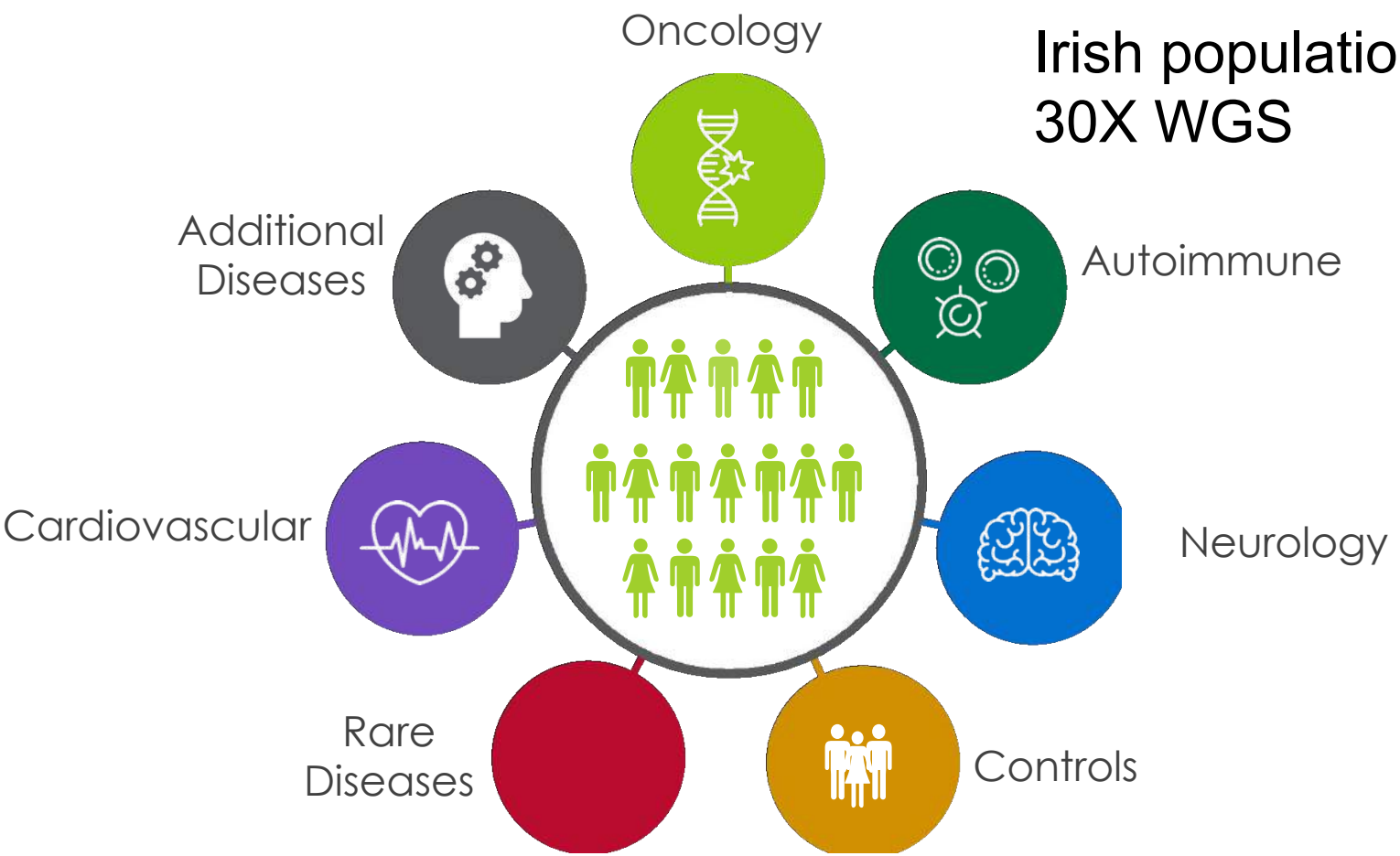
Using Nextflow to create scalable and reproducible pipelines @ Genomics Medicine Ireland

Simone Coughlan

@coughls

Disease-Specific and Cross-Disease Research

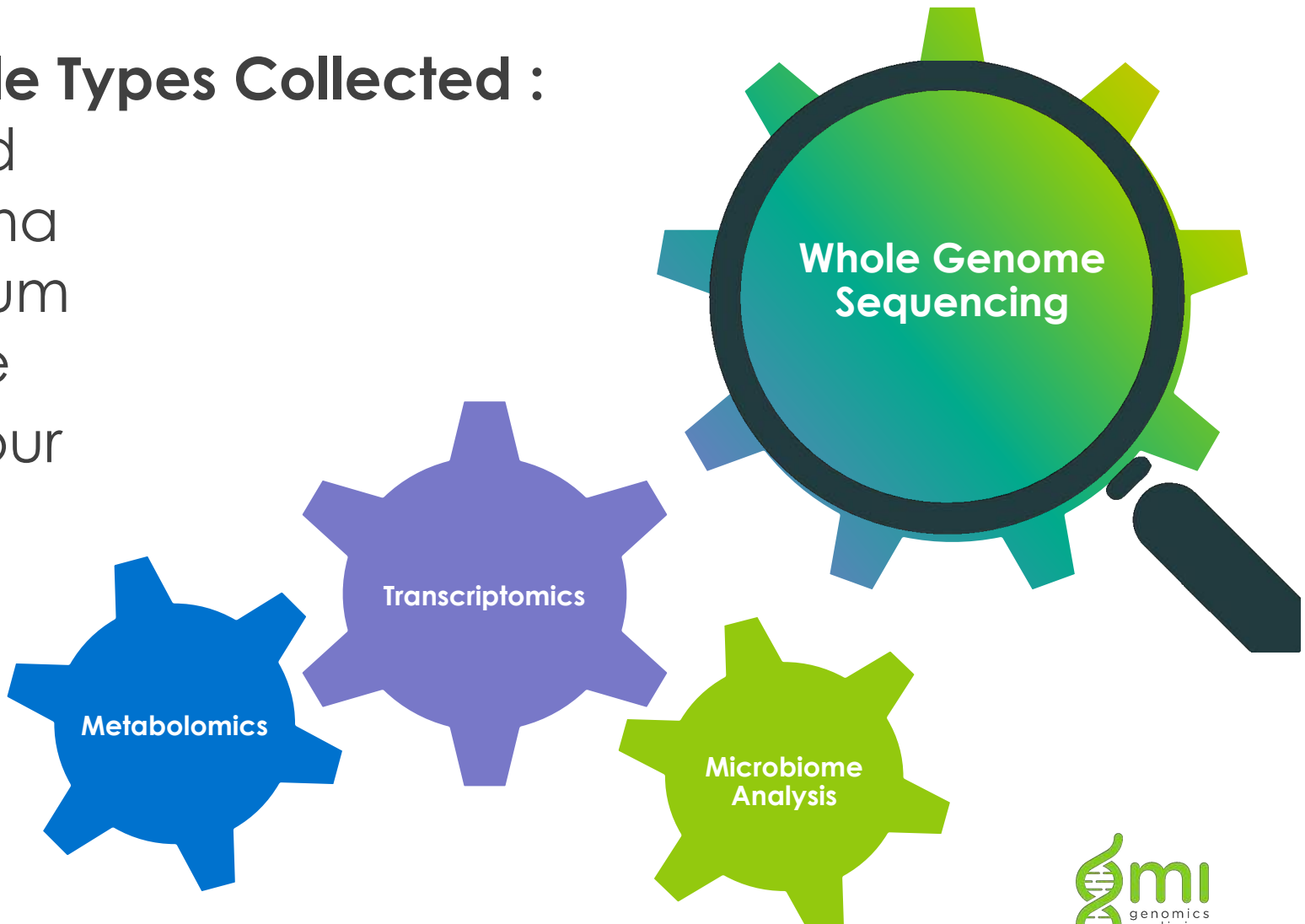


genomics medicine ireland

Multi-Omics Approach

Sample Types Collected :

- Blood
- Plasma
- Sputum
- Tissue
- Tumour



gmi genomics medicine ireland

GMI - Genomics Centre

- Opened 2018
- 10,000+ Sq.ft.
- High-volume sequencing
- 5 Novaseqs & 1 GridION
- CAP accredited

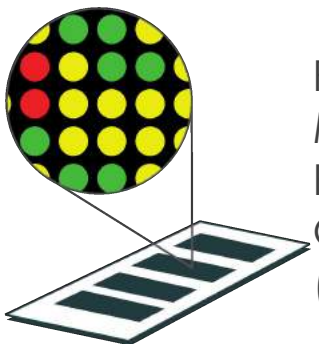
genomics medicine ireland

What Happens to Bio-Specimens

1

Nucleic acid extraction (FlexStar+), QIAsymphony for RNA

2



Precision Medicine Research Array Genotyping (Gene Titan)

3

Data QC(Roche Light cycler, Agilent TapeStation, Thermofisher Varioskan), Informatics

4

Next Generation Sequencing

5

Biobanking (DNA, RNA, plasma for metabolites, saliva, FFPE)

Biobank capacity for 500,000 samples

gmi genomics medicine ireland

Bioinformatics

- Team of 7
- Develop pipelines for both everyday and bespoke analysis
- Work closely with lab, research & IT

aws

gmi genomics medicine ireland

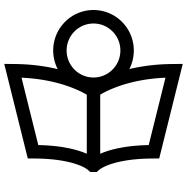
Why nextflow

Early 2017

Decided I needed to move away from bash scripts

Mid 2017

Needed to run a lot of files in parallel -also reproducibility!



Started reading about workflow managers- tried nextflow

Nov2017

Attended nf-hack 17

genomics medicine ireland

Why nextflow

- Ability to get off the ground fast!

- Easy to install

```
wget -qO- https://get.nextflow.io | bash
```

- Minimal code changes needed
- Good 
- Fast prototyping of working pipeline -  resume!
- Parallelism is implicit - can change machine and it adapts 
- Can scale up or out with ease
- User community so that can google errors/?!

Additional Benefits

- Separate folders for each task of a process
- Logging & reporting
- DAGs -> great for discovering channels you have made but not used!
- Ability to tag processes
- Seamless integration with gitlab/github and aws
- Very satisfying seeing lots of 'Submitted process'

genomics medicine ireland

Image ref: https://www.nextflow.io/docs/latest/tracing.html

Learning nextflow in GMI

Started small

Parallelised a python few scripts in one script on one instance

+

Started making dockerfiles

Triggered by large amount of time spent installing a piece of software that not in conda

=

Implemented first dockerised nextflow pipeline

Easy to run, reproducible and logged!

gmi genomics medicine ireland

Learning nextflow

Annotation pipeline for VCFs for validation

- open source & own python scripts in nextflow
- VCF -> CSV of variants of interest

+

Concordance pipeline

- QC check
- Started using conda
- Uses nf-core base template
- VSCode
- docker, conda & test configs

nf-core

CONDA

docker

aws

gmi genomics medicine ireland

nf-core

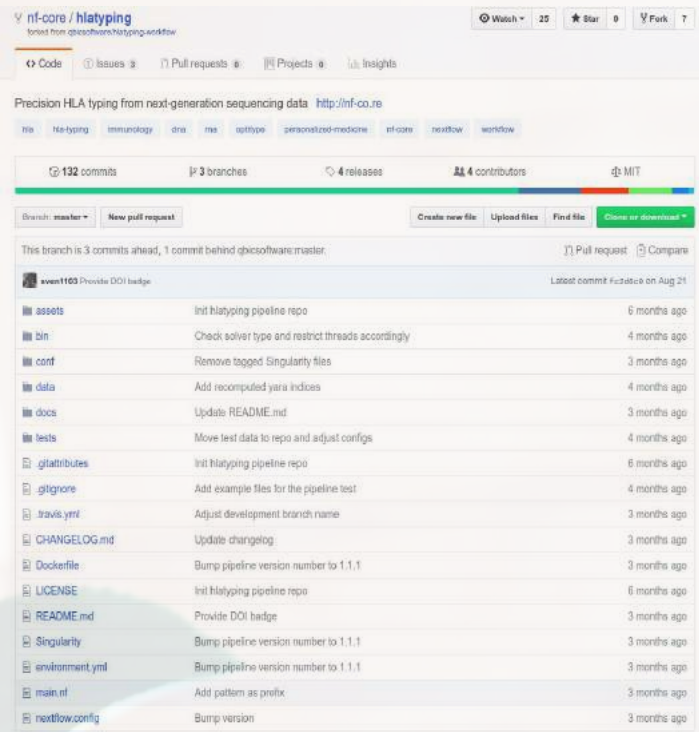
- Proper layout of repository and templates

- Looks like you have done lots of work with two commands!

```
pip install nf-core
nf-core create -n nextbigthing -d "This pipeline analyses data from the next big 'omics technique"
```

- Comes with some documentation which served as a good base to alter and add my own documentation

- Separate config files for docker,aws, testing so learnt how to use these and make my own



https://github.com/nf-core/hlatyping

https://github.com/nf-core

Bioinformatics pipelines

- Genotyping qc pipeline
- Concordance of genotyping and sequencing
- Copy number variation calling
- VCF annotation

gmi genomics medicine ireland

Nextflow and Conda

- Was convinced of the benefits of using conda by a colleague
- Started using it for creating environment.yml file to keep with python scripts
- Then saw blog..

nextflow Features Quick start Examples Developers Blog About Us

Conda support has landed!

Paolo Di Tommaso 05 June 2018

Nextflow aims to ease the development of large scale, reproducible workflows allowing developers to focus on the main application logic and to rely on best community tools and best practices.

For this reason we are very excited to announce that the latest Nextflow version (`0.30.0`) finally provides built-in support for Conda.

CONDA

gmi genomics medicine ireland

Nextflow and Conda

- Conda = even faster development
- I get conda from miniconda3 - can still make environments that use python2 if needed from it
- No need to look up and install each piece of software
- I google for the conda package to get installation command and then adapt for nextflow

genomics medicine ireland

Nextflow & Conda Example

https://anaconda.org/bioconda/bcftools

Installers

conda install 1)

linux-64 v1.9

osx-64 v1.9

To install this package with conda run:

```
conda install -c bioconda bcftools
```

In nextflow

2)

```
process bar {

conda 'bioconda::bcftools'

script:
"""
bcftools ...
"""

}
```

Using bcftools version 1.9

conda 'bioconda::bcftools=1.9

Using bcftools version 1.9 and python version 3.6

conda 'bioconda::bcftools=1.9 python=3.6'

genomics medicine ireland

Nextflow & Conda Config

- While writing script, each process has its own conda environment with tools needed by the process
- I usually use the same environments for multiple processes
- But may use different conda env where two tools are incompatible
- Easier to put in a conda config file, and reference that in nextflow.config file
- Can deploy pipeline with different tools using ‘-profile’

gmi genomics medicine ireland

Conda Config File Example

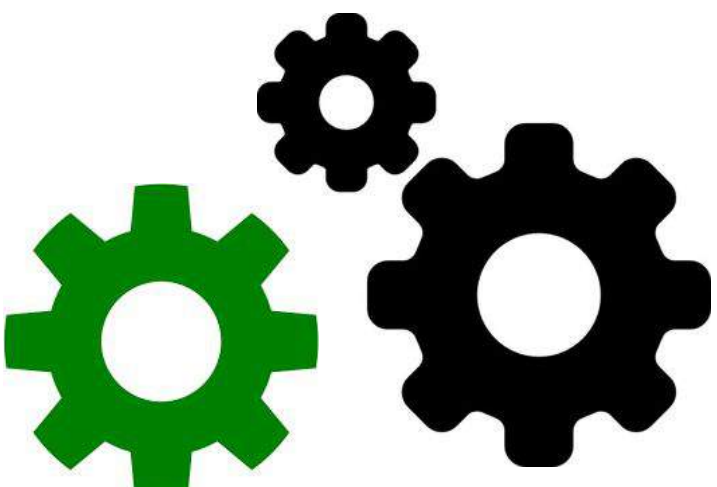
conf/conda.config

```
/*
 * -------------------------------------------------
 *  pipeline-nf  Nextflow conda config file
 * -------------------------------------------------
 * A 'conda' config file
 */

process {

    withName: 'foo' {
    conda = 'python=2.7'
    }
    withName: '!foo'  {
    conda = 'python=3.6 biopython'

    }
}
```



Docker Image from Conda

- Make conda environment file

```
##conda
conda create -n myenv python=3.6
source activate myenv
#can install more software if needed now; then
conda env export > environment.yml
source deactivate
```

- Make Dockerfile and build image

```
##Make dockerfile
mkdir docker_image_folder
cd docker_image_folder

##Dockerfile
$less Dockerfile
FROM continuumio/miniconda3
MAINTAINER yourname  <youremail>
LABEL authors="youremail" \
    description="python3.6 docker image"

COPY environment.yml /
RUN conda env update -n root -f /environment.yml && conda clean -a

##Build docker image
docker build -t dockerimage/pythonimage .
```

Multiple buckets & keys

- Normally nextflow can automatically pull a file from S3 given a filepath (use default profile)
- IAM role avoids needing to handle keys = preferred

- Can add one set of keys in nextflow.config as follows:

```
aws {
  accessKey = '<Your AWS access key>'
  secretKey = '<Your AWS secret key>'
  region = '<AWS region identifier>'
}
```

- But have multiple keys in AWS credentials file which are specified using profiles when using aws client e.g.

```
#show files in alpha and beta buckets
aws s3 ls --profile alpha alpha_bucket
aws s3 ls --profile beta beta_bucket
```

genomics medicine ireland

Multiple buckets & keys

- Workaround -
  Explicitly download files in a process
  Put full path to files in csv and read in csv to channel
  Then set profiles for downloading as below

```
process download_file {

   script:
    //Set profile to use for downloading
    if ( filename.toString().contains('alpha_bucket')) {
        profile = 'alpha'
    } else {
        profile = 'beta'
    }

    """
    #!/bin/bash
    aws s3 cp $filename .  --profile $profile
    """
}
```

Multiple buckets & keys

- Or use a python script with boto3 and set different sessions depending on the bucket name and reference this in a nextflow process

```
#python boto3
boto3.session.Session(profile_name="")
```

- Other benefits of leveraging boto3 e.g. to check for storage class of object

- Exposing credentials to docker
  Don't want to add credentials to image (security) or if using public image

```
#in nextflow.config
docker {
runOptions =  '-v ~/.aws/credentials:/root/.aws/credentials'
}

#run pipeline as normal
nextflow run main.nf -profile docker
```

gmi genomics medicine ireland

Spaces in names

- Filenames with spaces cause pipeline to fail if not enclosed by quotes

Example

```
#make two test files -one with a space in the name
#and one with no space
touch test1.txt
touch 'test 2.txt'
```

genomics medicine ireland

```
$less test-noquotes.nf

#!/usr/bin/env nextflow

params.files="./*.txt"

files_ch = Channel
    .fromPath(params.files)
    .ifEmpty { error "Cannot find any files matching: ${params.files}" }
    .map { file -> tuple(file.name - ~/.txt$/ , file) }

process test_noquotes {

   tag "$filename"

    input:
    set filename, file(txt_file) from files_ch

    output:
    file("${filename}.num_lines.txt") into files_out

    script:
    """
    wc -l $txt_file > ${filename}.num_lines.txt
    """

}
```

files_ch
[test1, test1.txt]
[test 2, test 2 .txt]

files_out
[test1.num_lines.txt]
[test 2.num_lines.txt]

genomics medicine ireland

```
nextflow run test-noquotes.nf
N E X T F L O W  ~  version 0.31.1
Launching `test-noquotes.nf` [mighty_golick] - revision: 4fd66e33bb
[warm up] executor > local
[d1/9be607] Submitted process > test_noquotes (test 2 )
[44/32fe45] Submitted process > test_noquotes (test1)
ERROR ~ Error executing process > 'test_noquotes (test 2 )'

Caused by:
  Process `test_noquotes (test 2 )` terminated with an error exit status (1)

Command executed:

  wc -l test 2 .txt > test 2 .num_lines.txt

Command exit status:
  1

Command output:
  (empty)

Command error:
  wc: 2: No such file or directory
  wc: .txt: No such file or directory
  wc: 2: No such file or directory
  wc: .num_lines.txt: No such file or directory

.....
```

Spaces in names - handle

Replace

```
script:
"""
wc -l $txt_file > ${filename}.num_lines.txt
"""
```

with

```
script:
"""
wc -l "$txt_file" > "${filename}.num_lines.txt"
"""
```

genomics medicine ireland

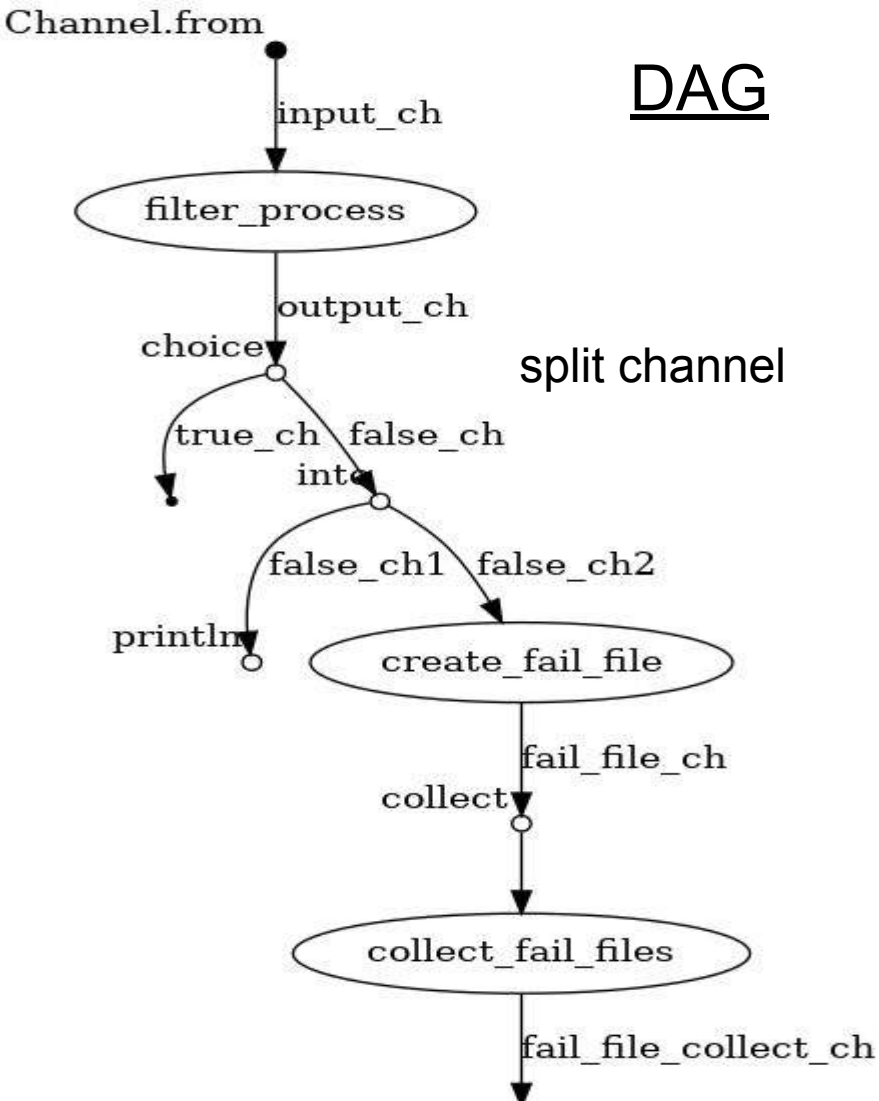
Spaces in names - it runs!

```
nextflow run test-withquotes.nf
N E X T F L O W  ~  version 0.31.1
Launching `test-withquotes.nf` [loving_heisenberg] - revision: a5de9955a6
[test1, /home/.../test1.txt]
[test 2 , /home/.../test 2 .txt]
[warm up] executor > local
[a0/68819e] Submitted process > test_noquotes (test1)
[65/8de638] Submitted process > test_noquotes (test 2 )
/home/.../work/a0/68819e9de3f74e6229a6b753044827/test1.num_lines.txt
/home/.../work/65/8de63864dff08db1ebdc82d72a1659/test 2 .num_lines.txt
```

genomics medicine ireland

Splitting one channel based on stdout

Aim: Split a channel into two channels based on a stdout result -> 1 true and 1 false channel and report contents of false channel in a file




```
nextflow run filter-example.nf -with-dag filter-example-dag.png
```

```
#!/usr/bin/env nextflow

params.outdir = "./RESULTS"

input_ch = Channel.from(1,2,3,4,5)

process filter_process {

    input:
    val x from input_ch

    output:
    set stdout, x into output_ch

    script:
    if( x >=3 ) {
    """
    #!/bin/bash
    printf "True"
    """
    }
    else {

    """
    #"/bin/bash
    printf "False"
    """
    }
}
```

input_ch
1
2
3
4
5

(five runs of filter_process)

output_ch
[False, 1]
[False, 2]
[True, 3]
[True, 4]
[True, 5]

```
/*
 * Filter to keep only true results from output_ch
 */

false_ch = Channel.create()
true_ch = Channel.create()

output_ch.choice( true_ch, false_ch ) { a -> a[0] =~ /^True.*/ ? 0 : 1 }

false_ch.into{false_ch1; false_ch2}
false_ch1.println()
```

genomics medicine ireland

```
//Send info on each task in channel to a file
process create_fail_file{

    tag "$status - $num"

    input:
    set status, num from false_ch2

    output:
    file("*.fail.csv") into fail_file_ch

    script:
    """
    echo "$status", "$num" > "${num}.fail.csv"
    """
}

//Collect all files
process collect_fail_files{

    publishDir "${params.outdir}/", mode: 'copy'

    input:
    file("*") from fail_file_ch.collect()

    output:
    file("all_fail.csv") into fail_file_collect_ch

    script:
    """
    #!/bin/bash
    cat * > all_fail.csv
    """
}
```

genomics medicine ireland

```
nextflow run filter-example.nf -with-dag filter-example-dag.png
N E X T F L O W  ~  version 0.31.1
Launching `filter-example.nf` [backstabbing_saha] - revision: fb18aa8331
[warm up] executor > local
[92/79a3be] Submitted process > filter_process (5)
[60/9ce23a] Submitted process > filter_process (1)
[ac/ea33ed] Submitted process > filter_process (2)
[ab/ed2b51] Submitted process > filter_process (3)
[b5/35166a] Submitted process > filter_process (4)
[False, 2]
[False, 1]
[e6/71e3d8] Submitted process > create_fail_file (False - 2)
[b4/4e1fc9] Submitted process > create_fail_file (False - 1)
[d3/d6c6e1] Submitted process > collect_fail_files
```

gmi genomics medicine ireland

Acknowledgements

- Genomics Medicine Ireland & Bioinformatics Team
- Paolo Di Tommaso
- The nextflow community
- Phil Ewels (nf-core)
- Anna Sole Amat (organising)
- We are hiring Bioinformaticians! (https://genomicsmed.ie/careers/)

gmi genomics medicine ireland