Using Galaxy to provide a NGS Analysis Platform

GTC’s NGS & Bioinformatics Summit Europe
October 7-8, 2013 in Berlin, Germany.

(public version)

Hans-Rudolf Hotz (hrh@fmi.ch)
Friedrich Miescher Institute for Biomedical Research
Basel, Switzerland
Friedrich Miescher Institute
- funded by the Novartis Research Foundation
- affiliated institute of Basel University

311 employees
(incl. 90 PhD students, 94 Post Docs)

Epigenetics
(7 research groups)

Cancer
(7 research groups)

Neurobiology
(8 research groups)

Technology Platforms
Computational Biology – Cell Sorting – Imaging and Microscopy – C. elegans
Functional Genomics – Histology – Mass Spectrometry – Protein Structure

- member of Swiss Institute of Bioinformatics
Analyzing NGS data in a Bioinformatics Core Facility is fascinating because

- scientists keep coming up with new kind experiments

- new algorithms to deal with NGS data are developed continuously

- there is a new (improved) sequencing instrument on the market every few months
Analyzing NGS data in a Bioinformatics Core Facility is **difficult** because

*people with different background/training are interested in using NGS*

the “average” lab scientist is looking for the red button to press

bizarre output from the sequencer ➔ publication in *Nature*

the “average” statistician is creating wonderful blots.....

...nobody understands
and the Bioinformatics Helpdesk is caught in the middle....

- different and new instruments
- the “average” statistician
- the “average” lab scientist
- different experiments
- new algorithms
- limited IT-resources
.....looking for a solution

limited IT-resources can be solved *(with money)*

new instruments follow the "literature" and
ew algorithms test the new open source
different experiments tools yourself

the "average" lab scientist learn R/Bioconductor
the "average" statistician flexible environment
turn a command line tool *(R script)* into a ‘red button’

*the ‘red button’ should be as similar as possible to the command line process*
the solution:

http://galaxyproject.org

a flexible environment which allows you to turn command line tools into ‘red buttons’
“Galaxy is an open, web-based platform for data intensive biomedical research. Whether on the free public server or your own instance, you can perform, reproduce, and share complete analyses.”

The Galaxy Team is a part of BX at Penn State, and the Biology and Mathematics and Computer Science departments at Emory University.

The Galaxy Project is supported in part by NSF, NHGRI, The Huck Institutes of the Life Sciences, The Institute for CyberScience at Penn State, and Emory University.
http://galaxyproject.org

....and I am NOT part of the Galaxy Team!

I am just a member of the worldwide community of many Galaxy users, adopters, developers, evangelists, etc.
what does Galaxy?

- provides a GUI (i.e. the ‘red button’) to (Bioinformatics) command line tools in your web browser
- manages/stores your (raw) data and results
- allows you to create workflows
- allows sharing and reproducing your analysis
Use Galaxy

http://wiki.galaxyproject.org/BigPicture/Choices

public and free web service: http://usegalaxy.org/

deploy your own Galaxy server:

local server

cloud (Amazon Machine Images)

Galaxy appliance (offered by BioTeam)
why are we using Galaxy

- open source software
- it provides a standard set of tools
- we can add our own scripts and tools
  
  turn open source tools into a ‘red button’

- the Galaxy community is huge and the software is established (first publication in 2005)

- a local installation is simple to set up

- it is flexible (you can adjust it to your needs)

in use at the FMI since 2007
it is really simple to install

requirements:
- a Mac or linux PC with Python and Mercurial

just 3 commands:
- hg clone https://bitbucket.org/galaxy/galaxy-dist/
- cd galaxy-dist
- sh run.sh

...and it is ready in your web browser at:

http://localhost:8080
Hello world! It’s running...

To customize this page edit static/welcome.html.

This project is supported in part by NSF, NH.
how does it work

Upload File (version 1.1.3)

File Format:
- Auto-detect

Which format? See help below

File:

Browse...

TIP: Due to browser limitations, uploading files larger than 2GB is guarded by the method (below) or FTP (if enabled by the site administrator).

URL/Text:

foo 1 2

Here you may specify a list of URLs (one per line) or paste the contents.

Convert spaces to tabs:
- Yes

Use this option if you are entering intervals by hand.

Genome:
- Click to Search or Select

Execute

History

8 bytes

1: Pasted Entry
1 region
format: interval, database: ?
Info: uploaded interval file
view in GeneTrack

foo 1 2

FMI
Friedrich Miescher Institute
for Biomedical Research
how does it work
Galaxy out of the box

input tools:
- text box / upload file / url
- access to UCSC table browser and ensembl biomart

tools for file conversion and text manipulation

tools for table calculation, basic set-theory and operation on genomic intervals
adding more tools

Galaxy Tool Shed

http://wiki.galaxyproject.org/Tool%20Shed

enables sharing of Galaxy tools across the Galaxy community

more than 1000 tools available

handles 3rd party dependencies

your own tool

ideally submitted to the Tool Shed
As long as you can run it on the command line, you can incorporate it into Galaxy.

- add the executable or script (R, perl, python, bash, etc)
- write a tool definition file
- add it to the list of tools
command line to ‘red button’

bash-3.2$ ls
bed_to_gff_converter.py  foo.bed
bash-3.2

bash-3.2$ cat foo.bed
foo 1 2
bash-3.2$

bash-3.2$ ./bed_to_gff_converter.py foo.bed foo.gff
1 lines converted to GFF version 2.
bash-3.2$

bash-3.2$ ls
bed_to_gff_converter.py  foo.bed  foo.gff
bash-3.2$ cat foo.gff
##gff-version 2
##bed_to_gff_converter.py
foo  bed2gff  region_0    2   2   0   +   .   region_0;
bash-3.2$
This tool converts data from BED format to GFF format.

command line to ‘red button’

no need to define/design a GUI!
sort of a ‘red button’
a few more highlights

re-run a tool with the same parameters (or different parameters)
display in the built in or a web genome browser
what happened to the poor guy sitting at the Bioinformatics Helpdesk....

- different and new instruments
- the "average" statistician
- different experiments
- new algorithms
- limited IT-resources
- the "average" lab scientist
- Bioinformatics Helpdesk
He is a Galaxy administrator

...and he adds the R script to Galaxy

the lab scientist can repeat the analysis in the ‘friendly’ web browser and not on the ‘scary’ command line
He is a Galaxy administrator

different and new instruments

new algorithms

different experiments

...and he adds the new tools to Galaxy

and everybody can test them

without any delay
He is a Galaxy administrator

- no need to buy extra hardware
- Galaxy provides tools to track and report jobs (errors are flagged)
- Galaxy provides tools to limit disk space
- Galaxy allows you to share data
NGS analysis at the FMI (with the old pipeline)

pre-processing → import

alignment
to (multiple) genomes
to (multiple) annotation DBs

deepseq repository

extract
counts
wig files
QC reports
The old NGS pipeline has been
....just a bunch of Perl scripts
....just a simple file system

The new NGS pipeline is

a Bioconductor package: “QuasR”
(Quantification and Analysis of Short Reads)

the new NGS pipeline is

new Bioconductor package: **QuasR**  
(Quantification and Analysis of Short Reads)

- package that provides an end-to-end analysis solution for tag counting applications
- ships with the aligners Bowtie and SpliceMap
- creates alignments from within R
- provides a set of simple to use functions to create a large variety of count-tables
- provides an additional layer of abstraction on top of pre-existing tools in Bioconductor
QuasR Bioconductor Package

qAlign
- raw data
  - sequence reads

experiment type
- paired, bisulfite, conditions, replicates

analysis type
- genome, auxiliaries, spliced, SNPs

Storage

qProject
- sequences
  - ACGTACGTAACG
  - GTAGCTAGCGAATT
  - CTAACGAAATT
  - CGTAGCTAGCGAATT
  - CTAGCGAATT

parameters
- aligner, ...

alignments
- ACGTACGTAACG
  - GTAGCTAGCGAATT
  - CTAACGAAATT
  - CGTAGCTAGCGAATT
  - CTAGCGAATT

qQCReport
- quality control

qCount
- query regions
- counts
  - gene1 323 318
  - gene2 25 17
  - gene3 211 187
  - ...

qExportWig
- wiggle file

qMeth
- query regions
- methylation
  - CpG_1 34 32
  - CpG_2 28 27
  - CpG_3 113 26
  - ...
  - ...
  - ...

Storage
it is really simple with QuasR

```r
sampleFile <- "data/samples_chip.txt"
genomeFile <- "genomes/hg19.fa"

proj <- qAlign(sampleFile, genome=genomeFile)
qExportWig(proj, binsize=10)
```

but still to scary / complicated

how can Galaxy help?
a general NGS workflow

align reads

\[\rightarrow\] BAM file

extract data

\[\rightarrow\] WIG file

data is hidden in Galaxy
NGS analysis at the FMI

- pre-processing
- import
- deepseq repository
- alignment to (multiple) genomes
to (multiple) annotation DBs

data is inside "Galaxy"

data is outside "Galaxy"
storing data outside of Galaxy

- the Galaxy ‘aligner’ stores the BAM file in the central NGS repository and creates just a log file for Galaxy
- the Galaxy ‘extract’ tool knows the location of the NGS repository

allows to share with non-Galaxy users
successfully finished annotation of test_20130820 to ce6-ceV01-aln2

and now the ‘command line geek’ can make a BED file

[geek@xenon1 ~]$ extractData.pl -f -s p -m 100 -i test_20130820 ce6-ceV01-aln2 genome | frag2bed.pl -t -q -U - | head -5

track name='test_20130820'
chr1  10493   10528   sq39319  1       +
chr1  10736   10764   sq74484  1       +
chr1  11442   11477   sq1340   1       +
chr1  13799   13834   sq84955  1       +

[geek@xenon1 ~]$
allows to share with non-Galaxy users

command line

```
extractData.pl -f -s p -m 100 -i test_20130820 ce6-ceV01-aln2 genome |
frag2bed.pl -t -q -U - > test_20130820-ce6-ceV01-aln2.bed
```

Galaxy tool definition file

```
extractData.pl -f $strand $maxhits $ignCnts $sampleSelect.sampleId $genome-$annot-aln2 genome |
frag2bed.pl -t -q $summary.ucsc - > $output
```
and doing the same in Galaxy
Galaxy will give you a platform where you can offer your local NGS pipeline with a graphical user interface without compromising the freedom of the command line.
Acknowledgment

Michael Stadler    Christian Hundsrucker
Anita Lerch       Tim Roloff       Lukas Burger
Dimos Gaidatzis   Stefan Grzybek

....and all the people from the “Galaxy”

http://galaxyproject.org