Building Workflows with Nextflow

Tobias Schraink , Stephen Kelly

March 28, 2018

New York University, New York, New York, USA

https://github.com/tobsecret/BADAS_Nextflow_Tutorial

"In silico workflow management systems are an integral part of large-scale biological analyses"

- Ease of development
- Standardized format
- Consistent structure
- Reproducibility
- Manage complexity

Nextflow

- Domain-specific language
- parallel asynchronous execution
- adaptation of existing pipelines
- built on Groovy (Java)
    - no user ‘installation’ required

Key Aspects

- Executes tasks in any scripting language
  - bash, R, Python, etc.
- built-in support for Docker, Singularity, environment modules
- built-in support for AWS, HPC schedulers (SGE, SLURM, LSF, etc.)
- **decoupling of pipeline tasks from task-execution logic and environment management**
  - allows for greater portability
  - Docker + Nextflow = 100% portable pipeline

Design

- '**Channels**' and '**Processes**'
  - Channels: uni-directional pipes to pass files, values, data, etc. to processes
  - Processes: tasks to be performed in the pipeline
- Processes executed in isolation from each other, communicate via channels

Basic Examples

Pipeline script:

Input Channel

task Process

main.nf

```
1  Channel.from( ['Sample1','Sample2','Sample3','Sample4'] ).set { samples }
2
3  process print_sample {
4      echo true
5
6      input:
7      val(sampleID) from samples
8
9      script:
10     """
11     echo "[print_sample] ${sampleID}"
12     """
13 }
```

output:

```
[2018-04-04 16:13:59]
kellys04@acc38pathlabmac01:~/projects/nextflow-demos/print-samples2$ ./nextflow run main.nf
N E X T F L O W  ~  version 0.28.0
Launching `main.nf` [cranky_brattain] - revision: e13962af7d
[warm up] executor > local
[f7/7bbaa0] Submitted process > print_sample (1)
[3e/b1a9f9] Submitted process > print_sample (4)
[da/a02b84] Submitted process > print_sample (3)
[f4/abcab6] Submitted process > print_sample (2)
[print_sample] Sample4
[print_sample] Sample3
[print_sample] Sample2
[print_sample] Sample1
```

Managing file input & output

- **Don't!**

- Nextflow handles this automatically, you only need to manage process 'input' and 'output' in your pipeline

```
process align_sample {

    input:
    file 'reference.fa' from genome_ch
    file 'sample.fq' from reads_ch

    output:
    file 'sample.bam' into bam_ch


    script:
    """
    bwa mem reference.fa sample.fq \
          | samtools sort -o sample.bam
     """

}
```

```
process index_sample {

    input:
    file 'sample.bam' from bam_ch

    output:
    file 'sample.bai' into bai_ch


    script:
    """
    samtools index sample.bam
     """
}
```

Nextflow Advantages

- extremely portable & lightweight
- makes pipelines easy to run, maintain, troubleshoot
- robust process isolation, input/output file management
  - failed compute jobs don't affect pipeline re-runs
- **greatly reduces overhead of environment management & task execution on HPC cluster**
  - **greatly reduces pipeline code complexity and debugging!!**

Notes

- very active developer support
  - primary developer Paolo responds quickly to posts on Google Groups, GitHub
  - Google Cloud Platform for genomics integration expected end of 2018
- Pairs well with Docker/Singularity for dependency management, Makefiles for config management & execution shortcuts
- lots of helpful extra features
  - HTML pipeline reports, email output & notifications
- Knowledge of Groovy & Java not required but it helps

Hands-on Workshop Session

- https://github.com/tobsecret/BADAS_Nextflow_Tutorial

Examples

- Nextflow tutorial: https://github.com/nextflow-io/hack17-tutorial
- Nextflow examples: https://github.com/nextflow-io/examples
- Pipeline examples: https://github.com/nextflow-io/awesome-nextflow
- This workshop: https://github.com/tobsecret/BADAS_Nextflow_Tutorial
- Some more demo workflows: https://github.com/stevekm/nextflow-demos

Resources

- Nextflow Homepage: https://www.nextflow.io/
  - Publication https://www.nature.com/articles/nbt.3820
  - More slides: https://speakerdeck.com/pditommaso/enabling-reproducible-in-silico-data-analises-with-nextflow
- Nextflow Docs: https://www.nextflow.io/docs/latest/getstarted.html
- Nextflow GitHub: https://github.com/nextflow-io/nextflow
- Nextflow Google Group: https://groups.google.com/forum/#!forum/nextflow
- Groovy docs
  - http://groovy-lang.org/documentation.html
  - http://docs.groovy-lang.org/latest/html/documentation/