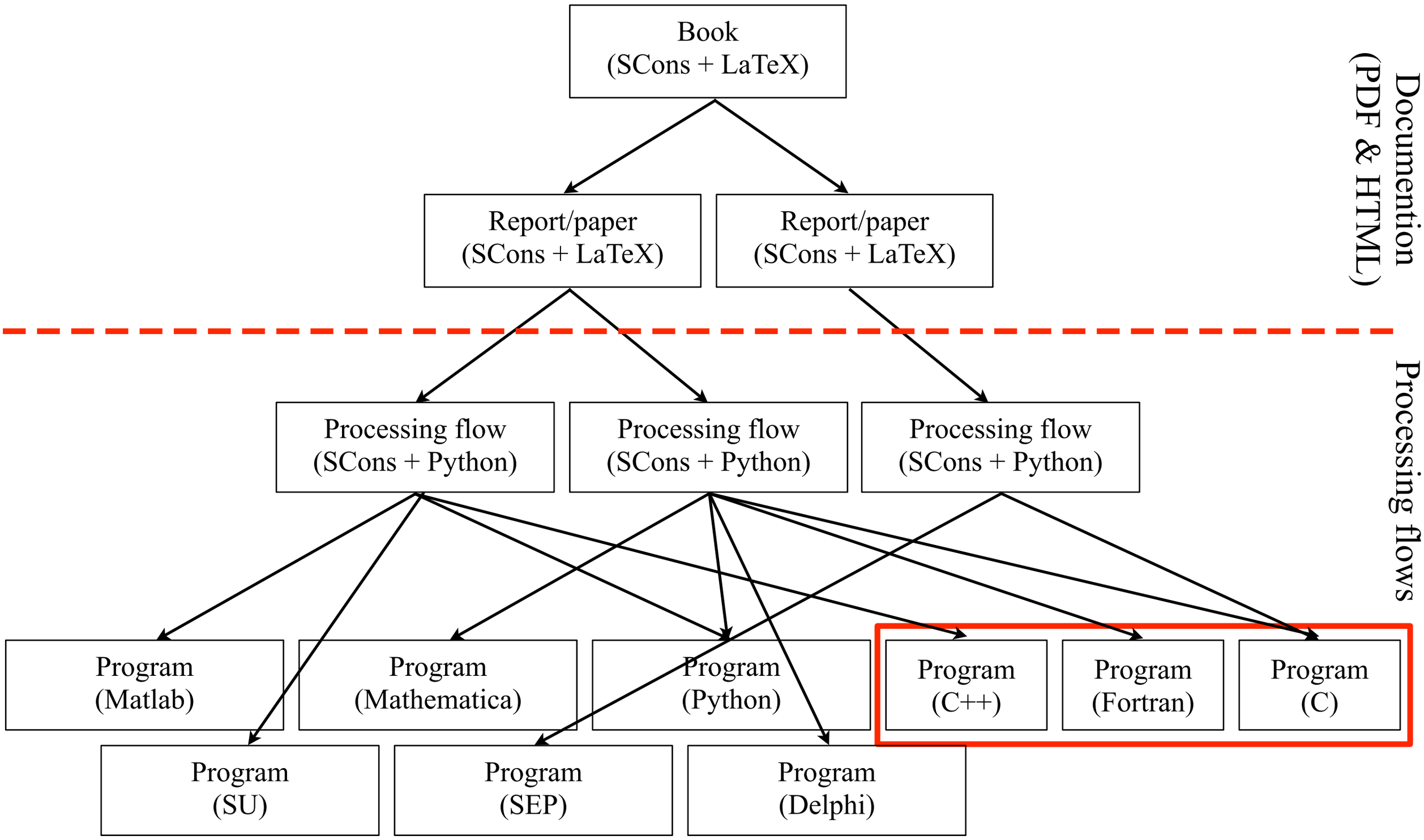
```
#!/usr/bin/env python

import numpy
import rsf
par = rsf.Par()
input  = rsf.Input()
output = rsf.Output()
assert 'float' == input.type
n1 = input.int("n1")
 n2 = input.size(1)
assert n1
clip = par.float("clip")
assert clip
trace = numpy.zeros(n1,   'f')
for i2 in xrange(n2): # loop over traces
    input.read(trace)
    trace = numpy.clip(trace,-clip,clip)
    output.write(trace)
```

Write your own Madagascar funtions

- fortran-77
- fortran-90

Command line

3 levels of usage



Processing flows

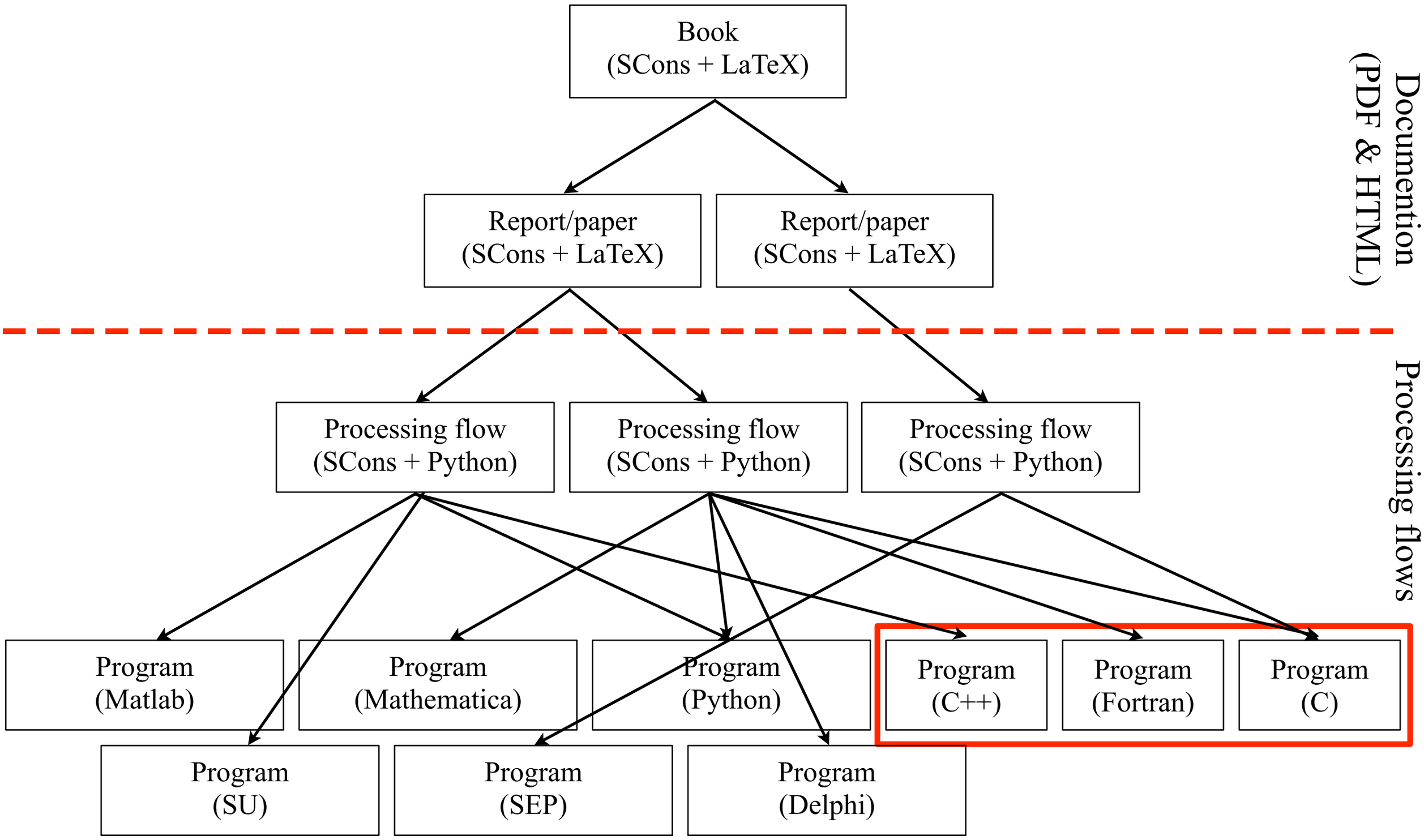
- SCons and Madagascar
  - 4 main SCons commands
    - Fetch(filename,directory,ftp_server)
    - Flow(outputs,inputs,commands)
    - Plot(graph_name,input,plot_command)
    - Result(graph_name,inputs,command)
  - processing flow is written in a file named *SConstruct*
  - default SCons targets
    - scons, scons lock
    - scons file.rsf, scons graph.vpl
    - scons graph.view
  - CAUTION: Python (and thus SCons) use indentation, rather than curly braces, to delimit blocks
    - an increase in indentation comes after certain statements
    - a decrease in indentation signifies the end of the current block
- SCons, Madagascar, and Python

Demo

- scons basics
- processing flows
  - SCons and Madagascar
  - SCons, Madagascar, and Python
  - SCons, SU, and Python

Command line

3 levels of usage



Documentation

- SCons, Madagascar, (Python) and LaTeX
  - steps
    - run processing flows involved in the report/paper (scons)
    - lock results (scons lock)
    - write report/paper
    - compile report/paper (scons pdf)
    - read paper (scons read) or use your favorite PDF reader...
  - LaTeX style, included package, etc. defined in a *SConstruct* file

- Webpages
  - scons html

Demo

- write a report/paper with figures linked to code used to generate them
- make webpages out of report/paper

Useful Madagascar links

- main page
  - http://rsf.sourceforge.net
- installation instructions
  - http://egl.beg.utexas.edu/RSF/book/rsf/rsf/install_html/
- programming interfaces
  - http://egl.beg.utexas.edu/RSF/book/rsf/rsf/api_html/