Functional Big-Data Genomics

Ashish Agarwal, Sebastien Mondet,
Paul Scheid, Aviv Madar, Richard Bonneau, Jane Carlton, Kris Gunsalus

Center for Genomics and Systems Biology
Department of Biology
New York University

Commercial Users of Functional Programming
Copenhagen, Denmark
Sep 15, 2012
2000:  8 YEARS  $3,000,000,000
Now:  10 days  $10,000
Computational Stack

• **Website**
  - data visualization, upload/download

• **Analysis Pipeline**
  - run 3\textsuperscript{rd} party tools, distribute jobs, track results

• **LIMS (Laboratory Information Management System)**
  - metadata tracking of samples, libraries, protocols, etc

• **Systems Infrastructure**
  - operating systems, packages, virtual machines, network
The Whole System

Wet-Lab Tech.

Prof.

Bioinformatician

HiSeq 2000

Library Submission

Local Storage

Transfer

HPC Cluster + Servers

Website

HTTPS

Demultiplexing

Statistics

Alignment

SSH
Big-Data is Not Just Big

- **Volume**
- **Variety**
- **Velocity**
The **Types** of our DSL

- More expressive than SQL
- More closely models the schemas we need
- We have full control over DSL objects

```
type =
| Bool
| Timestamp
| Int
| Real
| String
| Option of type
| Array of type
| Record of (string * type) list
| Enumeration of string list
| Function of type * type
| Volume of volume
```
DSL Compiler

- From a given program in the DSL, we *automatically* generate
  - SQL scripts to initialize and reset the database
  - About ~10k lines (and growing) of OCaml strongly typed database reads/inserts
  - Web widgets
  - Figures in this talk
A Program in our DSL

(record sample
  (name string)
  (organism organism option)
  notable)
Samples, Organisms, and Protocols

- **sample**
  - name: String
  - organism: organism option
  - note: String option

- **protocol**
  - name: String
  - doc: protocol_directory
  - note: String option

- **organism**
  - name: String option
  - informal: String option
  - note: String option

- **protocol_directory**
  - .../protocols/

- **barcode_provider**
  - none
  - bioo
  - bioo_8_beginning
  - illumina
  - nugen
Managing HiSeq Data

**hiseq_raw**
- flowcell_name: String
- read_length_1: Int
- read_length_index: Int option
- read_length_2: Int option
- with_intensities: Bool
- run_date: Timestamp
- host: String
- hiseq_dir_name: String

**transfer_hiseq_raw**
- hiseq_raw: hiseq_raw
- availability: inaccessible_hiseq_raw
- dest: String
- hiseq_raw

**delete_intensities**
- hiseq_raw: hiseq_raw
- availability: inaccessible_hiseq_raw
- hiseq_raw

**dircmp_raw**
- hiseq_raw: hiseq_raw
- availability: inaccessible_hiseq_raw
- hiseq_checksum
Generating SQL
Entire NYU GenCore Data Schema (in our new DSL)
Virtual Filesystem

• Database entries have to map to physical files

• Files in DSL are often recomputable
• But don’t have to exist on physical filesystem, or can exist in compressed form

• File metadata is recorded
  – e.g. FASTQ quality-score encoding = 33 or 64
• Easier file restructuring/renaming
• Easier migration to new clusters
Function Values

• No lambdas
• Only hardcoded function constants
• Output values are “normal” data
• Implementation must account for
  – failures
  – time lags
  – serialization
Website – powered by Ocsigen

GENCORE HOME

Welcome

This is Gencore's website; for library submission information see the FAQ or GenCore's Google-Docs.

Menu

- HiSeq 2000 Runs
- Persons
- Libraries
- Function evaluations
- Layout Navigator
Auto-generated Forms

![Auto-generated Forms Image]

<table>
<thead>
<tr>
<th>g_id: Identifier</th>
<th>3179</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actions:</td>
<td></td>
</tr>
<tr>
<td>- You may add a new person</td>
<td></td>
</tr>
<tr>
<td>- You may modify this person</td>
<td></td>
</tr>
</tbody>
</table>
Data Migrations

val migrator: S-Expression -> S-Expression

$ hitscore dump-to-file backup_v42
$ ./migrator backup_v42 backup_v43
$ hitscore wipe-out-database
$ hitscore init-database
$ hitscore load-file backup_v43
$ hitscore verify-layout

We have done over 25 data migrations!
OCaml Has Many Libraries

- Core & Batteries
- Lwt – lightweight threads
- Ocsigen – web programming framework
- Biocaml
- PG’OCaml
- Xmlm
- Ocamlnet
- ... and many more

Database -> Domain Logic -> Web Interface
Experience with OCaml

• The Good
  – Good libraries
  – Industrial strength
  – Hackable
  – Option to be unsafe feels safe
  – Excellent performance

• Could Be Better
  – Public relations
  – Build system
  – Blessed libraries
  – More libraries
Functional Programming in Biology

• “Functional programming” is becoming a recognized term
• Programmers are desperately needed
• Be sure to distinguish software engineering from data analysis
• Key to success:
  – acquire domain expertise
  – build software fast
• Discuss programming scientifically
Conclusions

• The Genomics Sequencing Core at NYU CGSB runs on OCaml
• Entire system built by ~1.3 programmers
• First version: in production within 2 months
• Biology needs you!