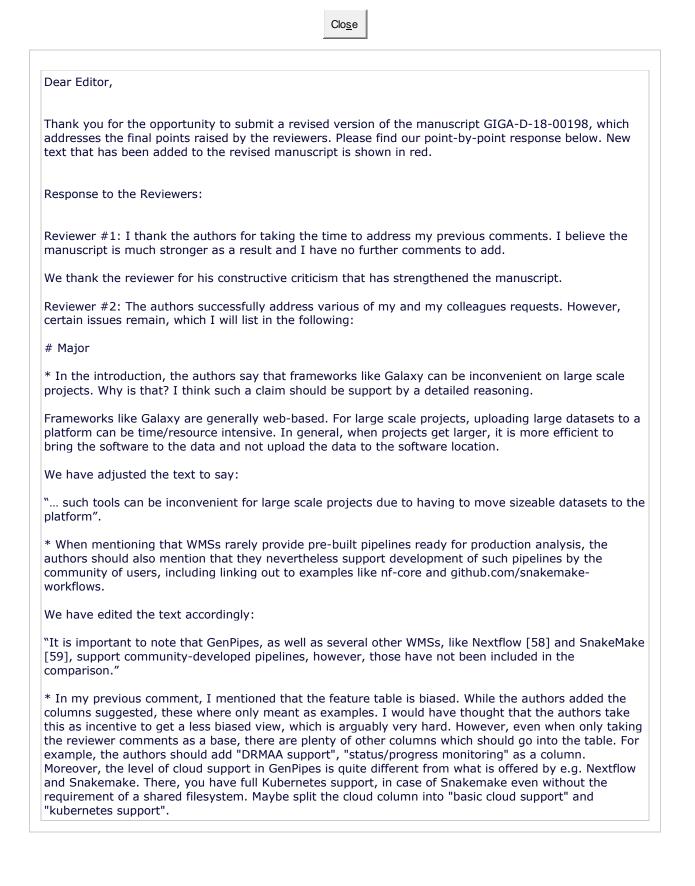
Author's Response To Reviewer Comments



Our intentions with Table1 was to provide the reader with an overview of the features of several tools in the field but not necessarily an exhaustive list. We did not design the table to be biased towards GenPipes as we only modified one of the most comprehensive tables we found in recent manuscripts (Griffith & Griffith et al.). Based on that initial table and reviewers' comments, we added 3 features and 1 WMS. Although more could be added, it would also start cluttering the table and make it difficult to extract meaningful information.

"status/progress monitoring" is already included in the table under "Tracking". We have modified the column name to make it less ambiguous.

Concerning "DRMAA support" and splitting the "cloud support" column into "basic cloud" and "kubernetes support", we feel that this is highly technical/specific for the average user.

* The installation mechanism for new software tools (outside of what is provided out of the box) (explained here: https://bitbucket.org/mugqic/genpipes/src/master/#markdown-header-modules), seems like manually redoing all the work that is already solved by package managers like conda or container engines like singularity. For example, Bioconda provides a library of over 4000 bioinformatics software packages which can be readily used from any WMS that supports conda, and Biocontainers provides the same for container based deployment (which lacks conda's ability to rapidly compose custom combinations of tools though). In order to make the comparison fair, the feature table should therefore contain two columns called "package-manager-integration" and "container-integration". For an example of what level of integration I am referring to, see

https://snakemake.readthedocs.io/en/stable/snakefiles/deployment.html#integrated-package-management and

https://www.nextflow.io/docs/latest/conda.html?highlight=conda.

Bioconda offers a collection of packages and not an integrated system and can be quite heavy in memory requirements. Hence, we think that "package-manager-integration" is not necessarily an indication of the strength of the WMS. It is a specific choice, one that offers ease of installation but has its pitfalls as well. GenPipes does not use package managers by design. GenPipes manages its own libraries making sure there is no conflicting libraries in the process. For users who do not want to install GenPipes manually, we offer a Docker container that has also been tested with Singularity. We have updated the GenPipes' bitbucket documentation to highlight the availability of the GenPipes' Docker container.

"container-integration" has already been included in the table under the "Cloud/Container" column.

* I am pleased to see that GenPipes indeed supports aggregation over many samples. What remains is the question whether the only entity to aggregate over are samples. If so, only over all samples or is it possible to express e.g. an arbitrary grouping of samples? Moreover, what about other properties, e.g. for scanning a parameter space? I suggest to somehow reflect the different ways of aggregation in the feature table, maybe using the terms that I mentioned in my first review.

GenPipes is a flexible python framework that aggregates over readsets, samples and other entities, like chromosomes, based on the pipeline. Arbitrary groupings of samples can be defined in pipelines that use design files, like chipseq and rnaseq. Scanning parameter space can be done by adjusting the configuration files. There isn't a set of limited/defined aggregation methods we use; aggregation is used based on each pipeline's needs. The user can refer to the documentation of each pipeline to see what is possible. For user implemented pipelines, there is no restriction on the aggregations possible.

We have added the following lines to the text to highlight some of these points:

- "... GenPipes can aggregate and merge samples as indicated by the readset file."
- "... Configuration files are customizable, allowing users to adjust different parameters."
- "... Custom sample groupings can be defined in the design file."

Minor

* Please mention in the caption of the feature table that community based workflows are not considered in the comparison. It might otherwise be that readers overlook this in the main text.

We have added test to the caption as follows:

"Modified from Griffith & Griffith et al. [57]. Note that community-built pipelines are not considered in the Pipelines section of the table."

* Figure S1 contains a lot of typos, e.g. "reasdet", which I guess is supposed to be readset?

Thank you, we have corrected the typos in Figure S1.

Clo<u>s</u>e