Madagascar Course in TRIP

Yujin Liu

TRIP, CAAM, Rice

09/01/2013

Agenda

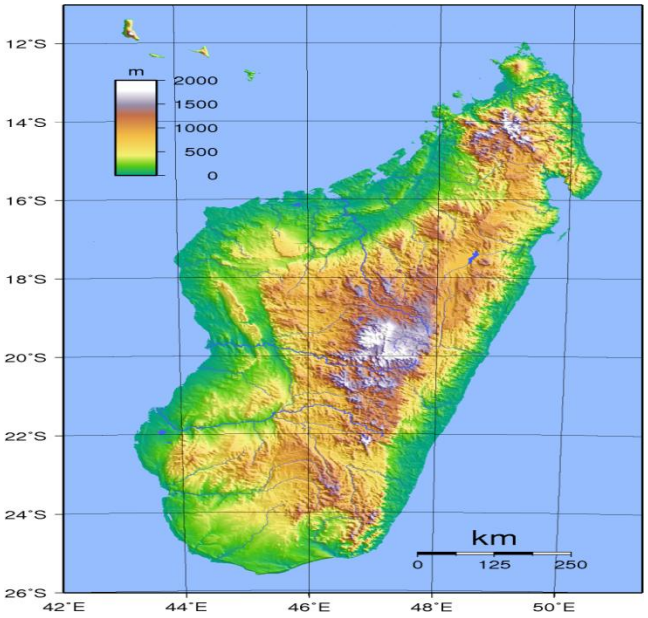
1. Introduction
2. Basic knowledge
3. Example
4. Discussion and Conclusion

Agenda

1. Introduction
2. Basic knowledge
3. Example
4. Discussion and Conclusion

What's Madagascar here?

- It's not an island.



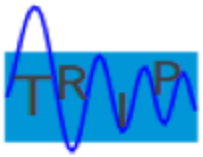
What's Madagascar here?

- It's not an island.
- It's not an animation.

What's Madagascar here?

RICE
Unconventional Wisdom

- ☐ It's not an island.
- ☐ It's not an animation.
- ✓ It's an open-source software package for multidimensional data analysis and reproducible computational experiments



What's Madagascar here?

- ☐ **It's not an island.**
- ☐ **It's not an animation.**
- ✓ **It's an open-source software package for multidimensional data analysis and reproducible computational experiments**

"It is a big chore for one researcher to reproduce the analysis and computational results of another […] I discovered that this problem has a simple technological solution: illustrations (figures) in a technical document are made by programs and command scripts that along with required data should be linked to the document itself […] This is hardly any extra work for the author, but it makes the document much more valuable to readers who possess the document in electronic form because they are able to track down the computations that lead to the illustrations." **(Claerbout, 1991)**

What's Madagascar here?

RICE
Unconventional Wisdom

- ☐ It's not an island.
- ☐ It's not an animation.
- ✓ It's an open-source software package for multidimensional data analysis and reproducible computational experiments

◆ **Developers:**

S. Fomel, P. Sava, T. Alkhalifah, Y. Liu, K. Schleicher, W. Symes, J. Shragge, J. Dellinger, G. Hennenfent, J.Rickett, W. Burnett et. al

TRIP

What's Madagascar here?

RICE
Unconventional Wisdom

- ☐ It's not an island.
- ☐ It's not an animation.
- ✓ It's an open-source software package for multidimensional data analysis and reproducible computational experiments

◆ Developers:

S. Fomel, P. Sava, T. Alkhalifah, Y. Liu, K. Schleicher, W. Symes, J. Shragge, J. Dellinger, G. Hennenfent, J.Rickett, W. Burnett et. al

AND YOU!

TRIP

What's Madagascar here?

RICE
Unconventional Wisdom

Why use Madagascar?

RICE
Unconventional Wisdom

① It's a modern package;

② It's a test-driven package;

③ It's an open-source package;

④ It use a simple, flexible, and universal data formate.

Why use Madagascar?

RICE
Unconventional Wisdom

① It's a modern package;

② It's a test-driven package;

③ It's an open-source package;

④ It use a simple, flexible, and universal data formate.

Compared with SU, SEPlib and other open-source packages,

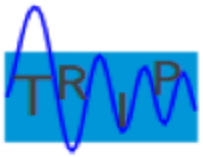
It seems to be the most potential one.

TRIP

Why use Madagascar?

RICE
Unconventional Wisdom

1. Generate an idea



Why use Madagascar?

RICE
Unconventional Wisdom

1. Generate an idea

2. Implement the idea

- **API can be C,C++,F77,F90,Matlab,Python,Java**
- **More than 1000 modules in M**

TRIP

Why use Madagascar?

RICE
Unconventional Wisdom

1. Generate an idea

2. Implement the idea

- API can be C,C++,F77,F90,Matlab,Python,Java
- More than 1000 modules in M

3. Test the idea

- Construct the workflow using Python
- More than 500 scripts, more than 5000 figures in M

TRIP

Why use Madagascar?

RICE
Unconventional Wisdom

1. Generate an idea

2. Implement the idea

- API can be C,C++,F77,F90,Matlab,Python,Java
- More than 1000 modules in M

3. Test the idea

- Construct the workflow using Python
- More than 500 scripts, more than 5000 figures in M

4. Publish the idea

- Using Python and Latex
- More than 150 papers in M

TRIP

Why use Madagascar?

RICE
Unconventional Wisdom

1. Generate an idea

2. Implement the idea

- API can be C,C++,F77,F90,Matlab,Python,Java
- More than 1000 modules in M

3. Test the idea

- Construct the workflow using Python
- More than 500 scripts, more than 5000 figures in M

4. Publish the idea

- Using Python and Latex
- More than 150 papers in M

Madagascar makes them easier

TRIP

Why use Madagascar?

RICE
Unconventional Wisdom

1. Generate an idea

2. Implement the idea

- API can be C,C++,F77,F90,Matlab,Python,Java
- More than 1000 modules in M

3. Test the idea

- Construct the workflow using Python
- More than 500 scripts, more than 5000 figures in M

4. Publish the idea

- Using Python and Latex
- More than 150 papers in M

Madagascar makes them easier

TRIP

How to learn Madagascar?

① Internet;

How to learn Madagascar?

① Internet;

② **Self-documentation;**

How to learn Madagascar?

① Internet;

② **Self-documentation;**

③ **This course;**

How to learn Madagascar?

① Internet;

② Self-documentation;

③ This course;

④ Practice by yourself.

Agenda

1. Introduction
2. Basic knowledge
3. Example
4. Discussion and Conclusion

2. Basic knowledge

RICE
Unconventional Wisdom

2.1 Installation

2.2 RSF format

2.3 Basic command

2.4 Plot command

2.5 SCONS

TRIP

2.1 Installation

RICE
Unconventional Wisdom

Prerequisites:

Ubuntu:

```
sudo apt-get install scons openmpi-bin libopenmpi-dev freeglut3-dev g++ gfortran libgd2-xpm-dev libglew1.6-dev \
libx11-dev libxaw7-dev libnetpbm10-dev swig python-dev python-scipy python-numpy libtiff4-dev scons units \
libblas-dev libcairo2-dev liblapack-dev libavcodec-dev python-epydoc
```

Mac OS:

Install necessary Mac OS X applications using:
•MacPorts, an easy-to-use system for compiling, installing, and upgrading open-source software on Mac OS X.
•Fink, a tool that brings the full world of Unix Open Source software to Mac OS X.
Installation on Mac OS X Mountain Lion requires the following:
1.Xcode: Download and install the development tools from Apple using their App Store application. In Xcode, enable and install the command-line tools in Xcode/Preferences/Downloads to get access to programs like **svn**, **make**, etc.
2.X11: Install X11 libraries from Xquartz.
3.gcc: Install the Gnu C compiler from HPC Mac OS X. The Lion version also works on Mountain Lion.
4.SEGTeX: To use **SEGTeX**, you may need TeX Live. **MacPorts** and **Fink** provide an easy way to install it with commands `sudo port install texlive` or `sudo fink install texlive`.

2.1 Installation

1. Download:

```
svn co http://svn.code.sf.net/p/rsf/code/trunk RSFSRC
```

```
svn update
```

2. Configuration:

```
./configure API=c++,f90 --prefix=/directory/where/you/want/installed
```

3. Building and installing the package

```
scons; scons install
```

4. User setup

If your shell is sh or bash, add to your `$HOME/.bashrc` and `$HOME/.bash_profile`

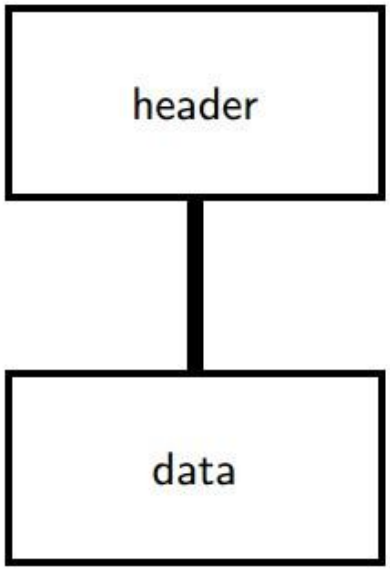
```
source $RSFROOT/share/madagascar/etc/env.sh
```

2.2 RSF format

- The main design principle behind the RSF data format is KISS ("Keep It Short and Simple"). The RSF format is borrowed from the SEPlib data format originally designed at the Stanford Exploration Project (Claerbout, 1991). The format is made as simple as possible for maximum convenience, transparency and flexibility.

Header file
- Text
- Small
- Portable

Data file
- ASCII or binary (native or XDR)
- Large (Huge)
- Path under $DATAPATH



http://www.ahay.org/wiki/Guide_to_RSF_file_format

2.3 Basic command

List of all programs: sfdoc -k .

Look for specific programs: sfdoc -k keyword

Print out documentation: sfprog without arguments

Single program: [< in.rsf] sfprog [par1=] [par2=] [...] [> out.rsf]

Multiple programs: [< in.rsf] sfprog1 [par=] | ... | sfprog2 [par=] [> out.rsf]

2.3 Basic command

Print out data: sfdisfil < in.rsf

Print out header: sfin file0.rsf

Print out data attributes: sfattr < in.rsf

Write header: sfput < in.rsf key1=val1 [...] > out.rsf

Move header and data: sfmv in.rsf out.rsf

Copy header and data: sfcp in.rsf out.rsf

Delete header and data: sfrm file1.rsf file2.rsf [...]

Packing header and data: [< in.rsf] sfprog [> out.rsf] out=stdout

Exchange dataset between systems: < in.rsf sfdd form=xdr out=stdout > out.rsf

2.3 Basic command

ASCII to RSF: echo in=in.asc data format=ascii float | sfdd form=native > out.rsf

RSF to ASCII: sfdd form=ascii out=out.asc < in.rsf > /dev/null

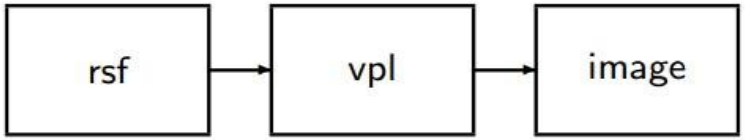
Conversion with SEG-Y:

sfsegyread tape=in.segy tfile= hfile=hfile bfile= > out.rsf
sfsegywrite tape=out.segy tfile= hfile= bfile= < in.rsf

Conversion with SU
sfsegyread su=y tape=in.su tfile= > out.rsf
sfsegywrite su=y tape=out.su tfile= < in.rsf

2.4 Plot command

- ".vpl" suffix
- Vector image can be scaled without affecting quality
- Displayed by *pen* programs
- Compact



MADAGASCAR plotting programs: `sfprog < in.rsf par= > out.vpl`

- `sfgraph`
- `sfgrey`
- `sfgrey3`
- `sfcontour`
- `sfdots`
- ...

pen progrms convert `.vpl` to images (`.eps`, `.gif`, `.png`, ...)

- `vppen`
- `xtpen`
- `pspen`
- ...

See the Plot directory

http://www.ahay.org/wiki/Tutorial#Plotting

2.5 Scons

- Build system (**S**oftware **Cons**truction)
- Written in **Python**
  - Configuration (**SConstruct** files) are Python scripts
- Built-in support for different languages
- Dependency analysis
- Parallel builds
- Cross-platform
- …

2.5 Scons

- Build system (**S**oftware **Cons**truction)
- Written in **Python**
  - Configuration (**SConstruct** files) are Python scripts
- Built-in support for different languages
- Dependency analysis
- Parallel builds
- Cross-platform
- …

Data processing with **rsf.proj**
Paper processing with **rsf.tex**
Book processing with **rsf.book**

2.5 Processing flow using Scons

- **Fetch('filename','dirname')**
  - A rule for downloading files from a server
- **Flow('target','source','command')**
  - A rule for making target from source
- **Plot('target','source','command')**
  - Like Flow but generates a figure file
- **Result('target','source','command')**
  - Like Plot but generates a final result

Agenda

1. Introduction
2. Basic knowledge
3. Example
4. Discussion and Conclusion

3. Example

How to realize extended modeling idea

from implementation to documentation in Madagascar?

3. Example

1. Generate an idea

- Understand the extended modeling idea from Dr. Symes

2. Implement the idea

- Modify the code of conventional modeling written by Dr. Sava

3. Test the idea

- Construct the workflow with Python to test your code
- More than 500 scripts in M may help you

4. Publish the idea

- Use Python and Latex to generate the paper
- More than 150 papers in M may help you.

3. Example

How to realize extended modeling idea

from implementation to documentation in Madagascar?

Please test the tar file I have sent to you

Agenda

1. Introduction
2. Basic knowledge
3. Example
4. Discussion and Conclusion

Discussion & Conclusion

1. Madagascar makes our research easier.

2. Reproducibility is the most appealing features in Madagascar. It can make our research more valuable and known by more people.

3. Madagascar can help us maintain our codes as everyone is developer.

4. Introducing IWAVE/IWAVE++ into Madagascar is very meaningful.

Enjoy it!

Q & A

http://www.ahay.org/wiki/Main_Page