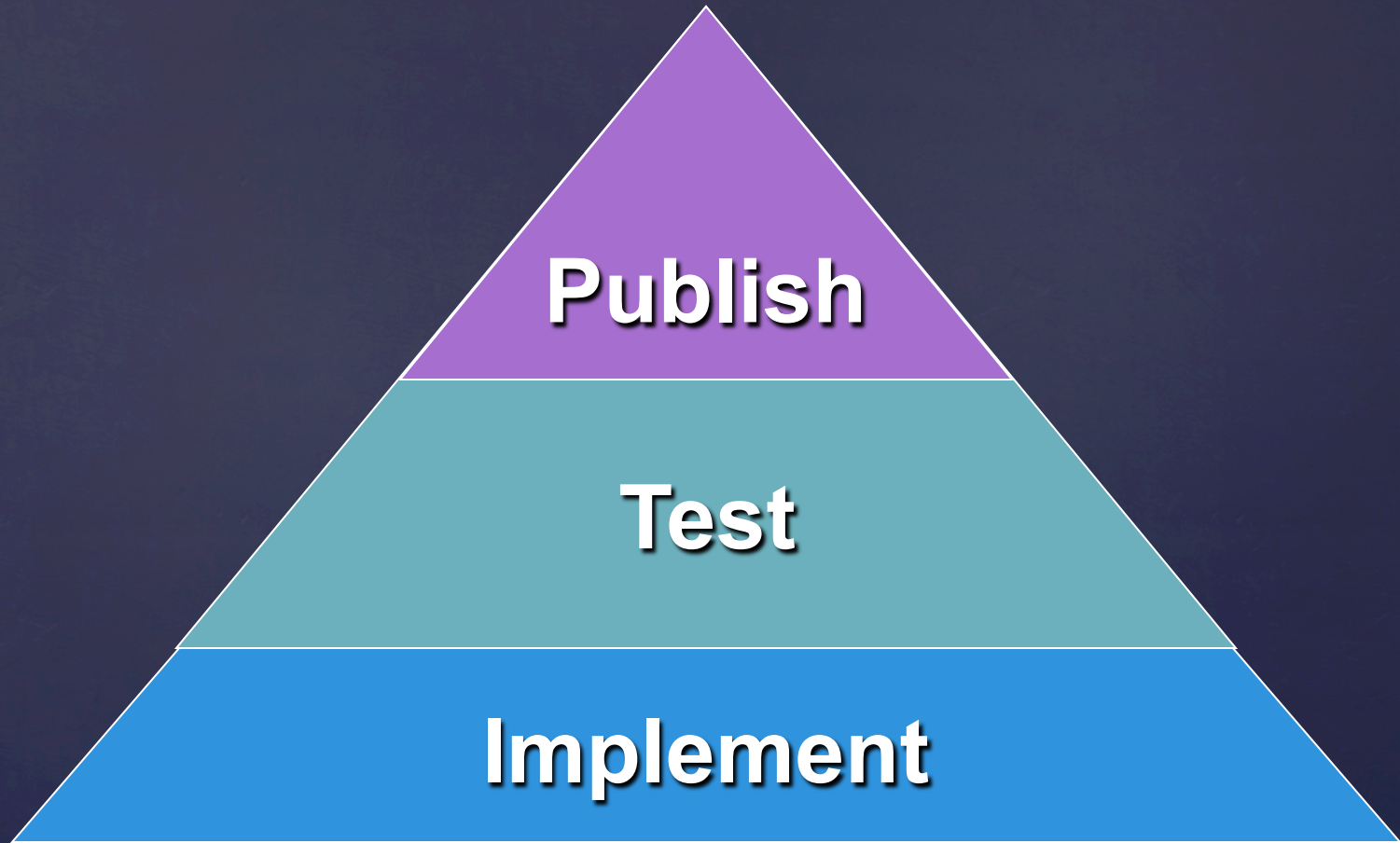


Madagascar Fundamentals

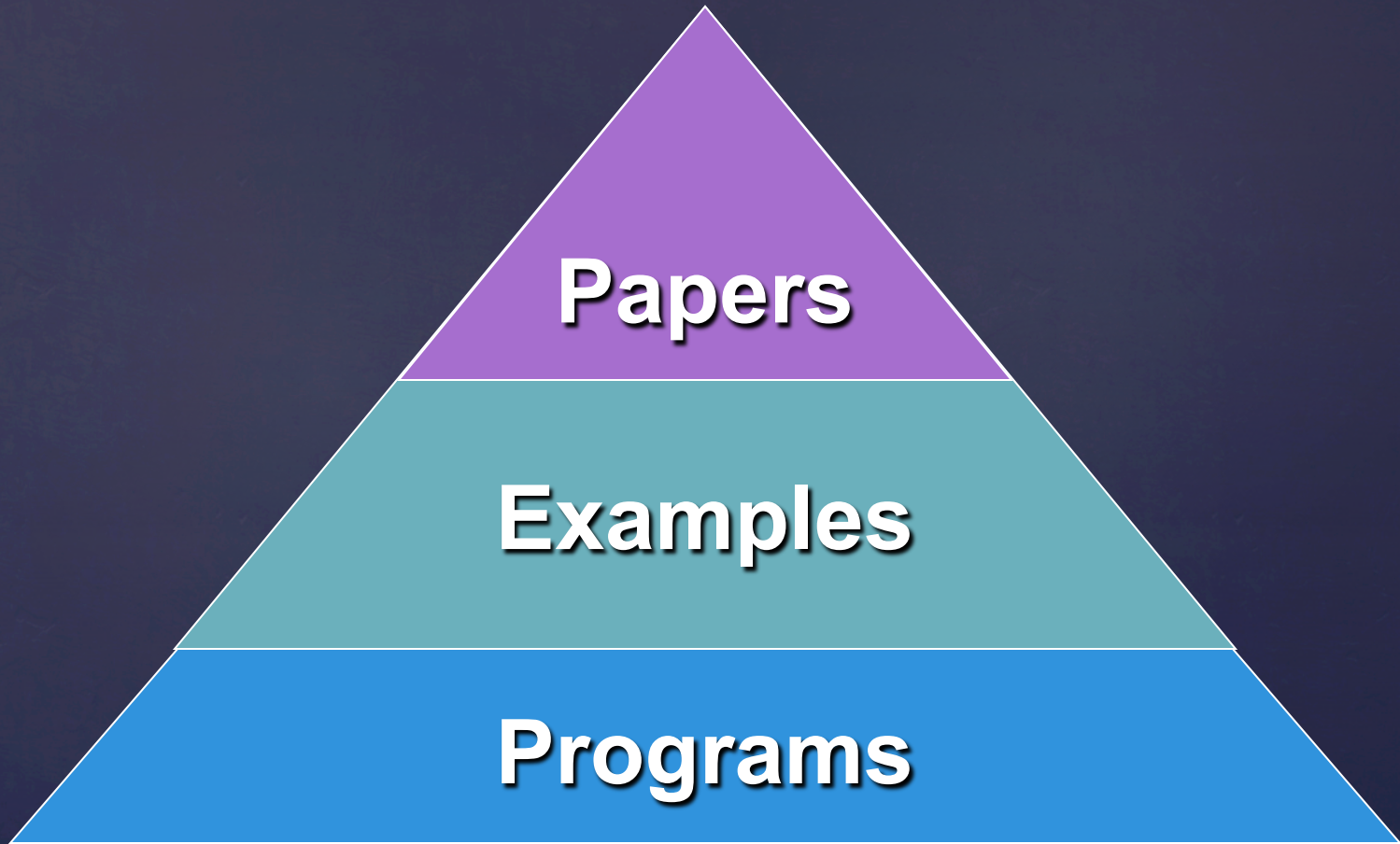
{ Sergey Fomel
July 20, 2012
Austin, Texas



Madagascar Pyramid



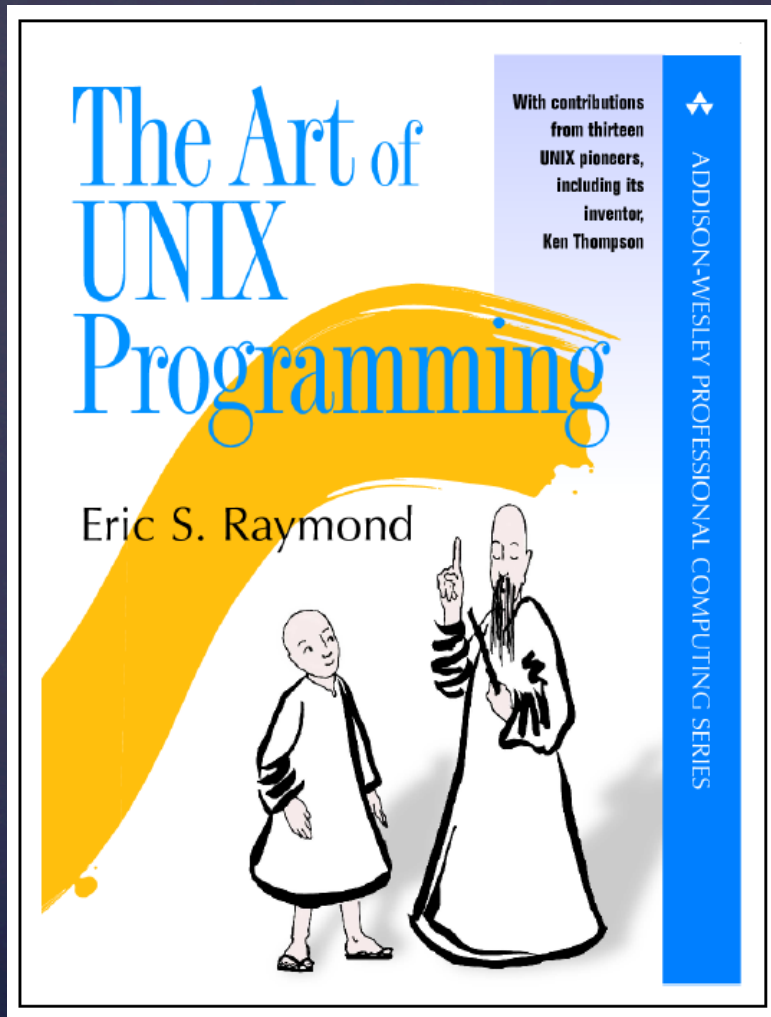
Madagascar Pyramid



- & RSF file format
- & Command-line usage
- & Introduction to SCons/Python

Outline

Madagascar Design



- ⌘ Data arrays as file objects on disk
- ⌘ “Write programs that do one thing and do it well. Write programs to work together. Write programs to handle text streams, because that is a universal interface.”
Doug McIlroy



& “To design a perfect anti-Unix, make all file formats binary and opaque, and require heavyweight tools to read and edit them. [...] If you feel an urge to design a complex binary file format, or a complex binary application protocol, it is generally wise to lie down until the feeling passes.” *Eric Raymond*

- & Principles
- & Header and data files
 - ⌘ Datapath
 - ⌘ Type
 - ⌘ Form
 - ⌘ Hypercube
- & Compatibility with other formats
 - ⌘ SEPlib
 - ⌘ SEG Y
 - ⌘ SU

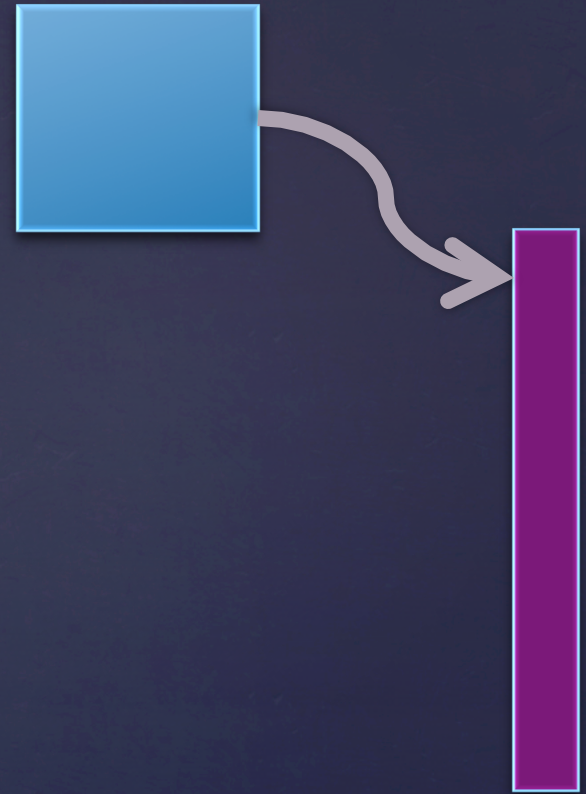
RSF File Format



- ⌘ datapath= parameter on the command line.
- ⌘ DATAPATH environmental variable
- ⌘ .datapath file in the current directory.
- ⌘ .datapath file in the user home directory.
- ⌘ Current directory.

Datapath Rules

& uchar,
& char
& int
& float
& complex
& short
& double

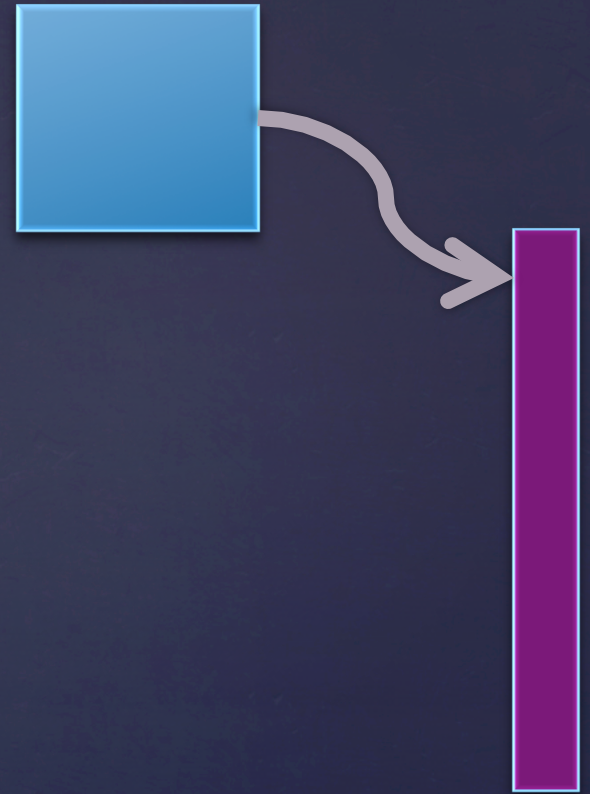


Data Type

& native

& XDR

& ASCII



Data Form

& n1, n2, ...

& d1, d2, ...

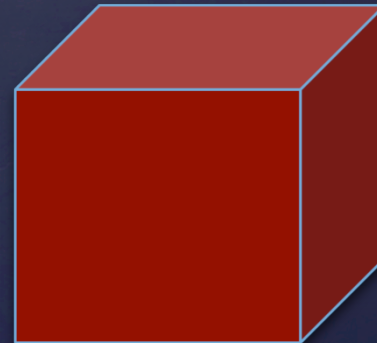
& o1, o2, ...

& label1, label2, ...

& unit1, unit2, ...

& label

& unit



Hypercube

- & RSF file format
- & Command-line usage
- & Introduction to SCons/Python

Outline

& Make (1977)

⌘ “Sendmail and **make** are two well known programs that are pretty widely regarded as originally being *debugged into existence*. That's why their command languages are so poorly thought out and difficult to learn. It's not just you - everyone finds them troublesome.” *P. van der Linden*

& GNU Make (1988)

& SCons (2000)

Evolution of Build Systems



- ⌘ Build system (**Software Construction**)
- ⌘ Written in **Python**
 - ⌘ Configuration files are Python scripts
- ⌘ Built-in support for different languages
- ⌘ Dependency analysis
- ⌘ Parallel builds
- ⌘ Cross-platform



What is SCons?

- & **Dynamic programming language**
- & **Clear, readable syntax**
 - ⌘ “friendly and easy to learn”
- & **Full modularity**
 - ⌘ “batteries included”
- & **Integrates with other languages**
 - ⌘ “plays well with others”
- & **Free and open-source**

What is Python?



⌘ "Python has been an important part of Google since the beginning, and remains so as the system grows and evolves. Today dozens of Google engineers use Python, and we're looking for more people with skills in this language." *Peter Norvig*

⌘ "Python is fun, free, runs on a broad range of platforms and has a large library of sophisticated modules, including numerical. It meets all our criteria for a first language." *John Scales & Hans Ecke*

Who uses Python?  python™

1. Variables and strings
2. Lists and dictionaries
3. For loop
4. If/else, indentation
5. Functions and modules



Python in 5 Easy Steps

```
env=Environment()  
env.Program("program", "file.c")
```

SCons for Program Compilation

```
scons -h  
scons -n  
scons -c  
scons -Q  
scons -j
```


- & 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

Madagascar Processing:
rsf.proj

- ⌘ RSF file format is simple, represents a multidimensional array as text/binary
- ⌘ SCons is a powerful and convenient build system adopted by Madagascar for data processing and reproducible documents
 - ⌘ Configuration files are Python scripts

Summary