Response to the reviewers (GIGA-D-18-00483)

This is the response to the reviews of our manuscript `https://doi.org/10.5281/zenodo.1966881`, submitted to GigaScience (GIGA-D-18-00483) on 2018-12-08 and addressed in the updated preprint `https://doi.org/10.5281/zenodo.3196309`, re-submitted 2019-05-23.

We thank the reviewers for their critical assessment of our work. In the following we address their concerns point by point.

Reviewer 1

Reviewer Point P 1.1 — This paper is well structured and well written but there is a point to be addressed in the evaluation. Table 5 says that the enactment of Alignment Workflow with 'cwltool' with enabling provenance capture on MacOS could not be tested due to insufficient hardware resources. Does it mean that the step (I) in 'Evaluation Activity' for Alignment Workflow could not be executed? If so, please clarify it.

Response: We agree with the reviewer on this point. We have included this information in the caption of Table 5.

Reviewer Point P 1.2 — Sometimes 'CWLProv' and its following word are accidentally concatenated. - e.g, p2. line 13 or 14 "CWLProvoutcome", p2. line 32 "CWLProv0.6.0"

Response: Thanks for pointing this out, we have fixed these typos.

Reviewer Point P 1.3 — Figure 1 uses the spelling 'artifacts' in level 1 but this paper mainly uses 'artefacts'. It is better to use consistent spelling.

Response: We have now made sure the spellings are consistent by modifying the diagram.

Reviewer Point P 1.4 — The left side of Figure 2 shows a GATK workflow but the caption says the right side is a workflow.

Response: We have edited the caption to fix this issue.

Reviewer Point P 1.5 — Table 5 says that the enactment of Somatic Variant Calling Workflow with 'toil-cwl-runner' due to a known bug. However, the link in the table is for a issue of 'cwltool', not 'toil-cwl-runner'. I got confused because the enactment of the same workflow with 'cwltool' works. If the linked issue has occurred in 'toil-cwl-runner' for the variant calling workflow, I recommend making a link to the issue of 'toil-cwl-runner' instead of 'cwltool'. It is less confusing.

Response: We faced similar issue of Docker mount denied when testing this workflow using cwl-toil-runner on Mac. The previous link was intended to give an idea of the nature of issue and the possible solutions proposed, which we tried and could not succeed to get it to work. However, we agree that there should be a separate issue for cwl-toil-runner. We have created and linked to GitHub issue (`https://github.com/DataBiosphere/toil/issues/2680`) to avoid confusion.

Reviewer 2

Reviewer Point P 2.1 — My main concern regarding this work is that it is often stated that the re-usability of workflow resources (methods/input or output data) is facilitated but it is difficult to evaluate this claim based on CWLProv features and the proposed experiments. It is clear that re-execution of workflows is facilitated but it is unclear to what extent produced/analysed data can be considered for secondary use.

Response: It is true that we have not explored all the possible ways data re-use could be facilitated (or hindered) by the CWLProv approach. Exploring this in detail would require developing multiple user scenarios and usability testing with independent domain-experts who had not seen the archived workflow before. We believe this would be extensive future work and out of scope for this manuscript.

We have explored some CWLProv consumption scenarios in *cwlprov-py* - we refer to for its documentation `https://pypi.org/project/cwlprov/` and `cwlprov --help`, in particular we would like to point out that commands like `cwlprov inputs` and `cwlprov outputs` can use identifiers of individual steps (CWLProv level 1) and nested workflows (CWLProv level 2); these would be harder to represent in a pure file structure without significant storage duplication or creative use of symlinks. Options like `cwlprov runtimes` and `cwlprov derived` calculate secondary information on demand based on the PROV trace. Further work would be needed to build a more researcher-oriented interface based on this tool (e.g. hard-coded for a particular workflow).

We have added an explanation of this to the end of section **Evaluation results**.

Reviewer Point P 2.2 — In addition, the "pragmatic" interoperability should refer to top-level provenance and thus domain-specific annotations referring to the scientific context of the computational experiment. The experiments don't clearly show how CWLprov goes into the direction of (still ambitious and challenging) domain-specific provenance.

Response: The framework of provenance and CWLProv as a standard conceptually can achieve all three states of interoperability in principle. If we achieve Level 3 by recording domain-specific information and contextual knowledge about the experiment, data used, output produced and the methods employed in the process, we will be able to satisfy requirements of pragmatic interoperability. However, the current implementation/prototype using cwltool described in this paper has achieved up-to Level 2 and we are working (as described in section **Provenance Profile Augmented with Domain Knowledge**) to implement Level 3 practically, see `https://github.com/common-workflow-language/cwlprov/issues/2` for details.

In the manuscript we have now more clearly mentioned the state of practical implementation, future direction, the conceptual maturity of provenance framework and CWLProv and finally the state of pragmatic interoperability throughout the manuscript. Having addressed this comment, we believe that the remaining comments about pragmatic interoperability by reviewer are hopefully resolved. We will refer to this comment below in response to other pragmatic interoperability comments.

Reviewer Point P 2.3 — I've also a technical concern regarding the FAIRness of the approach since some of the requirements could be addressed following the (5-star) Linked Data principles. This point should be addressed in the discussion.

Response: We have added discussion about 5-star linked data principles in section **Levels of Provenance and Resource Sharing** (second last paragraph).

Reviewer Point P 2.4 — Finally, I tried to browse the research objects provided as supporting material but unfortunately I could not access the resource. Logs are provided at the end of the review.

Response: As indicated under **Availability of supporting data and materials** we have mirrored the research objects on Zenodo as well, in addition we contacted Mendeley Data to raise the accessibility issue.

Introduction

Reviewer Point P 2.5 — In Key Points, 4th point, space is missing in "CWLProvoutcome"

Response: Fixed

Background and related work

Reviewer Point P 2.6 — The first paragraph of related works is too long.

Response: We have removed some details about the existing studies by mentioning them briefly.

Reviewer Point P 2.7 — "co-installability" ->what does it mean?

Response: This term was describing how software package managers such as Conda and Debian help in managing installation of multiple versions of the same software or managing installation of a set of software required for a given analysis. However, while addressing the comment that the background information is too long, we have re-written this section "**Workflow Software Environment Capture**" and as a result no longer have this term.

Reviewer Point P 2.8 — Some references could be added to works addressing the sharing of domain-specific annotated provenance, for instance, https://doi.org/10.1186/2041-1480-5-28, https://doi.org/10.1016/j.websem.2014.07.001, or "From Scientific Workflow Patterns to 5-star Linked Open Data" in TaPP'16.

Response: We agree with the reviewer that these are related citations. We have added Clark et al. (2014) and Gaignard et al. (2016) in section **Provenance Capture & Standardization** and Gaignard et al. (2014) in section **Level 3**.

Levels of Provenance and resource sharing

Reviewer Point P 2.9 — "... in Figure 1 that all WMs can benefit from and conform to without additional technical overhead" ->difficult to believe that there is no technical overhead

Response: We have changed the statement by replacing "no technical overhead" to "minimum technical overhead". If these levels are kept in mind from the beginning while designing a new workflow or a new system, it is possible to achieve these levels with very little technical overhead.

Reviewer Point P 2.10 — Table 1 ->the list of recommendations is quite long, some of the recommendations are overlapping (R9 and R19 could be merged, as well as R6 and R7). Grouping them, possibly through the proposed levels could ease the reading and understanding of these recommendations. In addition, R18 is too vague.

Response: We agree with merging R6 and R7 together as both are dealing with workflow annotation. However, we still would like to keep R9 and R19 separate as one of them refers to software environment and the other (R19) refers to the hardware resources.

Taking reviewer's feedback into consideration, we have loosely clustered the related recommendations in different classes. However, clustering based on levels seems reverse engineering as the levels were derived from these recommendations.

Reviewer Point P 2.11 — Figure1 ->in Level 0 "Results interpretation is questionable"" scientists will need some context (Level 3) to understand the produced results, he/she may be lost in all fine-grained provenance, and extracting important parameters would certainly be time-consuming and require technical expertise.

Response: We agree with the reviewer that to make complete sense of the results, the end user must have some domain information provided with the results. However, given the expertise level of end-user, it is possible to inspect the results and hence partially interpret some aspects of it. Therefore, we have modified Figure 1 to add the term "Partial interpretation of results" instead of "Results interpretation".

Reviewer Point P 2.12 — R2, R13, R16-18 are not mentioned in the Levels 0-3 descriptions.

Response: With merging the previous R6 & R7, now these numbers are R2, R12, R15-17. We have included R12, R16 and R17 in section **Level 1**. We have added R15 in section **Level 0** as open licensing should be applied at the lowest level and hence is applicable to all the levels above.

Reviewer Point P 2.13 — Level 2 paragraph 2: Re-enactment ->this feature already exists in make-like systems, such as snakemake, actively developed and used in the bioinformatics community.

Response: We have acknowledged the fact mentioned by the reviewer and discussed in the same paragraph (section **Level 2**).

Reviewer Point P 2.14 — Level 2: "meaningful for a user" ->which kind of user?

Response: We have clarified this statement.

CWLProv 0.6.0

Reviewer Point P 2.15 — "we have reused the BDBag approach based on BagIt" ->a short example of a Bag would have been useful.

Response: We have added a box to explain BagIt and simplified the text.

Reviewer Point P 2.16 — "We utilise mainly two serialisations of PROV [...]" ->why not using PROV-O to ease the linking of provenance information to other datasets as well as its analytics through querying or logical reasoning. This would also enhance findability on the web. This point should be part of the discussion.

Response: We have expanded on how we generate several PROV-O serializations (Turtle, N-Triples. JSON-LD), and why we don't require all of these in other CWLProv implementations.

Reviewer Point P 2.17 — "workflow/" ->the paragraph on "executable workflow objects" is hard to follow.

Response: We have rewritten this paragraph to hopefully explain better the reasoning using examples.

Reviewer Point P 2.18 — "metadata/" ->the discussion on URI schemes is hard to follow, again an example would help.

Response: We have rewritten this paragraph to hopefully explain better the reasoning using examples.

Reviewer Point P 2.19 — "Retrospective provenance Profile" ->is the production of wfdesc / wfprov RDF data automatic or manual?

Response: The production of data about workflow provenance is automatic.

Practical realisation of CWLProv

Reviewer Point P 2.20 — Figure 5 ->what does "relativised job object" mean?

Response: We have replaced "relativised job object" by "relitivised file paths for inputs". It refers to the input configuration file with input data paths relative to the RO they are part of (instead of hard-coded file paths).

Reviewer Point P 2.21 — Figure 5 ->which steps are the most costly (time/space)

Response: With the current proof-of-concept implementation, copying input and output data in "Content addressable Input artefacts" and "Add content addressable outputs" step of Figure 5 will take the most time as well as space in case of large data files. A production quality implementation would not have these overheads.

CWLProv evaluation

Reviewer Point P 2.22 — CWLProv supports syntactic, semantic, and pragmatic ->since pragmatic refers to scientific context/claims, etc., it is unclear how pragmatic interoperability is addressed.

Response: We have addressed this comment in response to P 2.2.

Reviewer Point P 2.23 — Why choosing these 3 bioinformatics workflows, do they cover different aspects of the evaluation? Maybe a single in-depth description would be enough.

Response: We have added few lines describing the choice of these three workflows in section **CWLProv Evaluation with Bioinformatics Workflows**.

Reviewer Point P 2.24 — "In addition, the resource requirement" ->this is a good example for R19, a link to R19 would be useful here.

Response: Thanks for pointing it out, we have added reference to R19 where the reviewer suggested as *"In addition, the resource requirements (identified in R19-resource-use and [...]) should also be satisfied by choosing a system with enough compute and storage resources for successful enactment."*

Reviewer Point P 2.25 — The re-enactment scenario is clear as well as the provenance queries scenarios but the interoperability evaluation is less clear towards the "pragmatic assumption" and domain annotations.

Response: We have addressed this comment in response to P 2.2.

Reviewer Point P 2.26 — Temporal and spatial overhead ->For the RNAseq and Alignment workflows, the Prov overhead appears as quite noticeable. Which part of the process (Fig. 5) would explain this difference?

Response: In the current proof-of-concept implementation with cwltool, we are keeping a copy of input and output data in the research object. Copying the data files (happening at "Content addressable Input artefacts" and "Add content addressable outputs" stage in Fig. 5) which are larger in case of these two workflows (as compared to somatic workflow that is using small test data provided with the workflow) contributes to the time difference between the with and without provenance executions. This fact is mentioned in paragraph 3 of section **Temporal and Spatial Overhead with Provenance** and possible solutions leading to potential future directions described in section **Big -omics Data**. With the future directions implemented, we think there will not be any overhead with respect to this aspect of the process.

Discussion and Future Directions

Reviewer Point P 2.27 — "Selected jobs provenance": this paragraph is a bit confusing since the lack of completeness of provenance was identified as the main issue, it highlights that this complete capture approach may raise human-tractability issues.

Response: We have rewritten the paragraph (**Improving CWLProv efficiency with selective provenance capture**) to indicate that the main concern is storage inefficiency of keeping shim step outputs, we also added the reviewer's point that collapsing "boring" parts can improve human-tractability.

Reviewer Point P 2.28 — In addition, users can add domain-specific annotations to data ->How? how difficult/easy it is?

Response: We have clarified this statement as "In addition, users can add standardised domain-specific annotations to data and workflows incorporating the constructs defined by external ontologies (e.g. EDAM) to enhance understanding of the shared specification and the resources it refers to."

This is another point that would best suited by best-practice guides like `https://view.commonwl.org/about#format` to indicate where to add which annotations.