Scripting for HPC

Drew Dolgert
Cornell Center for Advanced Computing
December 8, 2010

Assembler
Roming Rohde
mitp
UNIX
HAYDEN BOOKS
THINK C
Standard Libraries Reference
The Waite Group's
New C Primer Plus
THE C PROGRAMMING LANGUAGE
THE C++ PROGRAMMING LANGUAGE
THIRD EDITION
The Complete Reference
C++
Fourth Edition
Schildt
OSBORNE
Modern C++ Design
Alexandrescu
STL Tutorial and Reference Guide
Musser
Saini
INSIDE C# 2
Second Edition
C#
IN A NUTSHELL
O'REILLY
C# Primer
C# Precisely
C# Developer's Guide to ASP.NET, XML, and ADO.NET
C# Class Design Handbook:
Coding Effective Classes
McManus • Kinsman
Using csh & tcsh
Migrating to Fortran 90
Fortran 90 Explained
OXFORD
Fortran 95/2003
for Scientists and Engineers
Chapman
INSIDE MICROSOFT
.NET IL ASSEMBLER
Lidin
JAVA
IN A NUTSHELL
FLANAGAN
O'REILLY
lex & yacc
LaTeX
A Document Preparation System
Lamport
LaTeX Web Companion
Goossens • Rahtz
TEX
Kopka
Daly
WOLFRAM
THE MATHEMATICA BOOK
INTRODUCTION TO Maple
Monad
ML for the working programmer
Programming Perl
Schilli
PERL POWER!
PostScript
LANGUAGE REFERENCE
wxPython IN ACTION
Python Essential Reference
Programming Python
Tcl and the Tk Toolkit
Ousterhout
WINDOWS NT
SHELL SCRIPTING
modern compiler implementation in C:
basic techniques
Aho
Sethi
Ullman
Compilers
Principles, Techniques, and Tools
Structure and Interpretation of Computer Programs
Abelson and Sussman
COMPUTER ORGANIZATION & DESIGN
THE HARDWARE/SOFTWARE INTERFACE
Patterson
Hennessy
Computer Organization & Design
COMPUTER ALGORITHMS

What's a scripting language?

- Python
- Perl
- R
- Lua
- Tcl
- Ruby
- Bash, Csh

What scripting language is more popular than all of these?

What's It Look Like?

- Go to ~/labs/python
- Look at simple.py.
- Run it with
  python simple.py 100

What do you notice about how it differs from C, Fortran?

Dynamic = Built to Play

- Graphics – Matplotlib, VTK, gnuplot
- Numerics – Numpy, Scipy
- Data Transformation – XML, binary, HDF
- Networking – MPI, TCP/IP, web services

Can't Fortran do these things?

Luna the rabbit says RELAX.

Exercise: Pure Math Speed

- $ cd python
- Add python and numpy modules.
- $ time python pure_invert.py 100
- $ time python numpy_invert.py 100
- Is the code reasonable?


YOU
ARE
BEAUTIFUL

Top Level Logic

Integration, Larger loops, Middle Stuff

Innermost Subroutines

Replacing Top Level

Initialize()

Decide what you store.

Loop until done:

  Integrate()

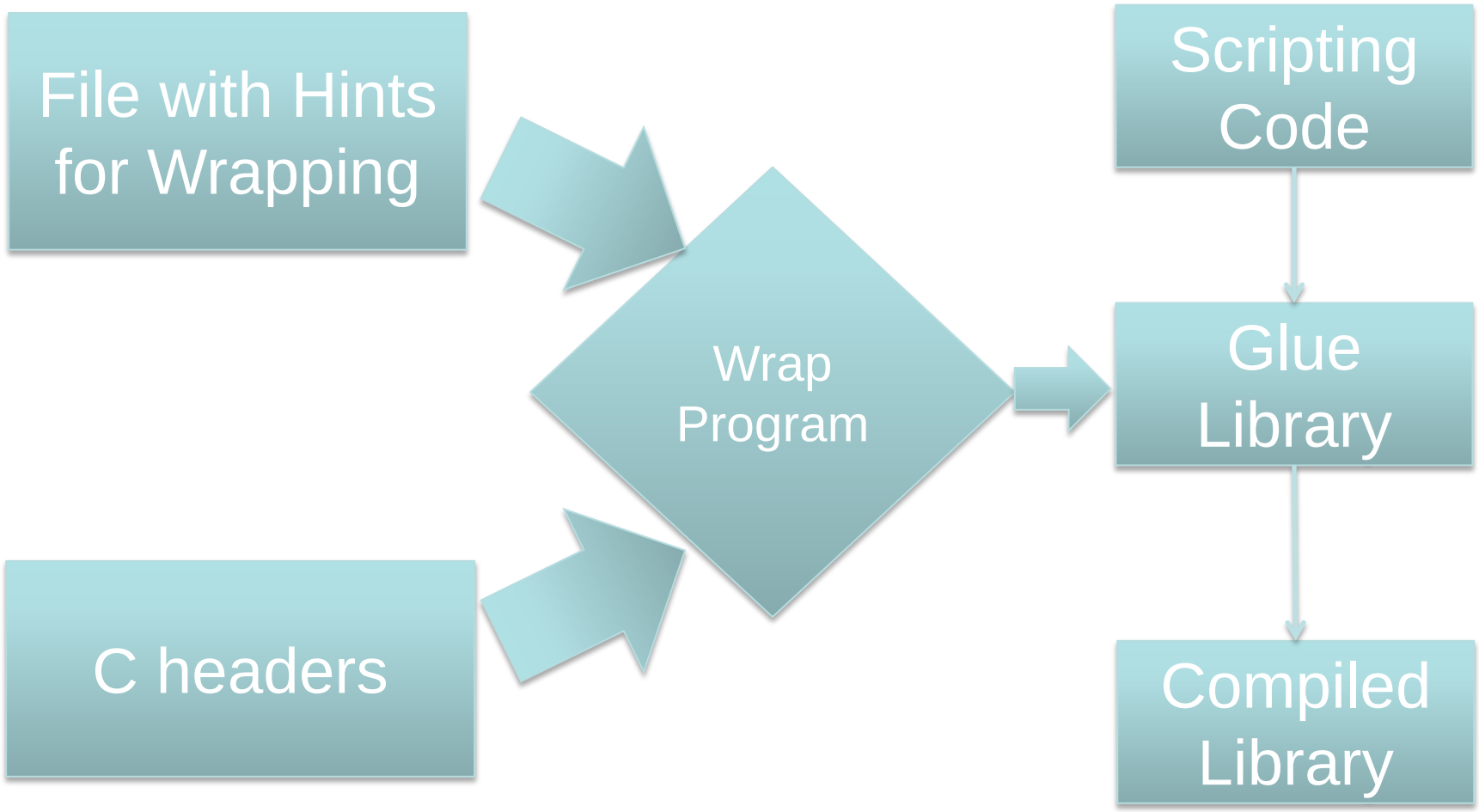
Write final results()

Write summary results()

Top Level

Glue

C or Fortran


File with Hints for Wrapping
C headers
Wrap Program
Scripting Code
Glue Library
Compiled Library

Python Glue - Pyrex

```
from mpi4py.mpi_c cimport MPI_Comm, MPI_COMM_WORLD,
MPI_Comm_split, MPI_Comm_rank

cdef extern from "library.h":
    void lammps_open(int argc, char** argv,
                     MPI_Comm  comm, void** lammps_ptr)
    void lammps_close(void* lammps_ptr)
    void lammps_file(void* lammps_ptr, char* file)
    char* lammps_command(void* lammps_ptr,
                                  char* command)
```

Almost All Fortran

Parameter File

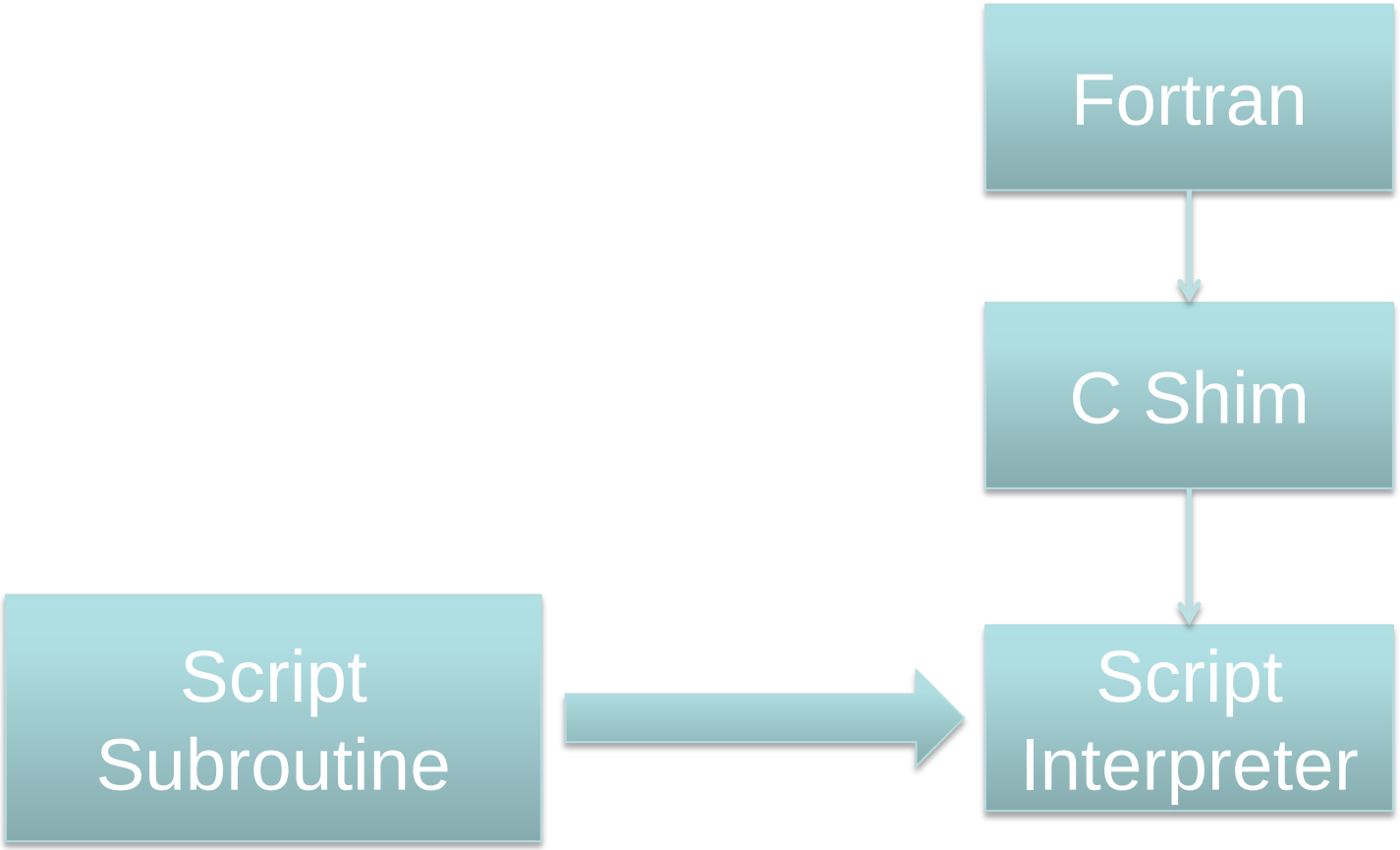
Initialization Subroutine

Occasional Visualization

C or Fortran

Glue

R, Python, Lua



Compiled keeps track of state of interpreter.

Try Exercise with Lua

Heresies

Let's do the whole thing in R, in parallel, on Ranger.

Write almost all of it, even integration, in Python.

Beazley, 1997

- http://www.python.org/workshops/1997-10/proceedings/beazley.html
- You can write inner loops in Python if you…

Always manipulate sets of atoms.

```
# A function written in Python
from SPaSM import *
def run(nsteps, Dt, freq):
    for i in xrange(0, nsteps):
        integrate_adv_coord(Dt)
        boundary_periodic()
        redistribute()
        force_eam()
        integrate_adv_velocity(Dt)
        if (i % freq) == 0) :

output_particles('Dat'+str(i))
```

Two Ways for Pure R

- Leave R untouched, but run it thousands of times.
- Use Rmpi library to make parallel R.

Not fast, but scalable, malleable.

R Labs

- Run Serial R Many Times
- Run an R MPI job

Both easier as a group.

Gratefully Borrowed

http://www.flickr.com/photos/jvuokko/374781736/

http://www.flickr.com/photos/lovenotfear/490469619/