Scripting in Practice

Turn in homework

Bioinformatics in Perl

An involved example

Biochemistry 101

- DNA made of 4 nucleotides
  - adenine, guanine, cytosine, thymine
- Triplets of nucleotides "translate" into 22 different amino acids
  - "gtc" translates to V (Valine)
- Amino acid chains are proteins
- Proteins "fold" into 3d shapes
- A particular protein's shape is important

Perl has strong bioinformatics support

- BioPerl modules
- Books:
  - Beginning Perl for Bioinformatics
  - Mastering Perl for Bioinformatics

DNA is text

- Researchers decided to keep everything as text
- DNA:
  - `gtcgaccacttagtcta`
- Amino Acids:
  - `VDHYIRLRQAGVEKPER`
- Data files are formatted text (example)

DNA is text

- Can parse with module (BioPerl, etc)
- Can write own parser easily
- Can whip together sloppy regular expression

The problem

- Lots of bioinformatics work is correlating existing data
- Data is stored in multiple online databases
- Start with some database IDs, and correlate protein folding information to DNA sequence.

Doing it by hand

- Compare J01609 DNA to 1rx1 protein folds
  - http://www.ebi.ac.uk/cgi-bin/expasyfetch?J01609
  - http://ca.expasy.org/tools/dna.html
  - http://pdb.rcsb.org/pdb/explore/sequenceText.do?structureId=1rx1&chainId=A
- Gluing together other programs (web applications)
  - We can do this

Technique: Web robots

- robots.txt
- Getting pages
  - call out to curl/wget
  - LWP::UserAgent
  - WWW::Mechanize
- Parsing pages
  - Regular expressions
  - HTML::TreeBuilder
  - WWW::Mechanize

Tactics: Write one small part at a time

- Test by hand
- Add debugging information
- Isolate into functions
- Write matching tests

Tactics: Get it done quickly

- die on any error
- Call out to another program instead finding the "right" module

HTML Management

An overview

HTML Management

- Converting XHTML to HTML5
- Converting static pages to a templating system
- Absolutely possible
- Small number of pages? Probably faster to do by hand.

HTML Management: Valid XHTML

- Is the input valid XHTML?
- Then it's valid XML
- High quality XML parsers are a dime a dozen
  - End up with a tree, recursively descend outputting new format as you go
- Skip scripting, use XSLT?

HTML Management: Messy HTML

- Not quite valid XHTML?
- Harder...
  - Real web browsers spend a lot of time handling invalid HTML gracefully.
- HTML::TreeBuilder (and others)
  - End up with a tree, recursively descend outputting new format as you go
- Regular expression abuse
  - More fragile, but easy to tune to your particular data

Product Configurator

An overview

Example

Database support

- Typically store options and choices in a database
  - relational model / SQL dominates
  - A well designed database is fast, scalable, and robust
    - See also: CS 564: Database Management Systems
- Perl's DBI module is excellent
  - Speaks to Oracle, Sybase, ODBI, MySQL, PostgreSQL, SQLite

Web application

- Low level: CGI module
- Medium level: CGI::Application
- High level: Maypole, Catalyst, Perl on Rails

Web 2.0 Sauce

- Highly responsive, "smart" web applications means Javascript
- AJAX - asynchronous *Javascript* and XML

Reading and Writing Data Files

An overview

Reading and Writing Data Files

- Good use for a module
  - Reusable
- Good use for an object
  - Clear organization
- *Does a module already exist?*

Example

Which scripting language do I use?

What is already there?

- Use what you know
- Use what your coworkers know
- Use what you're already using

- eg Condor is mostly Perl
- eg Linux init scripts are Bourne shell (/bin/sh)

Prefer a vibrant community

- Lots of people to help
- Lots of useful modules
- Ongoing debugging and optimization

- Perl, Python, and Ruby all have vibrant communities

I want this library

- Library X might be exactly what you need for your task. X usually only works with particular scripting language.
  - Ruby's Ruby on Rails
  - Perl's Template Toolkit
  - Python's matplotlib
- Of course, for popular libraries, there is probably an equivalent in other languages

I need portability

- Unix-like systems:
  - Bourne Shell (/bin/sh)
  - Perl
  - Python
  - A touch of Ruby

I need speed

- You probably don't
- The details will depend on your problem
  - Is there a fast modules for the slow part?

I'm writing a web application

- Dedicated language: PHP
- Frameworks: Ruby on Rails
- Low level: Perl CGI

I need to embed the language

- Lua
  - Lots of games: Crysis, STALKER, GRAW
- Maybe Javascript
  - Good enough for Flash games
  - Mozilla's spidermonkey
- Maybe Tcl :-p

Feedback

- What worked?
- What to change?
- What to add?
- What to cover?

Evaluations