Codefest 2014 Project Ideas

This shared document provides project ideas for OpenBio Codefest 2014 (http://www.open-bio.org/wiki/Codefest_2014) in Cambridge, MA on July 9th and 10th.

Please brainstorm ideas and directions that we'll use to organize development for the two days. Add new ideas into existing categories or start your own:

Provenance

- Improve reproducibility and provenance of bioinformatics pipelines. Explore integration with Arvados.
- Store W3C PROV data in the header of a BAM file (idea from Galaxy hackathon)

Installation/build/data management

- Finalize CloudBioLinux with upcoming Ubuntu 14.04 LTS release (due in mid-April). Ensure availability in multiple locations beyond us-east.
- Integrate NeuroDebian and CloudBiolinux, possibly start analyzing http://studyforrest.org using that deployed infrastructure (email roman@incf.org for more info).
- Create GNU Guix packages and add GNU Guix to Cloudbiolinux (as an option) - GNU Guix is very useful for cross-platform predictable software deployments (Pjotr Prins)
- Start a Cloud BioLinux repository at https://registry.hub.docker.com. Commit some ssh-able docker containers that contain some example tools or pipelines from Cloud BioLinux. ←-- Who is working on that? interesting, I would like to contact you!

Visualization

- A Drupal module that can integrate seamlessly with BioJS components, and that can be used as a way to install BioJS on sites that use Drupal.
- Better handling of read-level data in Biodalliance (and maybe alternative back ends in addition to direct BAM access -- *e.g.* GA4GH APIs?)
- Exploring d3.js and NVD3 for biological data visualization, using Galaxy circster as an example (http://www.biomedcentral.com/1471-2164/14/397).

Galaxy/Bio*

- Wrap relevant software for Galaxy and/or bcbio-nextgen.
    - Finish Galaxy's multiple datasets returned from generic datasource tool started by Matt Shirley @ Galaxy's Hackathon (https://trello.com/c/YIADVbkI)
    - Create a functioning Galaxy Tool for CNVkit

- Hack Galaxy to output citations for all the tools in a workflow. (John Chilton, Michael R. Crusoe)
- Design data structures to represent non-topological alignments of protein structures in BioJava (Spencer Bliven)
- Improve Biocaml's swiss-army knife app
- Extend Galaxy API and BioBlend library to provide ID/name of the tools that are currently running in a given history
- Modify Galaxy tool wrappers to work with dataset collections (FASTQ Parallel Groomer, Bowtie, SAM-to-BAM, MACS2 callpeak/bdgcmp, BedGraph-to-bigWig, etc).
- Updated trimmomatic in Galaxy and attempted to support dataset collections. (Simon Gladman, Michael Crusoe.)

ADAM/Data representation

- Coalesce similar data models and services from Picard/HTSJDK, ADAM, Google Genomics, GA4GH, O|B|F libraries
- RNA-seq processing with ADAM (Timothy Danford, Carl Yeksigian)

Other

- A tool (first, unfinished attempt) to import videos from openly licensed scholarly articles into YouTube, building on the existing tool that imports them into Wikimedia Commons.
- OpenPGP-GnuPG key signing, e.g. http://www.cryptnet.net/fdp/crypto/keysigning_party/en/keysigning_party.html
  To participate, create a PGP key pair if you haven't already and send the public key to a keyserver. Then bring small pieces of paper with your name and PGP key fingerprint on them to hand out to people (probably best for day 2 of the hackathon)
  After the session, you may fingerprint, validate, and sign the keys of those people you met at the session.
- CAcert WoT signing party if you want too:
  - http://www.cacert.org/
  - Also, I have invites for keybase.io too.
- Genome assembly - Incorporation of kmergenie into VelvetOptimiser to automatically pick start and end kmer sizes for searching. (Simon Gladman)

Tasks and accomplishments

Guillermo Carrasco, Science For Life Laboratory, Stockholm

My purpose the first day was to understand and learn more about the Arvados platform. I was talking with the people of the Curoverse team and they answered all my doubts and questions. I was going through the tutorial and reporting them the points I did not really understand, giving them this way some user experience feedback. I also submitted a minimal patch that solves a bug I found when debugging Crunch scripts: https://github.com/curoverse/arvados/pull/6

During the second day I have been working on trying to integrate the deployment and testing of bcbio-nextgen in Travis-CI. It is not working yet, but getting closer: https://travis-ci.org/guillermo-carrasco/bcbio-nextgen

Actually after several tries, seeing that it crashes after approx. 15 mins of installation (and has not finished yet), probably a reasonable conclusion is that bcbio-nextgen is too heavy for Travis-CI. Having CI for bcbio-nextgen should be done with another CI System like Jenkins, where you can have pre-configured environments (i.e “fresh” VM with the pipeline dependencies already installed).

ADAM Team
Frank Nothaft (Berkeley), Timothy Danford and Carl Yeksigian (Broad Institute), Anita Stout and Dennis Cunningham (Novartis Institutes for Biomedical Research)
Our goal was to jumpstart the development of an RNA-seq processing pipeline in Spark and ADAM. To keep our efforts coordinated, we targeted a reimplementation in ADAM of the “deFuse” method for fusion transcript discovery from paired-end NGS data (http://www.ploscompbiol.org/article/info%3Adoi%2F10.1371%2Fjournal.pcbi.1001138). This was a stretch goal, but working as a team we took significant steps towards a reimplementation based solely on the paper and its supplementary methods: a greedy set-cover implementation in Spark and a clique-finding method in GraphX, along with outlines of AdaBoost, coordinate liftover utilities, and a split-read alignment method using dynamic programming were all completed over the two day period (https://github.com/bigdatagenomics/RNAdam). We also spent some of our time merging the outstanding pull requests in the ADAM (https://github.com/bigdatagenomics/adam) and bdg-formats (https://github.com/bigdatagenomics/bdg-formats) repositories.

Lorena Pantano and Rory Kirchner, HSPH, Boston
My first days was to integrate a specific tools to detect contaminant in reads (kraken) into bcbio-nextgen. I did first an easy integration of the tools as part of the quality summary developed by Rory Kirchner. After the tool was running, I dedicated the second day to add some simple statistics into the project.yaml file, that is the output of the pipeline, so now you have an easy way to detect if your samples have reads mapped to any other species. The second day, we started to created a minimal database for kraken including only one chromosome of human, mouse and zebrafish, beside all bacteria and virus annotated in genbank. The next step will be to check if this minimal database is enough to detect contaminants from another species.

Biocaml
- Resume work on the command line application.
- Start a tutorial/demo on Biocaml with async_parallel and start setting-up benchmarks

Biopython Team (Peter Cock, Eric Talevich, Wibowo Arindrarto [bow])
We started brainstorming for several things to work on in the beginning of the first day. After two days, we accomplished the following:

- Triage & resolve existing bug reports (Bow, Peter, Eric)
- ABI parser updates (Peter, Bow)
- Updating SAM/BAM branch (Peter)
- Automated testing (& fixes) of examples in Bio.Phylo documentation (Eric)
- Port more API docs to reStructuredText for eventual Sphinx compatibility (Bow)
- Experiment with splitting the Biopython distribution into separately installable Python packages (Bow, proof of concept implementations at http://github.com/bow/poc_biopy and available in PyPI via pip)