Hello, World!

Design and specifications for SLIM's software framework

Progress and road map

SLIM group at UBC EOS

February 2008

SLIM
Seismic Laboratory for
Imaging and Modeling

About

- Our working environment
- Moving towards Python
- From serial to parallel
- Challenges
- Our released software
- SINBAD software release - February 2008
- Future directions
- Summary

SLIM
Seismic Laboratory for Imaging and Modeling

Our challenges

- Seismic applications are:
  - extremely large
  - computations intensive
  - IO intensive
- SLIM's technology is driven by multi-d (d>2) non-seperable transforms => need to move out of core
- Academic environment calls for
  - code reuse & reproducibility
  - fast learning curves

Our software

- In-core computations:
  - Matlab (Octave) and Python
- Out-off-core seismic tools:
  - Seismic Unix, SEP, Delphi, RSF
- Scripting tools:
  - Unix shells, SLIMpy, SCons
- Parallelism
  - PETSc (MPI), SLIMpy, pcmd

Moving towards Python

- In-core (element-wise operations) for prototyping:
  - NumPy
  - pyCurveLab
- Out-off-core:
  - SLIMpy(2) -> high-level algorithm programing (ANAs)
  - SCons -> simple scripting and dependency flow
  - Madagascar -> only for low-level element-wise operations; not for scripting algorithms

SLIM
Seismic Laboratory for Imaging and Modeling

Towards-Python benefits

- Independent of proprietary software licenses
- Reusability
  - Research focused on algorithms not “coding”
- Faster learning curve
  - Easier/faster prototyping and development
  - Easier/faster implementation of new algorithms

From serial to parallel

- Our iterative algorithms require lots of computations
- We need to:
  - accelerate computations
  - optimize utilization of resources in HPC environment
    - distribute memory footprint
    - distribute disposable IO to local filesystems
- To both compute faster and handle bigger problems

From serial ...

- Madagascar takes advantage of SMP systems:
  - through SCons command-line options to support parallel flows
  - recently introduced OMP and MPI utilities on a very trivial level to wrap linear operators by distributing data from a **single** pipe (over the slow direction), applying an operator to each file, and collecting results onto a **single** pipe again.
- SLIMpy have supported parallel flows on SMP systems through option by mechanism similar to SCons
- It turns out that it is nowhere near-enough.

... to parallel

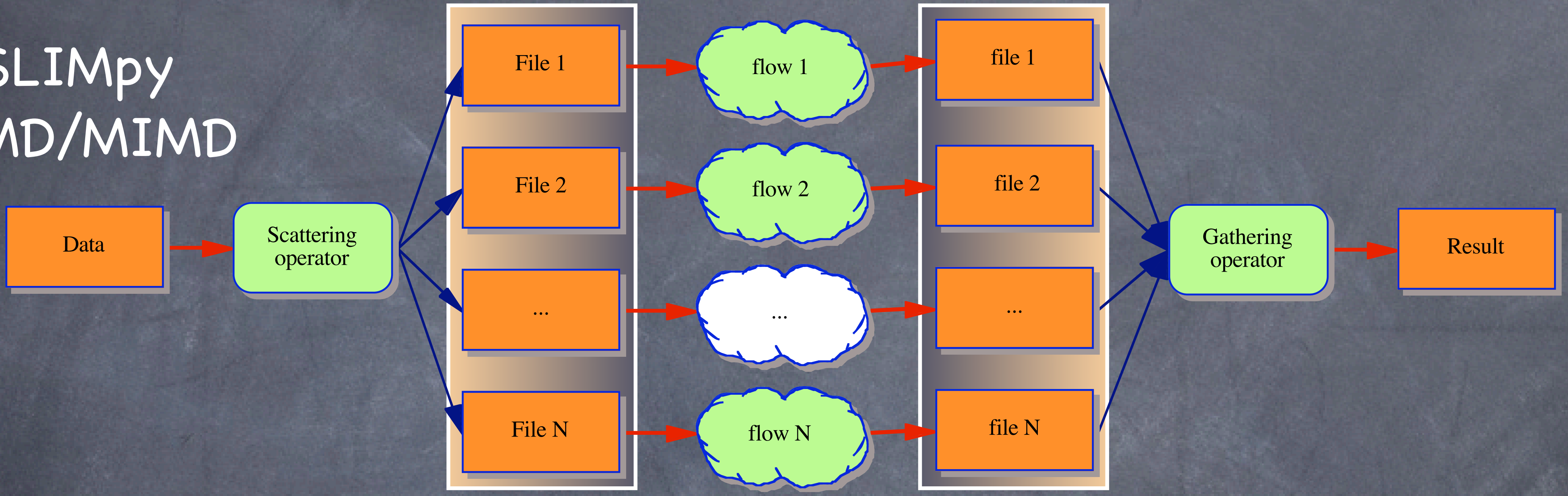
- We discarded Madagascar approach due to its limitations:
  - overhead of scattering/collecting operation for each operator
  - limited ability to scatter data in parallel IO fashion
  - inflexible scattering topology
- We have recently extended SLIMpy with:
  - prototype for parallel processing for both CPU and IO
  - low-level Madagascar-type utilities that support that prototype
    - flexible scatterig/collecting operator
    - XML metafiles

... to parallel

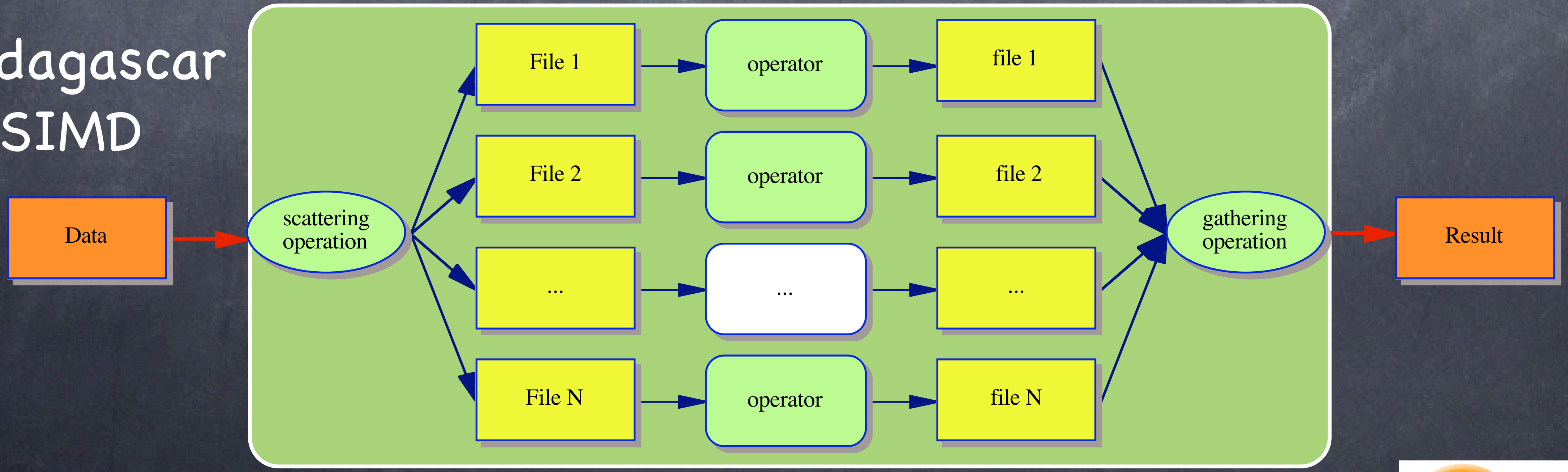
- The parallel-prototype in SLIMpy supports any level of parallelism nearly transparently to user: from serial, through embarrassingly parallel, to complex MPI domain-decomposition by:
  - data scattering/gathering operators
  - pipe-execution on remote CPUs
  - off-loading data from global file systems to CPU's local /tmp -> effectively providing parallel IO
  - parallel extension of RSF files - XML metafiles
  - out-off-core equivalent of both SIMD and MIMD computing models

Processing flows

SLIMpy
SIMD/MIMD



Madagascar
SIMD



Challenges

- Computational bottlenecks to be optimized:
  - long pipes, processing many files at the same time, introduce competitive IO on local file-system and hurt cumulative IO performance
  - long pipes can exceed memory resources
  - sophisticated algorithm can be IO costly; some suffer from in-core to out-off-core transition and have to be optimized

SLIM
Seismic Laboratory for Imaging and Modeling

Challenges

- Madagascar limitations to overcome:
  - lack of support for distributed file-components
  - problem as simple as:
    - limited integer representation of dimensions in RSF files; redundancy of either CurveLab or SurfBox transform can easily exceed the 4-byte integer limit even for relatively small problems
    - numerical precision; i.e., 4-byte float representation in Madagascar

SLIM
Seismic Laboratory for Imaging and Modeling

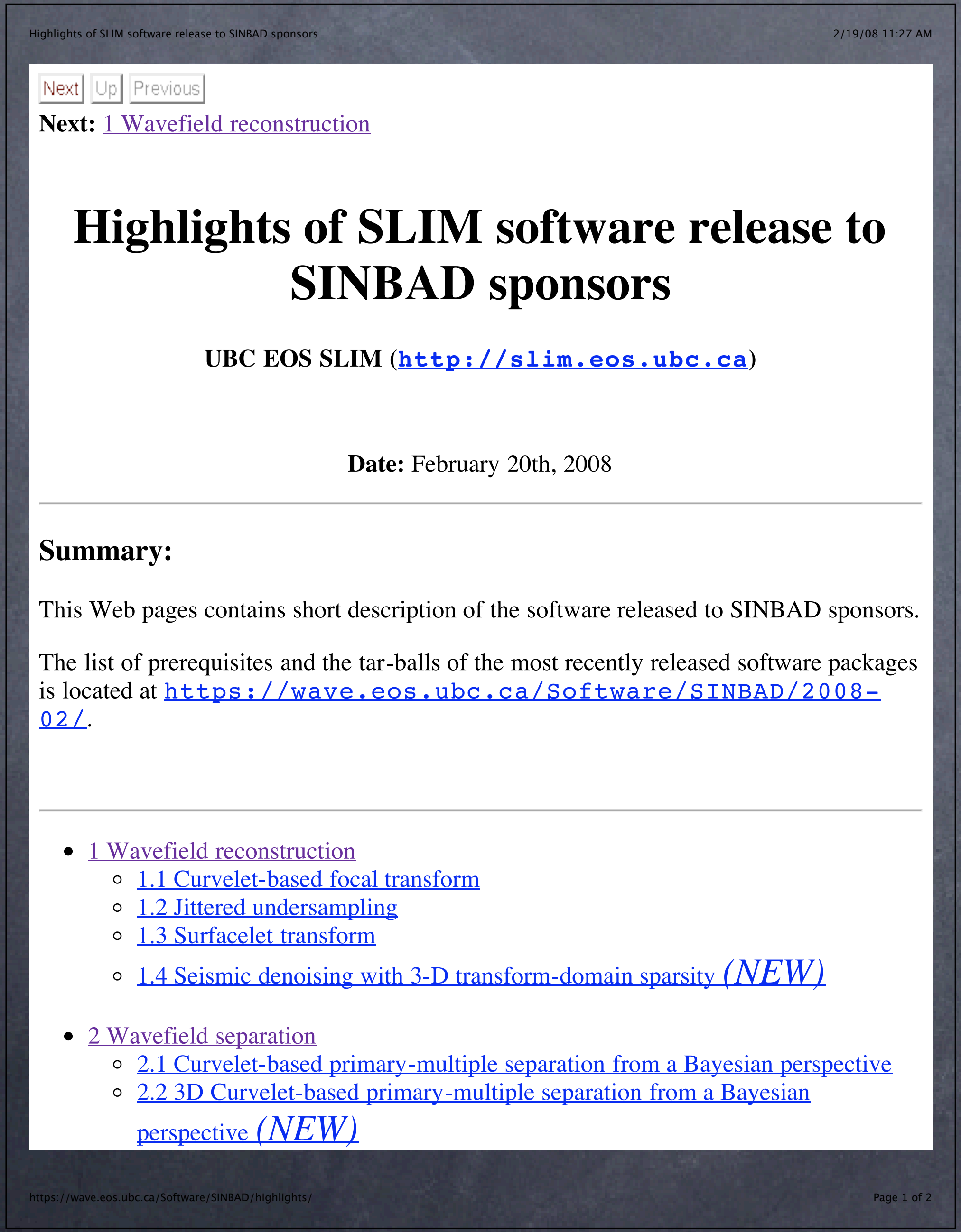
Our released software

- To public domain
  - Madagascar -> contribution in user/slim
  - pyCurveLab
- SINBAD software release ...

Software release - Feb 2008

- Highlights:
  - https://wave.eos.ubc.ca/Software/SINBAD/highlights/
- Packages:
  - https://wave.eos.ubc.ca/Software/SINBAD/2008-02/
- Support: a number of mailing lists (see below)

Software highlights



Highlights of SLIM software release to SINBAD sponsors 2/19/08 11:27 AM

Next | Up | Previous

Next: 1 Wavefield reconstruction

Highlights of SLIM software release to SINBAD sponsors

UBC EOS SLIM (http://slim.eos.ubc.ca)

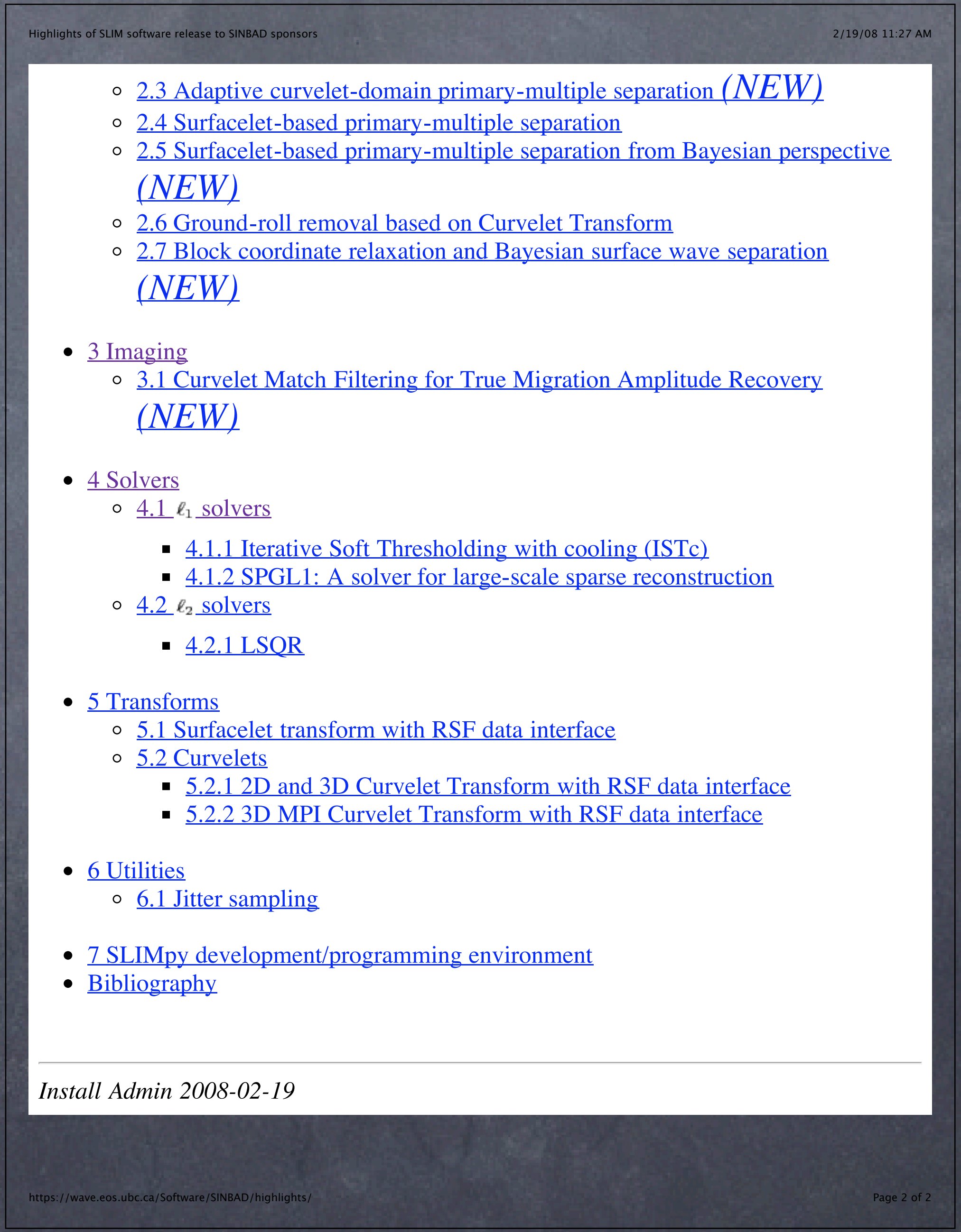
Date: February 20th, 2008

Summary:

This Web pages contains short description of the software released to SINBAD sponsors.

The list of prerequisites and the tar-balls of the most recently released software packages is located at https://wave.eos.ubc.ca/Software/SINBAD/2008-02/.

https://wave.eos.ubc.ca/Software/SINBAD/highlights/ Page 1 of 2



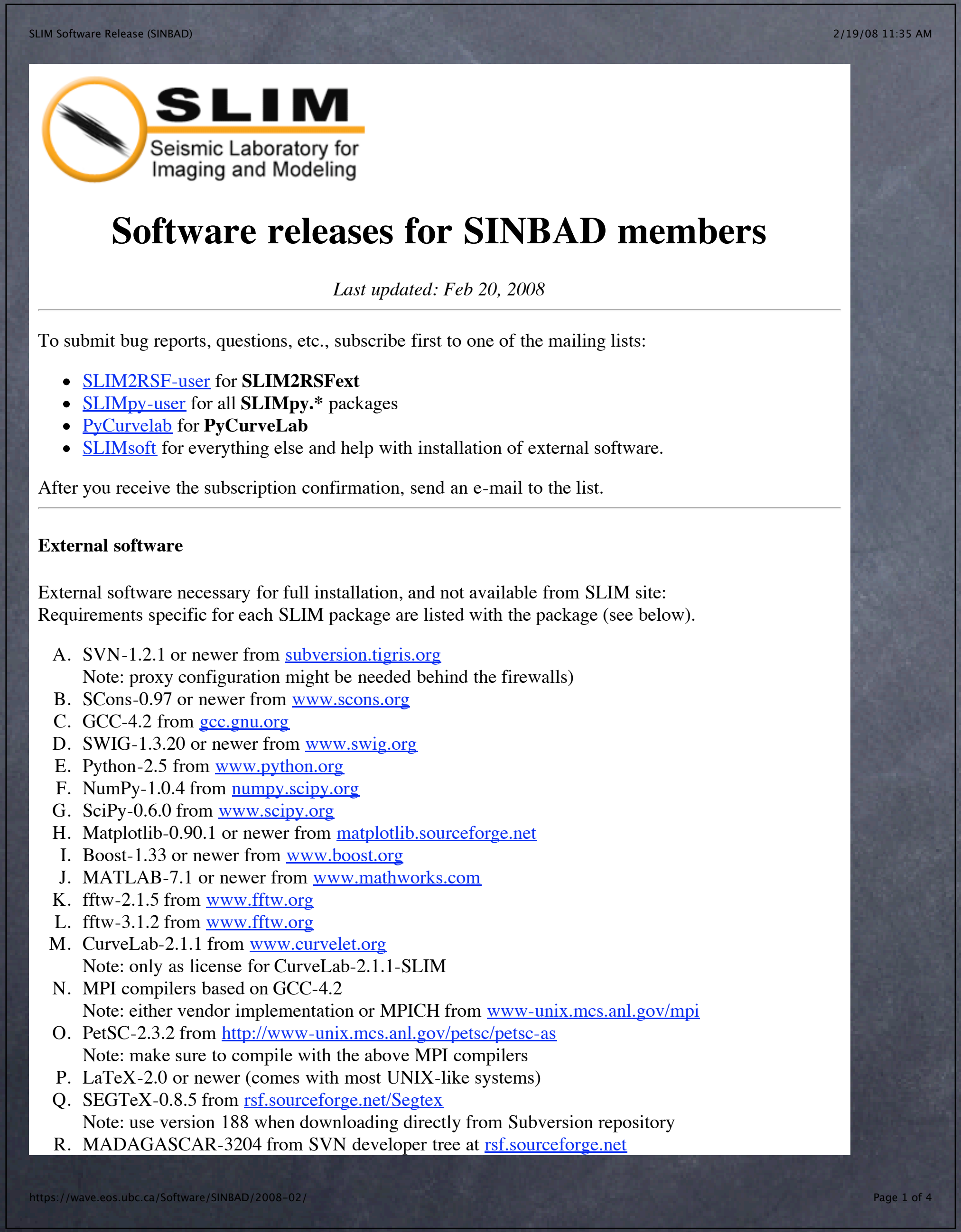
Highlights of SLIM software release to SINBAD sponsors 2/19/08 11:27 AM

Install Admin 2008-02-19

https://wave.eos.ubc.ca/Software/SINBAD/highlights/ Page 2 of 2

SLIM
Seismic Laboratory for Imaging and Modeling

Software release



SLIM
Seismic Laboratory for
Imaging and Modeling

Software releases for SINBAD members

Last updated: Feb 20, 2008

To submit bug reports, questions, etc., subscribe first to one of the mailing lists:

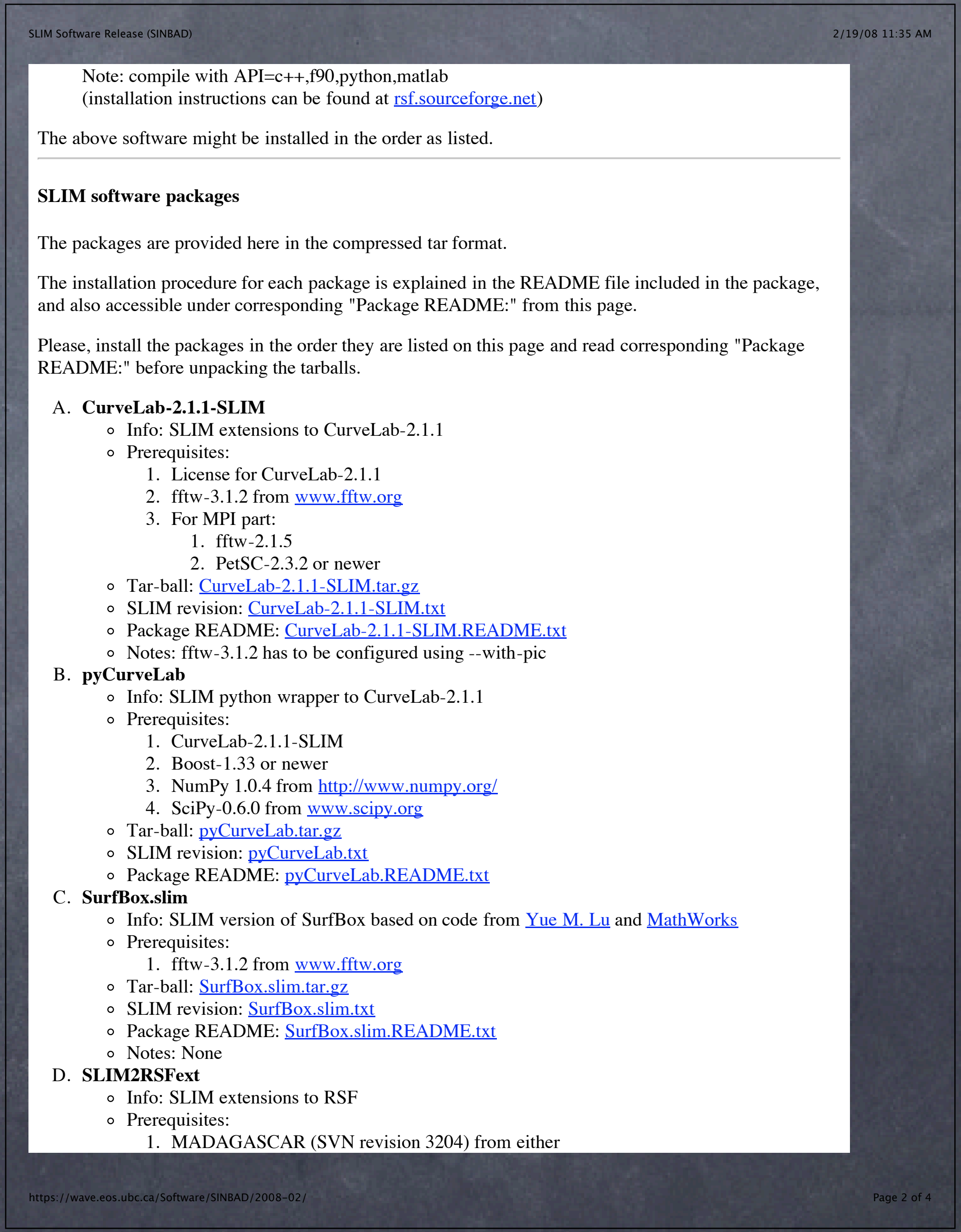
- SLIM2RSF-user for **SLIM2RSFext**
- SLIMpy-user for all **SLIMpy.*** packages
- PyCurvelab for **PyCurveLab**
- SLIMsoft for everything else and help with installation of external software.

After you receive the subscription confirmation, send an e-mail to the list.

External software

External software necessary for full installation, and not available from SLIM site:
Requirements specific for each SLIM package are listed with the package (see below).

A. SVN-1.2.1 or newer from subversion.tigris.org
Note: proxy configuration might be needed behind the firewalls)
B. SCons-0.97 or newer from www.scons.org
C. GCC-4.2 from gcc.gnu.org
D. SWIG-1.3.20 or newer from www.swig.org
E. Python-2.5 from www.python.org
F. NumPy-1.0.4 from numpy.scipy.org
G. SciPy-0.6.0 from www.scipy.org
H. Matplotlib-0.90.1 or newer from matplotlib.sourceforge.net
I. Boost-1.33 or newer from www.boost.org
J. MATLAB-7.1 or newer from www.mathworks.com
K. fftw-2.1.5 from www.fftw.org
L. fftw-3.1.2 from www.fftw.org
M. CurveLab-2.1.1 from www.curvelet.org
Note: only as license for CurveLab-2.1.1-SLIM
N. MPI compilers based on GCC-4.2
Note: either vendor implementation or MPICH from www-unix.mcs.anl.gov/mpi
O. PetSC-2.3.2 from http://www-unix.mcs.anl.gov/petsc/petsc-as
Note: make sure to compile with the above MPI compilers
P. LaTeX-2.0 or newer (comes with most UNIX-like systems)
Q. SEGTeX-0.8.5 from rsf.sourceforge.net/Segtex
Note: use version 188 when downloading directly from Subversion repository
R. MADAGASCAR-3204 from SVN developer tree at rsf.sourceforge.net



Note: compile with API=c++,f90,python,matlab
(installation instructions can be found at rsf.sourceforge.net)

The above software might be installed in the order as listed.

SLIM software packages

The packages are provided here in the compressed tar format.

The installation procedure for each package is explained in the README file included in the package, and also accessible under corresponding "Package README:" from this page.

Please, install the packages in the order they are listed on this page and read corresponding "Package README:" before unpacking the tarballs.

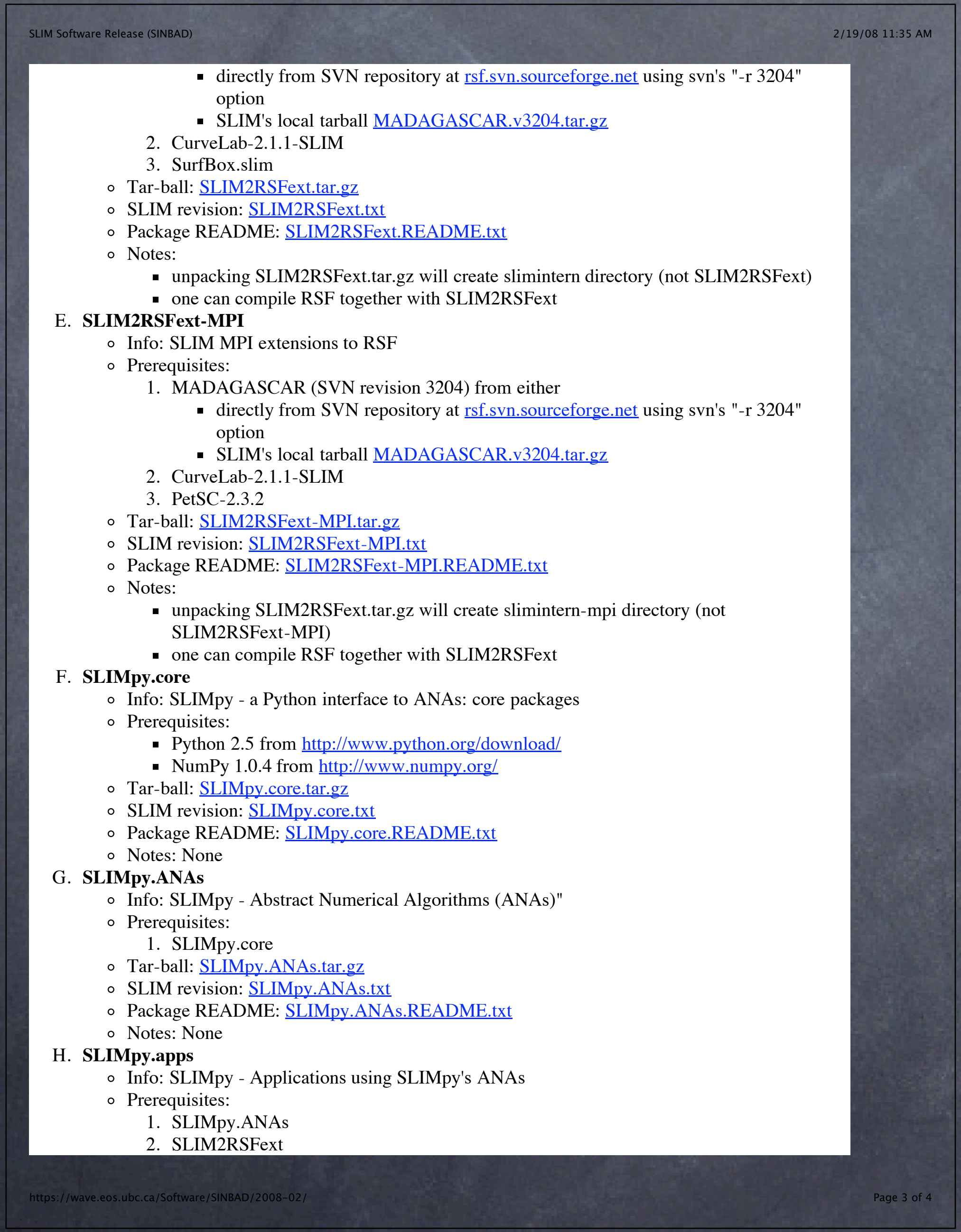
A. **CurveLab-2.1.1-SLIM**
   - Info: SLIM extensions to CurveLab-2.1.1
   - Prerequisites:
     1. License for CurveLab-2.1.1
     2. fftw-3.1.2 from www.fftw.org
     3. For MPI part:
        1. fftw-2.1.5
        2. PetSC-2.3.2 or newer
   - Tar-ball: CurveLab-2.1.1-SLIM.tar.gz
   - SLIM revision: CurveLab-2.1.1-SLIM.txt
   - Package README: CurveLab-2.1.1-SLIM.README.txt
   - Notes: fftw-3.1.2 has to be configured using --with-pic

B. **pyCurveLab**
   - Info: SLIM python wrapper to CurveLab-2.1.1
   - Prerequisites:
     1. CurveLab-2.1.1-SLIM
     2. Boost-1.33 or newer
     3. NumPy 1.0.4 from http://www.numpy.org/
     4. SciPy-0.6.0 from www.scipy.org
   - Tar-ball: pyCurveLab.tar.gz
   - SLIM revision: pyCurveLab.txt
   - Package README: pyCurveLab.README.txt

C. **SurfBox.slim**
   - Info: SLIM version of SurfBox based on code from Yue M. Lu and MathWorks
   - Prerequisites:
     1. fftw-3.1.2 from www.fftw.org
   - Tar-ball: SurfBox.slim.tar.gz
   - SLIM revision: SurfBox.slim.txt
   - Package README: SurfBox.slim.README.txt
   - Notes: None

D. **SLIM2RSFext**
   - Info: SLIM extensions to RSF
   - Prerequisites:
     1. MADAGASCAR (SVN revision 3204) from either

Software release



- directly from SVN repository at rsf.svn.sourceforge.net using svn's "-r 3204" option
- SLIM's local tarball MADAGASCAR.v3204.tar.gz

2. CurveLab-2.1.1-SLIM
3. SurfBox.slim

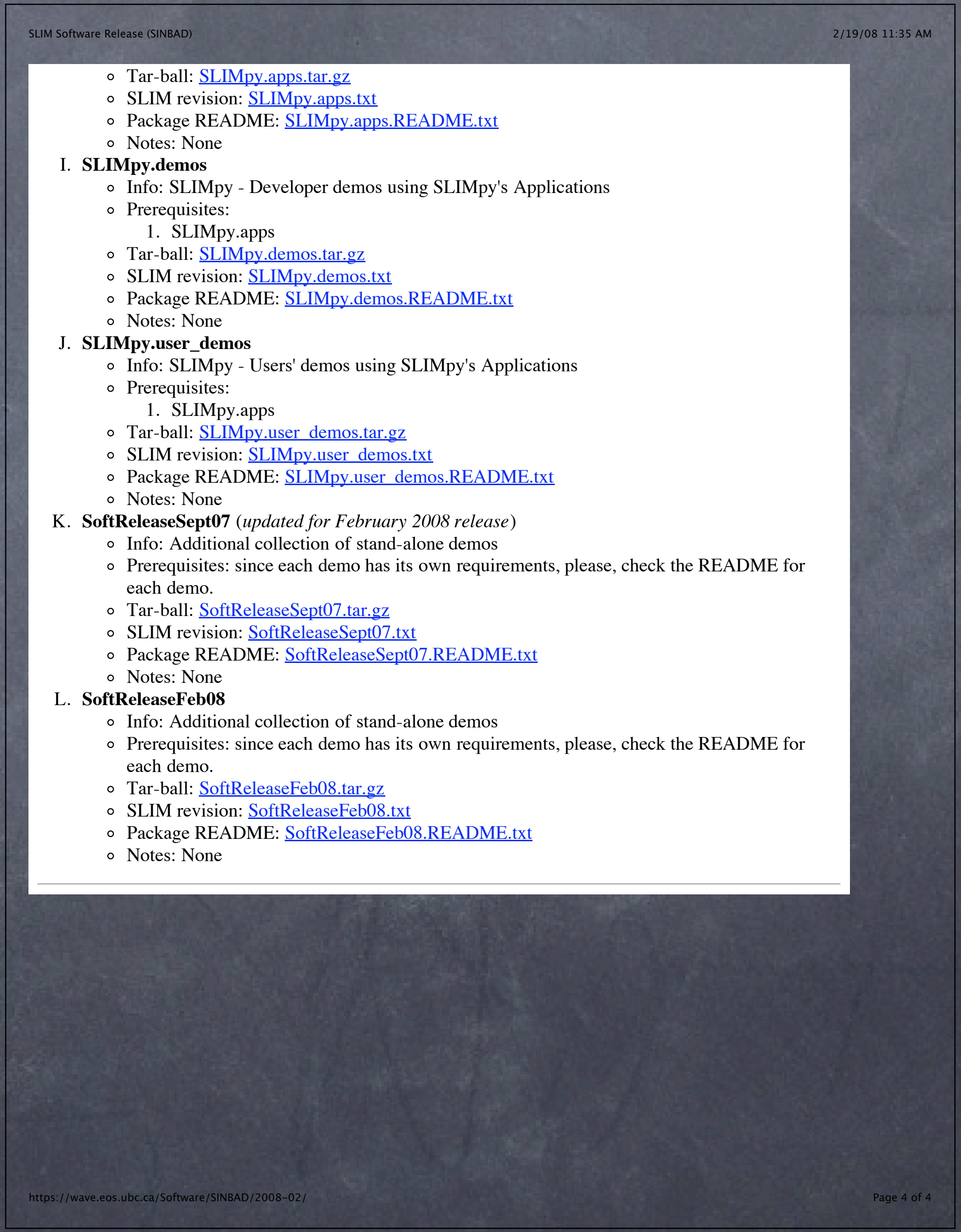
- Tar-ball: SLIM2RSFext.tar.gz
- SLIM revision: SLIM2RSFext.txt
- Package README: SLIM2RSFext.README.txt
- Notes:
  - unpacking SLIM2RSFext.tar.gz will create slimintern directory (not SLIM2RSFext)
  - one can compile RSF together with SLIM2RSFext

E. **SLIM2RSFext-MPI**
- Info: SLIM MPI extensions to RSF
- Prerequisites:
  1. MADAGASCAR (SVN revision 3204) from either
     - directly from SVN repository at rsf.svn.sourceforge.net using svn's "-r 3204" option
     - SLIM's local tarball MADAGASCAR.v3204.tar.gz
  2. CurveLab-2.1.1-SLIM
  3. PetSC-2.3.2
- Tar-ball: SLIM2RSFext-MPI.tar.gz
- SLIM revision: SLIM2RSFext-MPI.txt
- Package README: SLIM2RSFext-MPI.README.txt
- Notes:
  - unpacking SLIM2RSFext.tar.gz will create slimintern-mpi directory (not SLIM2RSFext-MPI)
  - one can compile RSF together with SLIM2RSFext

F. **SLIMpy.core**
- Info: SLIMpy - a Python interface to ANAs: core packages
- Prerequisites:
  - Python 2.5 from http://www.python.org/download/
  - NumPy 1.0.4 from http://www.numpy.org/
- Tar-ball: SLIMpy.core.tar.gz
- SLIM revision: SLIMpy.core.txt
- Package README: SLIMpy.core.README.txt
- Notes: None

G. **SLIMpy.ANAs**
- Info: SLIMpy - Abstract Numerical Algorithms (ANAs)"
- Prerequisites:
  1. SLIMpy.core
- Tar-ball: SLIMpy.ANAs.tar.gz
- SLIM revision: SLIMpy.ANAs.txt
- Package README: SLIMpy.ANAs.README.txt
- Notes: None

H. **SLIMpy.apps**
- Info: SLIMpy - Applications using SLIMpy's ANAs
- Prerequisites:
  1. SLIMpy.ANAs
  2. SLIM2RSFext



- Tar-ball: SLIMpy.apps.tar.gz
- SLIM revision: SLIMpy.apps.txt
- Package README: SLIMpy.apps.README.txt
- Notes: None

I. **SLIMpy.demos**
- Info: SLIMpy - Developer demos using SLIMpy's Applications
- Prerequisites:
  1. SLIMpy.apps
- Tar-ball: SLIMpy.demos.tar.gz
- SLIM revision: SLIMpy.demos.txt
- Package README: SLIMpy.demos.README.txt
- Notes: None

J. **SLIMpy.user_demos**
- Info: SLIMpy - Users' demos using SLIMpy's Applications
- Prerequisites:
  1. SLIMpy.apps
- Tar-ball: SLIMpy.user_demos.tar.gz
- SLIM revision: SLIMpy.user_demos.txt
- Package README: SLIMpy.user_demos.README.txt
- Notes: None

K. **SoftReleaseSept07** (*updated for February 2008 release*)
- Info: Additional collection of stand-alone demos
- Prerequisites: since each demo has its own requirements, please, check the README for each demo.
- Tar-ball: SoftReleaseSept07.tar.gz
- SLIM revision: SoftReleaseSept07.txt
- Package README: SoftReleaseSept07.README.txt
- Notes: None

L. **SoftReleaseFeb08**
- Info: Additional collection of stand-alone demos
- Prerequisites: since each demo has its own requirements, please, check the README for each demo.
- Tar-ball: SoftReleaseFeb08.tar.gz
- SLIM revision: SoftReleaseFeb08.txt
- Package README: SoftReleaseFeb08.README.txt
- Notes: None

SLIM
Seismic Laboratory for Imaging and Modeling

Software release

- MADAGASCAR.v3204 - SLIM contr. in user/slim (public)
- CurveLab-2.1.1-SLIM - our extensions to CurveLab
- pyCurveLab - Python interface to CurveLab (public)
- SurfBox.slim - our extensions to SurfBox
- SLIM2RSFext - RSF interface to CurveLab and SurfBox
- SLIM2RSFext-MPI - support for MPI
  - 3D-FDCT
  - MPI domain decomposition
  - parallel IO through XML metafiles

Software release

- SLIMpy packages
  - SLIMpy.core - core utilities
  - SLIMpy.ANAs - abstract numerical algorithms
  - SLIMpy.apps - applications
  - SLIMpy.demos - developer demos (probably to retire)
  - SLIMpy.user_demos - user demos (future mainstream)
- Demo colections:
  - SoftReleaseSept07 - MATLAB / Python / reproducible research
  - SoftReleaseFeb08 - Python / reproducible research

SLIM
Seismic Laboratory for
Imaging and Modeling

Software support

- Mailing lists:
  - SLIM2RSF-user:
    - http://slim.eos.ubc.ca/mailman/listinfo/slim2rsf-user
  - SLIMpy-user
    - http://slim.eos.ubc.ca/mailman/listinfo/slimpy-user
  - pyCurveLab
    - http://slim.eos.ubc.ca/mailman/listinfo/pycurvelab
  - SLIMsoft
    - http://slim.eos.ubc.ca/mailman/listinfo/slimsoft

SLIM

Seismic Laboratory for Imaging and Modeling

Future directions

- new operators through
  - command-line based seismic tools
  - SLIMpy operators
- new ANA algorithms in SLIMpy
- contributing to Madagascar
  - XML metafiles for parallel IO
  - named pipes to support pipes in distributed parallel IO
  - ...
- releasing SLIMpy core to community (support other seismic packages)
- programing moving near-entirely to SLIMpy (except for fast prototyping)

SLIM Seismic Laboratory for Imaging and Modeling

Summary

- We have built a large software framework for seismic-imaging applications.
- We have implemented new algorithms in this framework.
- We have build a number of low-level tools to support those algorithms.
- ... and the work goes on ...

We make a huge effort in streamlining/accelerating algorithm development and allowing researchers to focus on research rather than software development, and that efforts starts to pay off.

SLIM

Seismic Laboratory for Imaging and Modeling

Acknowledgments

This presentation was carried out as part of the SINBAD project with financial support, secured through ITF, from the following organizations: BG, BP, Chevron, ExxonMobil, and Shell. SINBAD is part of the collaborative research & development (CRD) grant number 334810-05 funded by the Natural Science and Engineering Research Council (NSERC).

SLIM
Seismic Laboratory for Imaging and Modeling

Good bye, World!