DNAnexus

DNAnexus Development Environment
Bioinformaticists & Developers

Today Agenda

- Introduction dx-toolkit
- Supported languages - bash, python, docker
- Supported resources
- App/Applet building experience
  Peter FitzGerald (bash)
  Carl McIntosh (Python)
  Skyler Kuhn (Docker)
- Open Discussion

DNAnexus provides a simplified, structured and managed access to Amazon Web Services (AWS) and Microsofts (Azure)

DNAnexus Applet System Requirements

Default: mem1_ssd1_x4

Common AWS instance types:

Name	Memory_GB	Storage_GB	CPU_Cores
mem1_ssd1_x2	3.8	32	2
mem1_ssd1_x4	7.5	80	4
mem1_ssd1_x8	15.0	160	8
mem1_ssd1_x16	30.0	320	16
mem1_ssd1_x32	60.0	640	32
mem2_ssd1_x2	7.5	32	2
mem2_ssd1_x4	15.0	80	4
mem2_ssd1_x8	30.0	160	8
mem3_ssd1_x2	15.0	32	2
mem3_ssd1_x4	30.5	80	4
mem3_ssd1_x8	61.0	160	8
mem3_ssd1_x16	122.0	320	16
mem3_ssd1_x32	244.0	640	32
mem1_ssd2_x2	3.8	160	2
mem1_ssd2_x4	7.5	320	4
mem1_ssd2_x8	15	640	8
mem1_ssd2_x16	30	1280	16
mem1_ssd2_x36	60	2880	36

- **Memory:**
  - AWS: 3.8 - 244 GB
  - Azure: 3.9 - 448 GB
- **Storage:**
  - AWS: 32 - 2,880 GB
  - Azure: 32 - 1,024 GB
- **Harddrive:**
  - Standard Drive
  - Solid-State Drive
- **Number of Cores:**
  - AWS: 2-36
  - Azure: 2-32

Common Azure instance types:

Name	Memory_GB	Storage_GB	CPU_Cores
azure:mem1_ssd1_x2	3.9	32	2
azure:mem1_ssd1_x4	7.8	64	4
azure:mem1_ssd1_x8	15.7	128	8
azure:mem1_ssd1_x16	31.4	256	16
azure:mem2_ssd1_x1	3.5	128	1
azure:mem2_ssd1_x2	7.0	128	2
azure:mem2_ssd1_x4	14.0	128	4
azure:mem2_ssd1_x8	28.0	256	8
azure:mem2_ssd1_x16	56.0	512	16
azure:mem3_ssd1_x2	14.0	128	2
azure:mem3_ssd1_x4	28.0	128	4
azure:mem3_ssd1_x8	56.0	256	8
azure:mem3_ssd1_x16	112.0	512	16
azure:mem3_ssd1_x20	140.0	640	20
azure:mem4_ssd1_x2	28.0	128	2
azure:mem4_ssd1_x4	56.0	128	4
azure:mem4_ssd1_x8	112.0	256	8
azure:mem4_ssd1_x16	224	512	16
azure:mem4_ssd1_x32	448	1024	32

https://wiki.dnanexus.com/API-Specification-v1.0.0/Instance-Types

Command-Line

DX-toolkit

The DNAnexus SDK (dx-toolkit) helps users utilize the DNAnexus platform to its full potential. It provides command-line tools to run Apps/applets from a remote command-line/ script. Additionally, it provides the environment for App/ applet development

- The dx-toolkit is Installed on Helix/Biowulf in the module system and can be run with the following command (Note: *can only be run on bioiwulf interactive nodes*)
  - module load DNAnexus

Development Environment

The main controlling module of an App/applet can be written in either of the following”

- Bash
- Python
- Docker

Helpful Web Pages

- https://wiki.dnanexus.com/Command-Line-Client/Quickstart
- https://wiki.dnanexus.com/Developer-Portal

DNAnexus External Resources

- Packgage Mangers
  - Advanced Packaging Tool (APT)
    - Libraries, Samtools, Bedtools, etc.
    - https://wiki.dnanexus.com/List-of-packages-available-in-the-Execution-Environment
  - Python Package Index (PyPI)
  - Comprehensive Perl Archive Network (Perl)
  - Ruby Gems (gem)
  - Comprehensive R Archive Network (CRAN)

DNAnexus How-to https://wiki.dnanexus.com/Execution-Environment-Reference

Applet Design Process

- Sketch out Workflow (*OMNIGraffle for example*)
- Required Resources (Asset Bundle vs Applet Resource) -> Detailed web link
- Applet Model (standard, parallelize, SPG) -> Detailed web link
- Input Elements -> Detailed web link
- Output
  - Properties and Tags
  - Directory Structure
- Final Touches
  - Documentation (MacDown) -> Detailed web link
  - Script.sh
  - Versions

DNAnexus dx-app-wizard

- From *Terminal* on Biowulf/Helix
  - module load DNAnexus
  - dx-app-wizard
- From *Terminal* on Local Computer
  - Download and Install DNAnexus Platform SDK https://wiki.dnanexus.com/downloads
  - dx-app-wizard

dx-app-wizard Applet Parameters

- Timeout policy[48h] (m | h | d)
- Applet Programming language: (Python | bash), but supports other languages
- Applet Access to Internet [N]:
- Applet Access to Parent Project: [N]
- Ubuntu 14.04
- Compute Nodes

dx-app-wizard Applet Parameters

Input Specification

You will now be prompted for each input parameter to your app. Each parameter should have a **unique name** that uses only the **underscore** "_" and **alphanumeric** characters, and **does not start with a number**.

1st input name (<ENTER> to finish): parameter

Label (optional human-readable name) []:

Your input parameter must be of one of the following classes:

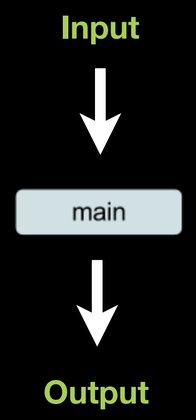
```
applet          array:file     array:record   file       int
array:applet    array:float    array:string   float      record
array:boolean   array:int      boolean        hash       string
```

Output Specification

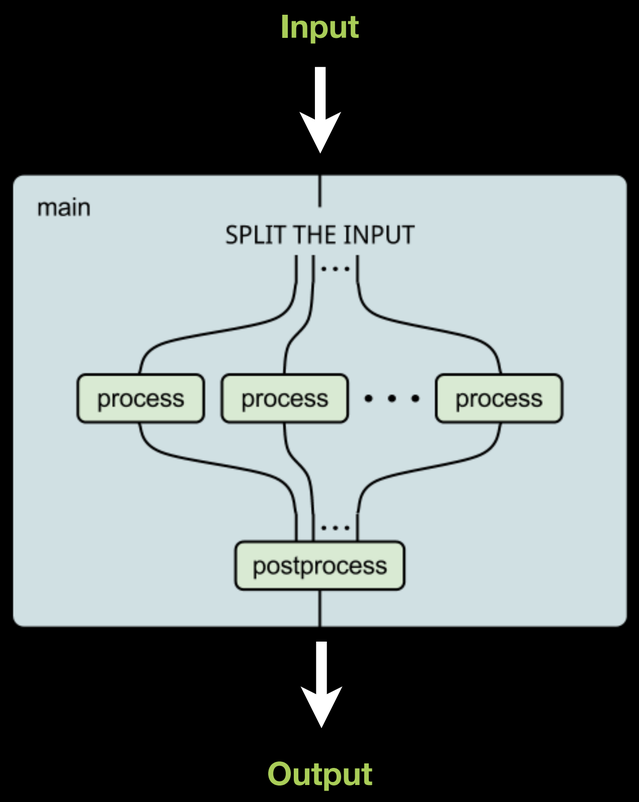
The same

DNAnexus dx-app-wizard

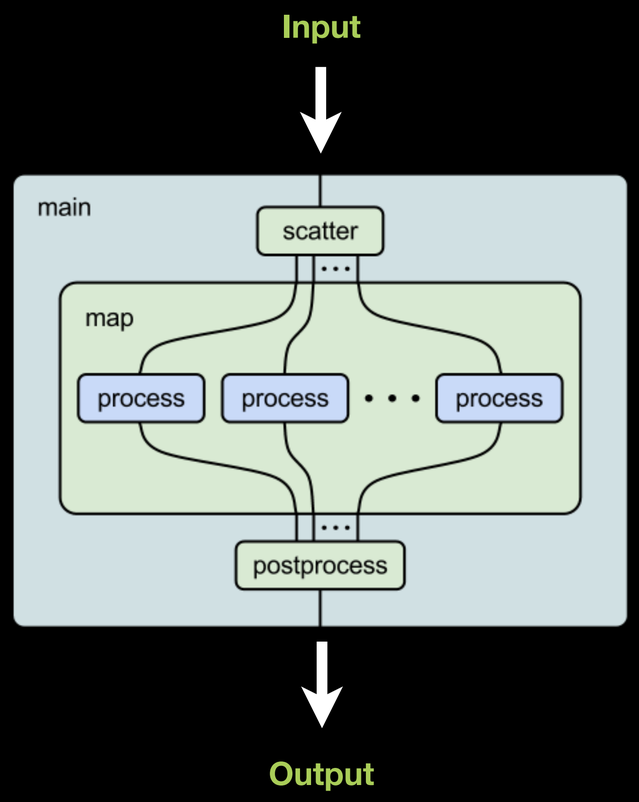
Basic



Parallelized



Scatter-Process-Gather



dx-app-wizard --template (basic | parallelized | scatter-process-gather)

Figures and a tutorial at DNAnexus: https://wiki.dnanexus.com/Developer-Tutorials/Parallelize-Your-App

Cloud Workstation

DNAnexus features a Cloud Workstation App that sets up a cloud workstation as an interactive computer node.

What are typical use cases for this app?
This app can be used as a workstation inside of the DNAnexus cloud platform. By running the app with --ssh or --allow-ssh, users can login to a machine inside of the DNAnexus cloud platform. From there, users can upload/download data to/from the project in which the app is run, perform data analysis, and install additional packages from sources such as apt, cran, pip, github, etc.

It's good for debugging, since you can do so interactively on the node as its running.

```
dx run app-cloud_workstation --ssh

unset DX_WORKSPACE_ID
dx cd $DX_PROJECT_CONTEXT_ID:

dx download file.txt ## download to workstation from parent project
dx upload  file.txt  ## upload to parent project from workstaion

dx terminate $DX_JOB_ID #terminate the session
```

- https://wiki.dnanexus.com/Developer-Tutorials/Cloud-Workstations

Genome Analysis Unit (GAU) DNAnexus Applet Development

Custom Work Flows developed by
Carl McIntosh and Peter FitzGerald (GAU)

ADAP

Using DNAnexus to make the ADAP program readily available to a naive audience. The program was originally written, many years ago, and has had several interface iterations (Web App, Standalone Mac/PC program). The program takes a DNA fasta file and recodes the sequence using alternate AA codons, to generate a new sequence **A**s **D**ifferent **A**s **P**ossible from the original, yet codes the same protein. This approach is useful in the over expression of proteins.

A Collaboration with Christopher Buck & Diana Pastrana (Laboratory of Cellular Oncology, NCI/CCR)

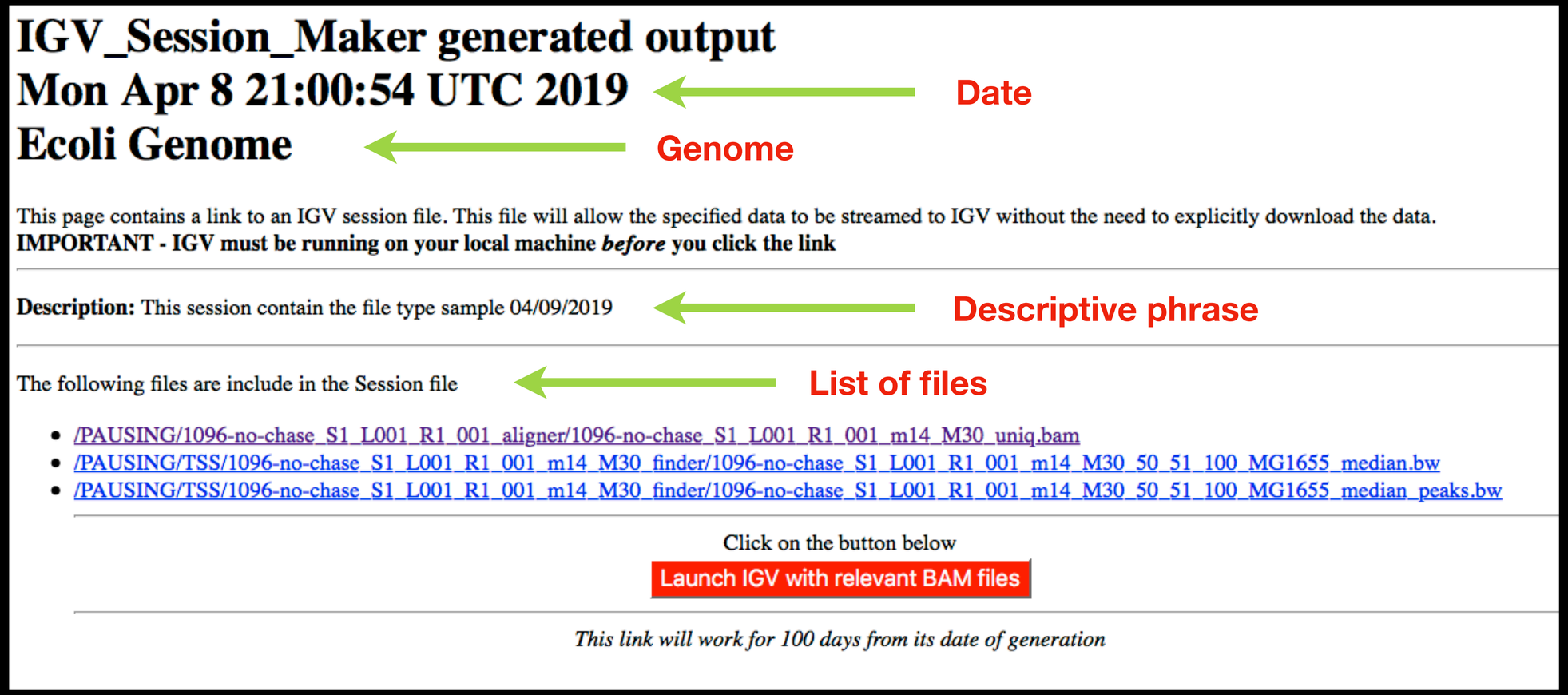
The simplest of applets - simple bash script (5 lines!), and a single binary from C code compiled on biowulf

```
#!/bin/bash
# The following line causes bash to exit at any point if there is any error
# and to output each line as it is executed -- useful for debugging
set -e -x

# Inputs
dx download "$input" -o input.fasta —no-progress
# make a directory for the output
mkdir -p out/results

# Processing
diana -f input.fasta -c /usr/lib/diana.codes > out/results/adap.log

# Outputs - upload the out/results directory

dx-upload-all-outputs
```

IGV Session Maker

- Designed to be a helper applet that allows easy visualization of large files (bam,vcf,big-wig) by a locally running copy of IGV, without the need to download the entire files. It can be run standalone or incorporated into an workflow. By launching from a custom built HTML page it provides a stable record of what is represented in the view.

- Applet consistis of a singe bash script, and used dx commands and variables

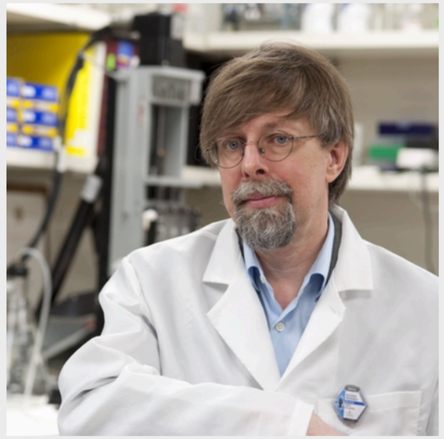


Pausing Peak Tools

NIH Intramural Research Program
Our Research Changes Lives

ABOUT US | OUR RESEARCH | NIH CLINICAL CENTER | NEWS & EVENTS | CAREERS | RESEARCH TRAINING

Search Principal Investigators



Mikhail Kashlev, Ph.D.

Senior Investigator

RNA Biology Laboratory

NCI/CCR

VIEW SITE

Building 560, Room 11-85A
Frederick, MD 21702-1201

301-846-1798

kashlevm@mail.nih.gov

Pausing Peak Tool

Biological system: Various microbial organisms
Primary goal was to identify RNA pausing sites, from netSeq data, and correlate with:

- genome postion
- gene expression
- transcription start sites (TSS)
- specific sequence motifs
- protected read length

Additionally, we needed the ability to compare the effect of different gene deletions.

Pausing Peak Tool

Pausing_Peak_Aligner

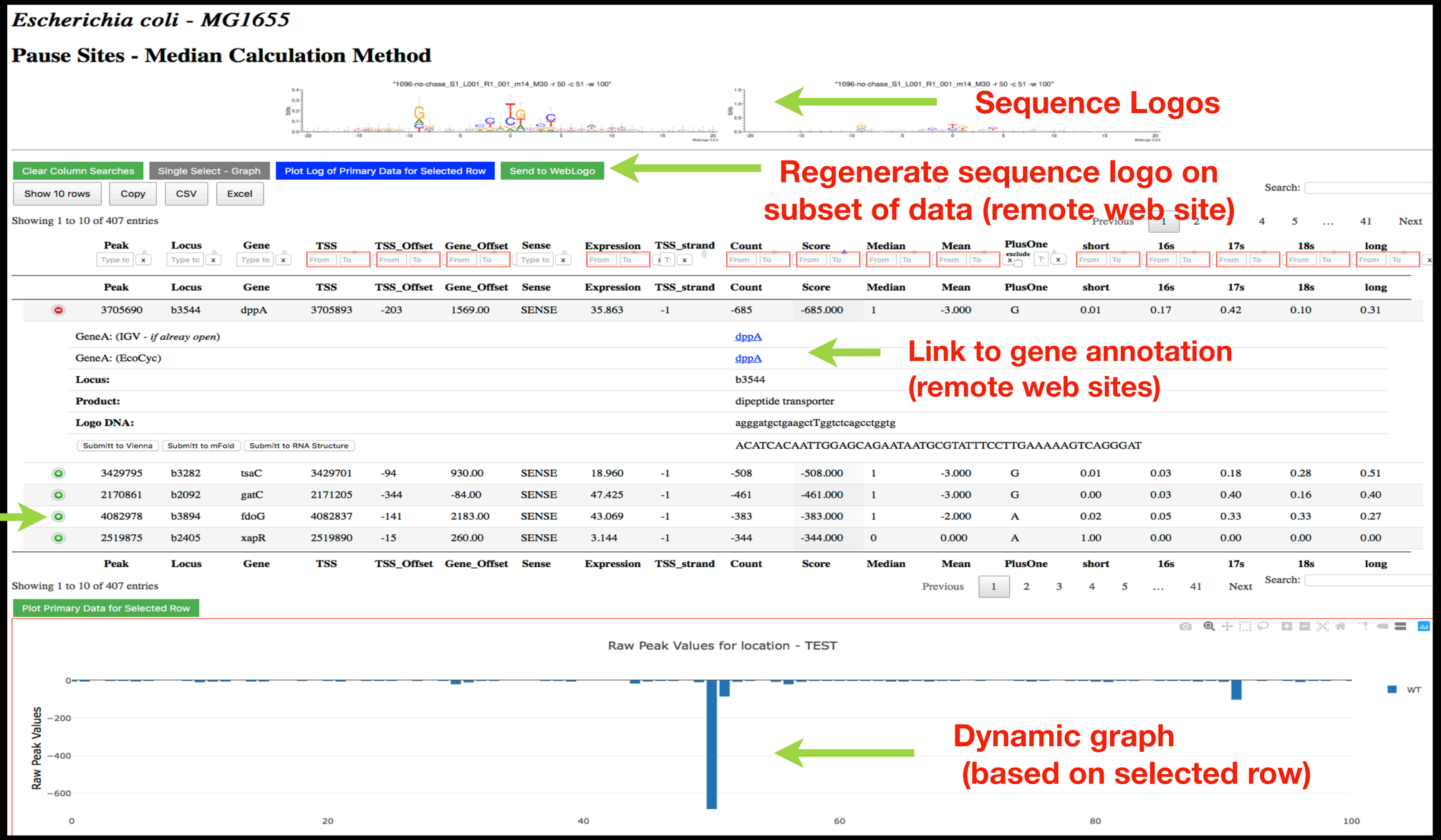
- Remove sequencing primers/adapters using cutadapt (java)
- Use molecular bar code to identify and remove duplicate molecules with BBmap (bin)
- Remove molecular bar code with cutadapt (java)
- Align vs genome with bowtie (bin)
- Get gene expression read count with Salmon (bin)

Pausing Peak Tool

Pausing_Peak_Finder

- Identify pause peaks from bam file modified samtools (bin)
- Generate big-wig files for location of 3’ ends of reads
- Annotate peaks with info relative to genes or TSS from sqlite DB
- Generate Interactive web pages using DateTable and Plotly (javascript)

Pausing Peak Tool



Copy/export selected subsets of data

Data filtering and sorting
- Text
- Value Ranges

Expanding Annotation

Sequence Logos

Regenerate sequence logo on subset of data (remote web site)

Link to gene annotation (remote web sites)

Dynamic graph (based on selected row)

Using tags to surface information about files and processes (*.bam)

CREATED BY	Peter Fitzgerald (by running pausing_peak_aligner_m3 in the job Pausing Peak Aligner Method 3)
CREATED	Apr 9, 2019 3:35 PM
MODIFIED	Apr 9, 2019 3:35 PM
TAGS	Escherichia x coli x NC_000913.2 x

PROPERTIES

Build	NC_000913.2
Genus	Escherichia
Species	coli
stats01 Aligner	Bowtie 1.2.2
stats02 Reads in this bam	3375586
stats03 FASTQ Reads Input	3585806
stats04 Cutadapt Reads Output	3528248
stats05 Clumpify Reads Output	3398154
stats06 Reads Input	3398154
stats07 Reads Unmapped	22568 (0.66%)
stats09 Mapped	3375586 (99.34%)

Tumor Mutational Burden

NATIONAL CANCER INSTITUTE
Center for Cancer Research

CCR Central | Login

Search... Search

CLINICAL TRIALS | RESEARCH | TRAINING | CAREERS | NEWS | ABOUT CCR

Home » Thoracic Surgery Branch » Haobin Chen, M.D., Ph.D.

Haobin Chen, M.D., Ph.D.

Assistant Clinical Investigator
Thoracic Surgery Branch

Dr. Chen's research focuses on developing novel epigenetic therapies for small cell lung cancer. He is board certified in internal medicine and board certified in medical oncology.

Areas of Expertise
1) lung cancer 2) epigenetics 3) molecular biology

CONTACT INFO

Haobin Chen, M.D., Ph.D.
Center for Cancer Research
National Cancer Institute
Building 10-CRC, Room 3-5848
Bethesda, MD 20892
Ph: 240-760-6177
haobin.chen@nih.gov

PERMALINK

Tumor Mutational Burden

Cancer Cell
Article

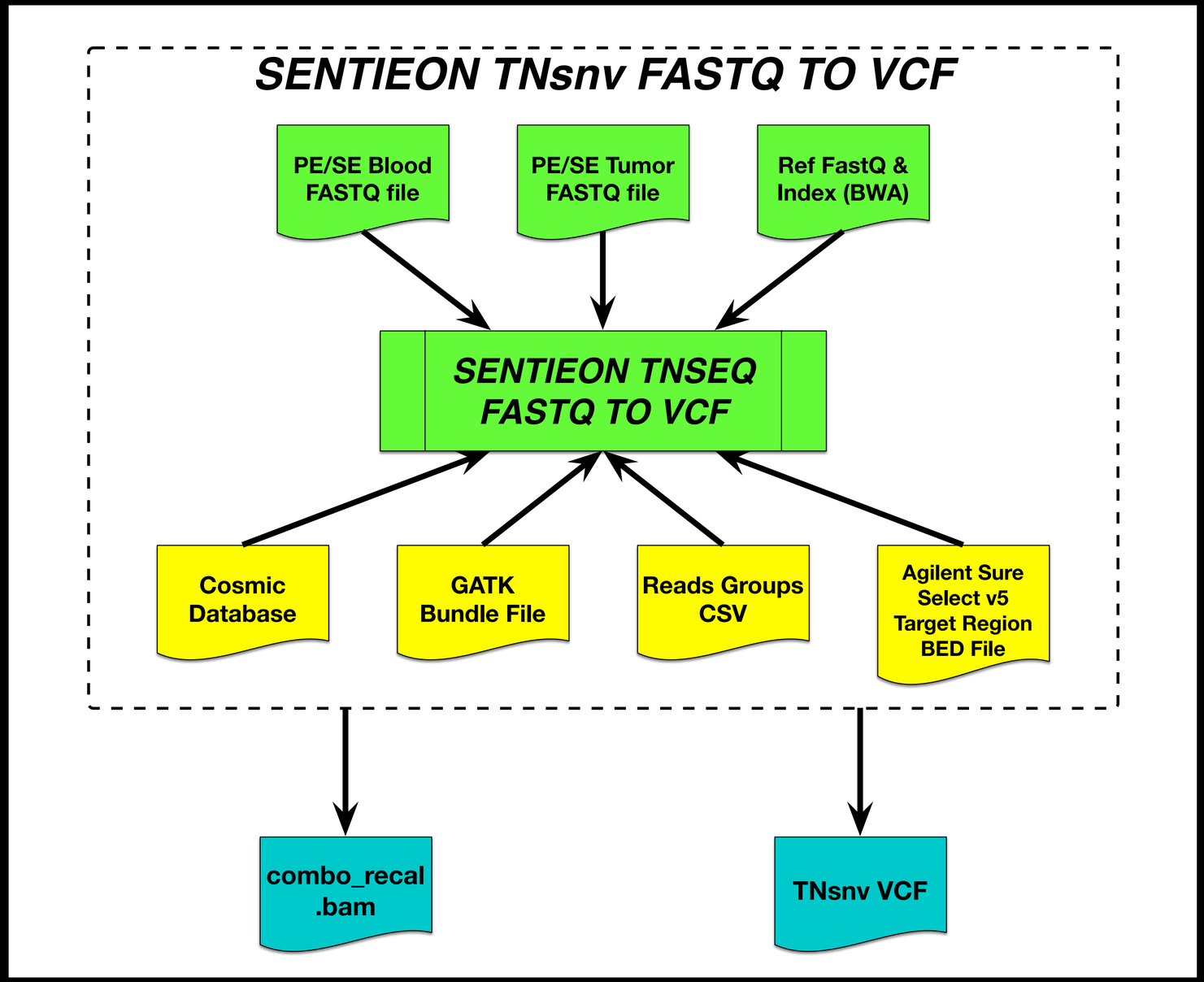
CellPress

Tumor Mutational Burden and Efficacy of Nivolumab Monotherapy and in Combination with Ipilimumab in Small-Cell Lung Cancer

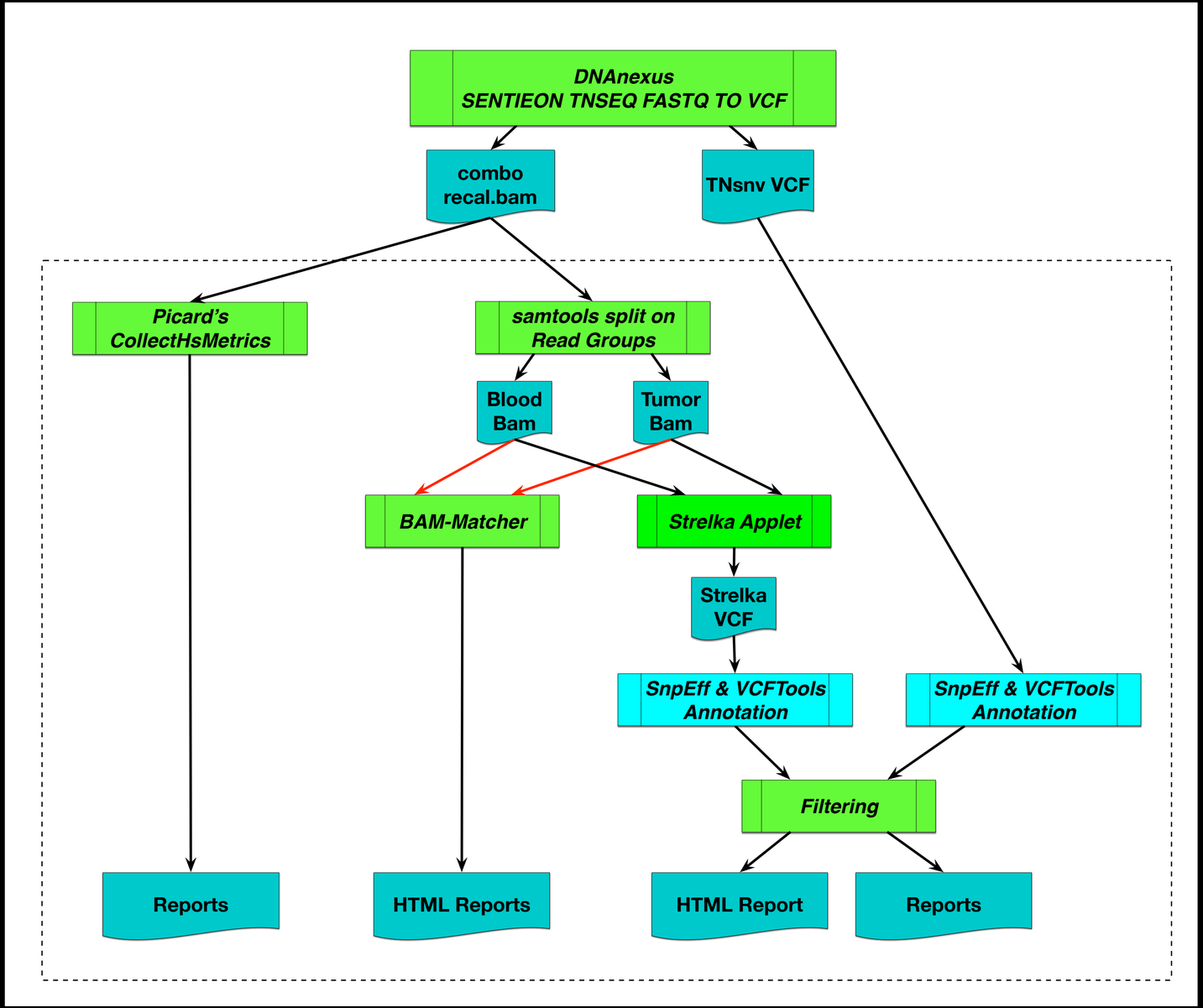
Matthew D. Hellmann,[1,11,12,*] Margaret K. Callahan,[1,11] Mark M. Awad,[2] Emiliano Calvo,[3] Paolo A. Ascierto,[4] Akin Atmaca,[5] Naiyer A. Rizvi,[6] Fred R. Hirsch,[7] Giovanni Selvaggi,[8] Joseph D. Szustakowski,[9] Ariella Sasson,[9] Ryan Golhar,[9] Patrik Vitazka,[9] Han Chang,[9] William J. Geese,[9] and Scott J. Antonia[10]

Cancer Cell. 2018 May 14;33(5):853-861.e4. doi: 10.1016/j.ccell.2018.04.001. Epub 2018 May 3.

Tumor Mutation Burden Workflow



Tumor Mutational Burden Workflow



Tumor Mutational Burden

- Annotation withsnpEff and snpSift runs on **TNsnv** VCF and **Strelka2** VCF
  - Cosmic67 - Catalogue of Mutations In Cancer, Welcome Sanger Institute
  - ExACv03 - Exome Aggregation Consortium, Broad Institute
  - 1000 Genomes - The International Genome Sample Resource
  - dbSNPv138 - Single Nucleotide Polymorphism Database
- Filter Process
  - Missense mutations AND Cosmic67, OR
  - Missense mutations AND NOT in 3 dbs (ExACv03, 1000 Genomes, dbSNPv138)
- Filter Process Count represents Tumor Mutational Burden Value

Salmon *RNA*seq

Salmon —Don't count . . . quantify!

Overview

Salmon is a tool for quantifying the expression of transcripts using RNA-seq data. Salmon uses new algorithms (specifically, coupling the concept of quasi-mapping with a two-phase inference procedure) to provide accurate expression estimates very quickly (i.e. wicked-fast) and while using little memory. Salmon performs its inference using an expressive and realistic model of RNA-seq data that takes into account experimental attributes and biases commonly observed in real RNA-seq data.

Workflow - Three Separate Applets

- Use Salmon to “align” to Gencode transcriptome and generate both quant files (Read Count and TPM per transcript) for both transcripts and genes. Additionally, gather data from optional bootstrap for subsequent use in Sleuth DEG program. This stage designed to run with separate node for each sample.
- Given a set of “read count files” (quant.sf) generate a combined count matrix or both transcripts and genes.
- Generate interactive HtML pages for each sample or combined samples for gene count, with graphic representaion of bootstrap distribution.
- Hand off DEG and other tertiary analyses to Shiny Apps iDEP and/or Biojupies.

Salmon *RNA*seq Elements

- Two Assets
  - salmon_0.12.0_asset
  - r_base_3.5.2_asset
- Cloud Workstation
  - https://wiki.dnanexus.com/developer-tutorials/cloud-workstations

Salmon *RNA*seq

Gene Expression Analysis

About this Applet
About GAU
About BTEP

Usefull Links

See a complete summary at DNAnexus Job Monitor.

Input Transcript Quant File Summary

1. brain_rep2_quant.sf
2. brain_rep3_quant.sf
3. brain_rep1_quant.sf
4. muscle_rep2_quant.sf
5. muscle_rep3_quant.sf
6. muscle_rep1_quant.sf

Input Gene Quant File Summary

1. brain_rep2_quant_genes.sf
2. brain_rep3_quant_genes.sf
3. brain_rep1_quant_genes.sf
4. muscle_rep2_quant_genes.sf
5. muscle_rep3_quant_genes.sf
6. muscle_rep1_quant_genes.sf

Instructions

1. Download Expression Tables from *DNAnexus*
   - Download RAW Counts Table for Transcripts
   - Download TPM (Transcripts Per Million) Counts Table for Transcripts
   - Download RAW Counts Table for Genes
   - Download TPM (Transcripts Per Million) Counts Table for Genes
2. Download and Edit Design Table in *Excel* or Text Editor for use in iDEP
   - Download Design Table
3. Select Analysis Site and Upload Expression Table
   - Upload an Expression Table File to BioJupies
   - Upload an Expression Table File to iDEP

Created by: Genome Analysis Unit

Get Help for an Applet *salmon_spg_wf*

Commands on Terminal

```
module load DNAnexus
dx login
dx select “DEMO_Project"
dx ls -l /SalmonWF/Applications/salmon_spg_wf
dx run /SalmonWF/Applications/salmon_spg_wf -- help
```

```
usage: dx run /SalmonWF/Applications/salmon_spg_wf [-iINPUT_NAME=VALUE ...]

Applet: salmon_spg_wf

salmon_spg_wf

Inputs:
  fastq_gz_list: -ifastq_gz_list=(file) [-ifastq_gz_list=... [...]]

  salmon_idx_file: -isalmon_idx_file=(file)

  Bootstrap Value: [-ibootstrap_value=(int, default=0)]
        Number of bootstraps to preform

Outputs:
  Salmon Results Directories (tar.gz): quant_sf_files (array:file)

  Batch of quant_sf files: quant_sf_s (array:file)
        Salmon quant.sf files.

  Batch of quant_genes_sf files: quant_genes_sf_s (array:file)
        Salmon quant.genes.sf files.

  Batch of abundance_h5 files: abundance_h5_s (array:file)
        Wasabi derived files need for Sleuth.
```

Upload Data for *salmon_spg_wf*

Commands on Terminal to Upload Data

```
dx mkdir /demo_data
dx upload *.fastq.gz --destination /demo_data
dx upload yeast_S288C_salmon_idx.tar.gz --destination /demo_data/yeast_S288C_salmon_idx.tar.gz
dx ls -l /demo_data
```

```
Project: GAU_Development (project-FVqKF6j0v1xv6fxK15Bzj9B2)
Folder : /demo_data
State   Last modified       Size      Name (ID)
closed  2019-04-10 13:43:20 81.42 MB  DST1_G418_B_R1.fastq.gz (file-FXg2fG80v1xkp89Y9jz3xvzY)
closed  2019-04-10 13:43:20 89.09 MB  DST1_G418_B_R2.fastq.gz (file-FXg2fJ80v1xx8p178Vv7B3XG)
closed  2019-04-10 13:43:25 108.73 MB DST1_G418_C_R1.fastq.gz (file-FXg2fJj0v1xq0189BbQpB6Fg)
closed  2019-04-10 13:43:30 116.46 MB DST1_G418_C_R2.fastq.gz (file-FXg2fK80v1xb8QJ47kG69jjG)
closed  2019-04-10 14:06:42 135.66 MB yeast_S288C_salmon_idx.tar.gz (file-FXg319Q0v1xkpVV6BvgfQy2G)
```

Run *salmon_spg_wf*

Commands on Terminal to run and see results for *salmon_spg_wf*

```
dx run /SalmonWF/Applications/salmon_spg_wf \
-ifastq_gz_list=/demo_data/DST1_G418_B_R1.fastq.gz \
-ifastq_gz_list=/demo_data/DST1_G418_B_R2.fastq.gz \
-ifastq_gz_list=/demo_data/DST1_G418_C_R1.fastq.gz \
-ifastq_gz_list=/demo_data/DST1_G418_C_R2.fastq.gz \
-isalmon_idx_file=/demo_data/yeast_S288C_salmon_idx.tar.gz \
-ibootstrap_value=0 \
--destination /demo_result
dx ls -l /demo_result
```

```
Project: GAU_Development (project-FVqKF6j0v1xv6fxK15Bzj9B2)
Folder : /demo_result
logs/
State   Last modified       Size      Name (ID)
closed  2019-04-10 14:13:55 81.93 KB  DST1_G418_B_abundance.h5 (file-FXg33yj0bKxY49FpJvfKb0fX)
closed  2019-04-10 14:13:55 218.73 KB DST1_G418_B_quant.sf (file-FXg33y00bKxv4BbZ5X7k0k3V)
closed  2019-04-10 14:13:55 179.09 KB DST1_G418_B_quant_genes.sf (file-FXg33yQ0bKxXG5PXJv13fBb5)
closed  2019-04-10 14:13:55 613.44 KB DST1_G418_B_salmon.tar.gz (file-FXg33x80bKxxBpp8JvVK4qBF)
closed  2019-04-10 14:13:55 82.36 KB  DST1_G418_C_abundance.h5 (file-FXg34000bKxqZVvY9VQXJp6G)
closed  2019-04-10 14:13:55 219.41 KB DST1_G418_C_quant.sf (file-FXg33z80bKxVbfF3BF3gXY6K)
closed  2019-04-10 14:13:55 179.46 KB DST1_G418_C_quant_genes.sf (file-FXg33zj0bKxVq47P9QxyQgXY)
closed  2019-04-10 14:13:55 616.78 KB DST1_G418_C_salmon.tar.gz (file-FXg33yj0bKxXG5PXJv13fBbJ)
```

Markdown Documentation

MacDown
Markdown Editor for Mac OS
https://macdown.uranusjr.com

```
<!-- dx-header -->
# salmon\_spg\_wf


This application takes

[Created by GAU](https://gau.ccr.cancer.gov)
<!-- /dx-header -->

<!-- Insert a description of your app here -->
# About Applet ...
Salmon Scatter-Process_Gather Workflow

This applet process a batch of pair-end _*.fastq.gz_ read files and runs
[Salmon](https://salmon.readthedocs.io/en/latest/).

To use the developwer's words:
> Salmon is a tool for **wicked-fast** transcript quantification from RNA-seq
data. It requires a set of target transcripts (either from a reference or de-
novo assembly) to quantify. All you need to run Salmon is a FASTA file
containing your reference transcripts and a (set of) FASTA/FASTQ file(s)
containing your reads. Optionally, Salmon can make use of pre-computed
alignments (in the form of a SAM/BAM file) to the transcripts rather than the
raw reads.

Developed by: [Fitzgerald, Peter (NIH/NCI) [E]] (<fitzgepe@mail.nih.gov>) and
[McIntosh, Carl (NIH/NCI) [E]] (<mcintoshc@mail.nih.gov>)

Group: [Genome Analysis Unit](https://gau.ccr.cancer.gov)

## Required Input Files
**_FASTQ Gzip Compressed Paired-end Files_** - A batch sample PE read files with
the form _*\_R1.fastq.gz_ and _*\_R2.fastq.gz_.

**_Salmon Index tar.gz File_** - A Salmon Indexed genome files with the form
_*\_salmon_idx.tar.gz_ .

## Input Parameters
**_Output Folder_** - Provide an output directory name for result files.

**_Instance type_** - Asking for more computer resources will reduce run time
and will cost more.
```

salmon_spg_wf

This application takes

Created by GAU

About Applet ...

Salmon Scatter-Process_Gather Workflow

This applet process a batch of pair-end *.fastq.gz read files and runs Salmon.

To use the developwer's words:

> Salmon is a tool for **wicked-fast** transcript quantification from RNA-seq data. It requires a set of target transcripts (either from a reference or de-novo assembly) to quantify. All you need to run Salmon is a FASTA file containing your reference transcripts and a (set of) FASTA/FASTQ file(s) containing your reads. Optionally, Salmon can make use of pre-computed alignments (in the form of a SAM/BAM file) to the transcripts rather than the raw reads.

Developed by: Fitzgerald, Peter (NIH/NCI) [E] and McIntosh, Carl (NIH/NCI) [E]

Group: Genome Analysis Unit

Required Input Files

FASTQ Gzip Compressed Paired-end Files - A batch sample PE read files with the form *_R1.fastq.gz and *_R2.fastq.gz.

Salmon Index tar.gz File - A Salmon Indexed genome files with the form *_salmonidx.tar.gz_ .

Input Parameters

Output Folder - Provide an output directory name for result files.

Instance type - Asking for more computer resources will reduce run time and will cost more.

COMMON Input Parameters

platform.dnanexus.com/projects/FVqKF6j0v1xv6fxK15Bzj9B2/data/SalmonWF/Applications

CONFIGURE: SALMON_SPG_WF (APPLET)

✔ SSH is allowed for this app.

salmon_spg_wf

About Applet ...

Salmon Scatter-Process_Gather Workflow

This applet process a batch of pair-end *.fastq.gz read files and runs Salmon.

To use the developper's words:

> Salmon is a tool for **wicked-fast** transcript quantification from RNA-seq data. It requires a set of target transcripts (either from a reference or de-novo assembly) to quantify. All you need to run Salmon is a FASTA file containing your reference transcripts and a (set of) FASTA/FASTQ file(s) containing your reads. Optionally, Salmon can make use of pre-computed alignments (in the form of a SAM/BAM file) to the transcripts rather than the raw reads.

Developed by: [Fitzgerald, Peter (NIH/NCI) [E]] (fitzgepe@mail.nih.gov) and [McIntosh, Carl (NIH/NCI) [E]] (mcintoshc@mail.nih.gov)

Group: Genome Analysis Unit

Required Input Files

FASTQ Gzip Compressed Paired-end Files - A batch sample PE read files with the form *_R1.fastq.gz and *_R2.fastq.gz.

Salmon Index tar.gz File - A Salmon Indexed genome files with the form *_salmon_idx.tar.gz .

Input Parameters

Output Folder - Provide an output directory name for result files.

Instance type - Asking for more computer resources will reduce run time and will cost more.

COMMON Input Parameters

Bootstrap Value (integer)

> Salmon has the ability to optionally compute bootstrapped abundance estimates. This is done by resampling (with replacement) from the counts assigned to the fragment equivalence classes, and then re-running the optimization procedure, either the EM or VBEM, for each such sample. The values of these different bootstraps allows us to assess technical variance in the main abundance estimates we produce. Such estimates can be useful for downstream (e.g. differential expression) tools that can make use of such uncertainty estimates. This option takes a positive integer that dictates the number of bootstrap samples to compute. The more samples computed, the better the estimates of varaiance, but the more computation (and time) required.

Output Files

Per sample file, the following files are produced:

- ***Salmon Results Directory tar.gz File*** - A file of form *__salmon.tar.gz_*. This is a directory that is tar.gz compressed and needs to be expanded using the command `tar -xzf tarfile` . These files are provided if you wish to do some custom analysis. Otherwise, it can be ignored.
- ***Salmon's Quant.sf File*** - A file of form *__quant.sf_*. This file contains counts.
- ***Kallisto's abundance.h5 File*** - A file of form *__abundance.h5_*. This files is transformed from Salmon's quant.sf file in a Kallisto's HDF (Hierarchical Data Format) file. This file is provied for downstream analysis using Sleuth.

Additional Files:

- ***Script File*** - This file is under development.

* Fields are required

Name: salmon_spg_wf *

Output Folder: *

Instance type: mem1_ssd1_x4 Select

COMMON

Bootstrap Value: 0 *

Reset to applet defaults | Save

DNAnexus Developer Pages

https://wiki.dnanexus.com/Developer-Portal

https://wiki.dnanexus.com/Developer-Tutorials/Intro-to-Building-Apps

Support Pages

- DNAnexus CCR Pilot (*https://gau.ccr.cancer.gov/dna-nexus-pilot-program/*)
- Slack Channel for CCR_DNAnexus Pilot (dnaxpilot.slack.com) (help, general, development)
- Creating Assets: https://gau.ccr.cancer.gov/about-dnanexus-asset/
- Building
- Example About Pages:
  - https://gau.ccr.cancer.gov/rnaseq_salmon/
  - https://gau.ccr.cancer.gov/salmon_spg_wf/
  - https://gau.ccr.cancer.gov/quant_sf2express_table/