**Responses to Comments from Reviewer 1**

**NOTE:** The following comments are responded in the revised manuscript using **Magenta** coloured text and the same are copied here.

**Reviewer’s Detailed Comments:**

The authors proposed an ontology for workflow evolution and its provenance. They validated the proposed ontology through competency questions using SPARQL queries. Overall, I think the paper provides a good contribution to defining an ontology for the workflow evolution process, but I want to raise a few minor points.

**Response:**

We are thankful to the reviewer for the valuable suggestions and the recommendation to improve the quality of this work. We have responded to the suggestions in the following way:

**Comment 1:**

*- I found the use of ontology ProvONE to define a workflow unnecessarily complicated, and I felt that it would have been better to use a much simpler ontology like P-PLAN.*

**Response:**

We appreciate the comment.

P-PLAN is an ontology that defines scientific processes. Scientific workflows are a specialised form of scientific processes and have a particular structure. Therefore, the authors of P-PLAN extend it and proposed OPMW for scientific workflows.

We opted for ProvONE because it is the most widely accepted model for scientific workflows and aligns with the workflow structure in SenapsLAND. However, the evolution model we proposed in this paper (ref. Section 4.2) can be used in combination with P-Plan, OPMW and Wfdesc Ontology. This point is made clear in the revised draft.

“We reuse workflow specification pattern shown in Figure 5 from ProvONE and provide an additional layer to capture workflow evolution provenance. However, the modular framework opted for this ontology design allows reusing other workflow provenance ontologies, including P-Plan, OPMW and Wfdesc ontology, as part of WePROV ontology. In that case, the WePROV-Workflow will represent the workflow specification pattern from these ontologies.”

**Comment 2:**

*- It is also not clear how competency questions were answered. The corresponding SPARQL questions seem overly complex. A brief explanation will help clear up how queries would work. Is there a working server where these queries can be tested?*

**Response:**

We have provided a short description of each query in the file available online at <https://github.com/anilabutt/WePROV-Workflow-Evolution-Provenance-Ontology/blob/master/CompetencyQuestions/SPARQL-for-CQs.txt>

The ontology is available online and the link is included in the paper. This ontology can be loaded into any ontology editor like Protégé and a SPARQL queries can be tested on this ontology.

Currently, we don’t have a working server; however, we are in the process of implementing a provenance management service on top of this ontology. A link will be made available to the users once the service is ready to use.

**Comment 3:**

*- I also would like to see more practical competency questions such as "What part of the workflow has been changed", showing the deleted / added input, output parameters or steps.*

**Response:**

Thanks for the comment. We worked with the SWfMS team to identify a series of tasks that need to be supported by a useful workflow evolution provenance ontology and listed them as functional requirements for this ontology. However, ontology can answer many such questions. For example, the second competency question (i.e., **CQ2:**

**Identify different changes in a version of the workflow.**) address the reviewer suggested query (i.e., "**What part of the workflow has been changed**"). The corresponding query returns changes (added/deleted/modified component) in a workflow.

**Comment 4:**

*- In page 2 line 8, the authors stated, "But these solutions rely on proprietary formats that make it difficult to reuse these solutions." Please elaborate on this.*

**Response:**

The following text has been included in the revised draft. (see line 98-100 on Page 3)

“Only a few SWfMSs proposed solutions to automatically capture the limited amount of workflow evolution provenance. For instance, Kepler records evolution provenance (i.e., revision Id, time of the revision, and the user who submitted the revision) in a relational schema. VisTrails records a workflow evolution or execution action using an XML representation. Each evolution is considered a new version of the workflow, which has a unique id, name, optional annotation, set of actions, and macros. However, both these solutions capture limited evolution provenance and record them in proprietary formats (i.e., relational schema and XML). Therefore, it is difficult for other workflow engines to reuse these solutions or integrate and process the provenance from different solutions.”