Nextflow integration for the Research Object Specification

Edgar Garriga Nogales
Centre for Genomic Regulation (CRG)
The Barcelona Institute for Science and Technology
Barcelona, Spain
edgar.garriga@crg.eu

Paolo Di Tommaso
Centre for Genomic Regulation (CRG)
The Barcelona Institute for Science and Technology
Barcelona, Spain
paolo.ditommaso@crg.eu

Cedric Notredame
Centre for Genomic Regulation (CRG)
The Barcelona Institute for Science and Technology
Barcelona, Spain
cedric.notredame@crg.eu

Keywords— nextflow, research object, reproducibility, provenance

I. Introduction

For reproducibility criteria to be met in a scientific context an increasing number of conditions need to be fulfilled. These conditions, explicited by the FAIR principle include traceability, reusability and data/methods permanent availability (findable). The challenge is not only to keep the right elements bundled together, but also to keep track of each component individual history (including individual updates) while associating every computational analysis with a transparent source. This issue, known as provenance, is the one we have been addressing in the context of this proposal. It is increasingly critical, at a time when a growing number of computational procedures are used to assess medical risks and take therapeutic decision. Our solution involves using the Research Object (RO)[1] specification that have allowed us to implement a method that enables the creation of a package collecting all provenance metadata of a computational experiment, so that it can be easily shared, archived and reproduced when needed.

In practice, keeping all the required information bundled together (paper, slides, methods, pipelines, etc) can be challenging, especially when adding the constraint of fine grain querying. The aim of this this proposal is to create a package based on the Research Object specification. Thanks to this procedure, all the provenance information of a computation experiment that can be easily collected, shared and archived

We show here how a slight adaptation of a workflow tool like Nextflow[2] can allow for the seamless transfer of unique ID tags to various elements of data thus making it easier to trace the data and its associated objects for citation purposes.

II. Method

Nextflow is a framework based on the dataflow programming model, which simplifies the writing of parallel and distributed pipelines. Given a multi-step pipeline, Nextflow allows explicit dependencies to be declared between tasks thus allowing output and input to be piped across the workflow, with specific operation possibly carried out between tasks (merge, sort, split, etc…). The tasks themselves are usually encapsulated in containers and deployed by Nextflow across computational platforms (Amazon cloud, legacy batch schedulers,Kubernetes, etc.). Being able to decompose a pipeline into multiple processes, possibly written in different scripting languages (Bash, Perl, Python, etc.) simplifies the pipeline development. A major advantage of Nextflow is its ability to deploy the execution of a pipeline across multiple platforms.

Research Object is a method for the identification, aggregation, and exchange of information. The primary goal is to provide a way of associating together related resources from the same project, (i.e. the pipeline, auxiliar scripts, data, slides or the final article).

The Research Object concept is motivated by a desire to improve the reproducibility of computational methods and experiments. Its main three principles are: *Identity*, providing a unique identifier to the project, as the DOI for the publication or the ORCID for the scientist. *Aggregation*, allowing the author to wrap all the elements used for the project (i.e. slides, article, scripts, etc.). With the Research Object, we can share all the elements of a project together with the same ID. Finally, the last main principle of

Research Object is the *Annotation*, a specific layer of metadata that explicitly defines the relation between elements, as well as their time and mode of production[3]. As such, the RO technology allows having in the same package human and computer readable data while making the projects traceable and FAIR compliant.

III. Implementation

The integration will make Nextflow able to produce a zip file with the following structure:

-A *Data* folder, with all the input files of our pipeline.

-The *Workflow* folder containing all the files of the pipeline base directory (e.g. the main Nextflow script or the config file of the pipeline).

-The *Snapshot* folder is the one used re-execute the pipeline with the same parameters if its needed.

-The *Metadata* folder contains the log file of the past executions, the metadata file with information about the container used and the Nextflow version. The metadata folder contains the provenance file too. With this file we can see how and when the intermediate files where generated, and which process used them as an input.

-The *Output* folder, where the output of the execution is stored.

-The *RO* folder, containing RO's manifest with information about the author and the creation of the RO.

IV. Results

The Nextflow-RO integration allows the creation of an RO package when executing a Nextflow workflow. The result is a single package with a unique identification (identity) containing all the important files of the project (aggregation) like the metadata, logs, results and the workflow directory. Another important value of this integration is the generation of the provenance annotation. This information makes it easier to share and reproduce an analysis. It also increases the transparency on the procedure behind the analysis. These three elements contribute towards the three core principles of RO.

The main current limitation involve capturing all the relevant data in a non-user dependent way. Another issue is the burden of metadata capture on the user side, since even small scale analysis can easily generate metadata with a size larger than both the raw data and the final analysis. In a future work we plan to continue developing this feature, evolving as much as possible with the community's feedback, we will try to increase the provenance level and make the generation process as user-friendly as possible.

References

[1] Khalid Belhajjame, Jun Zhao, Daniel Garijo, Matthew Gamble, Kristina Hettne, Raul Palma, Eleni Mina, Oscar Corcho, José Manuel Gómez-Pérez, Sean Bechhofer, Graham Klyne, Carole Goble (2015) Using a suite of ontologies for preserving workflow-centric research objects, Web Semantics: Science, Services and Agents on the World Wide Web, https://doi.org/10.1016/j.websem.2015.01.003

[2] P. Di Tommaso, et al. Nextflow enables reproducible computational workflows. Nature Biotechnology 35, 316–319 (2017)

[3] Farah Zaib Khan, Stian Soiland-Reyes, Michael R. Crusoe, Andrew Lonie, & Richard O. Sinnott. (2018). CWLProv - Interoperable Retrospective Provenance capture and its challenges. Zenodo. http://doi.org/10.5281/zenodo.1215611