PROCEEDINGS OF SCIENCE

Provenance model for Lattice QCD

Tanja Auge,[a] **Gunnar Bali,**[b] **Meike Klettke,**[a] **Bertram Ludäscher,**[c] **Wolfgang Söldner,**[b,*] **Simon Weishäupl**[b] **and Tilo Wettig**[b]

[a] *University of Regensburg, Faculty of Computer Science and Data Science, Germany*
[b] *University of Regensburg, Department of Physics, Germany*
[c] *University of Illinois at Urbana-Champaign, School of Information Sciences, USA*

E-mail: wolfgang.soeldner@ur.de

Like in many other research areas, workflow management becomes increasingly important in the community of Lattice QCD. Since for distinct observables high-precision results are mandatory nowadays, the analysis of the corresponding data becomes more and more complex. In this context we focus on the particular aspect of provenance tracking. We formulate a provenance model for Lattice QCD that includes the ensemble-generation and measurement parts of the Lattice QCD workflow following the W3C PROV standard. As many important provenance questions in our community require extensions of this model, we propose a multi-layered provenance approach that combines prospective and retrospective elements.

European network for Particle physics, Lattice field theory and Extreme computing (EuroPLEx2023)
11-15 September 2023
Berlin, Germany

*Speaker

PoS(EuroPLEx2023)038

1. Introduction and motivation

In today's state-of-the-art Lattice QCD (LQCD) simulations large amounts of data (i.e., many PetaBytes) are generated and analyzed. It is therefore useful to comply with certain rules when handling such data. Within the so-called *FAIR Guiding Principles for scientific data management and stewardship* such rules have been established [1] for a scientific context, where the abbreviation FAIR stands for *Findable*, *Accessible*, *Interoperable*, and *Reusable* (see Ref. [1] for a detailed discussion of these concepts). These guidelines are in particular relevant for LQCD in the context of handling data and metadata as well as workflow management. Note that the LQCD community has already taken steps towards implementing FAIR principles long before the term FAIR was established. The sharing of gauge-field configurations is organized within the *International Lattice Data Grid* (ILDG), an initiative which started about twenty years ago [2]. The current status of the project is presented in Ref. [3], and in Ref. [4] the ILDG metadata standard QCDml is described, which includes tracking information for ensemble generation. Another project named EspressoDB deals with systematic workflow and data management for LQCD [5]. In this contribution, we address the issue of workflow provenance and propose a provenance model for parts of the LQCD workflow [6].

The term *provenance*, in general, refers to information that describes the production process of an end product. In order to demonstrate the importance of provenance in LQCD we present a real-world example based on an incident that actually has happened. We have stored a set of configurations at an external research institute, where silent data corruption occurred on some configurations. If we had performed measurements on the corrupted configurations before the corruption was noticed, we would have obtained incorrect results. In this case, provenance information (e.g., checksums) is useful in two respects: On the one hand, provenance can identify the (downstream) measurement results affected by the corrupted configurations. On the other hand, provenance can trace the incorrect measurement results back (upstream) to the corrupted configurations.

Going beyond this example, there are a number of important questions that provenance can help answer in an automated way, such as

Q1 Which datasets are affected by an error or bug?

Q2 How are datasets affected by modifying a parameter?

Q3 Who was involved in generating the data?

Q4 Which codes and experts are needed to repeat a workflow?

Q5 Which data/parameters are needed to (re-) produce a result?

These and other questions address different aspects of provenance, as we will discuss below. This contribution is organized as follows. In Section 2 the W3C PROV standard is briefly described. In the following section this standard is used to construct a provenance model for LQCD workflows, and in Section 4 we propose an extension to the W3C PROV standard motivated by the requirements of LQCD. Note that this contribution is a summary of Ref. [6], where more details of our work are presented.

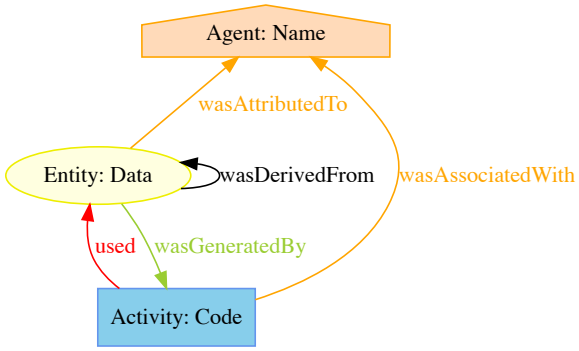


Figure 1: Basic concepts and relations of the W3C PROV standard (based on [7], figure from [6]).

2. W3C PROV concepts

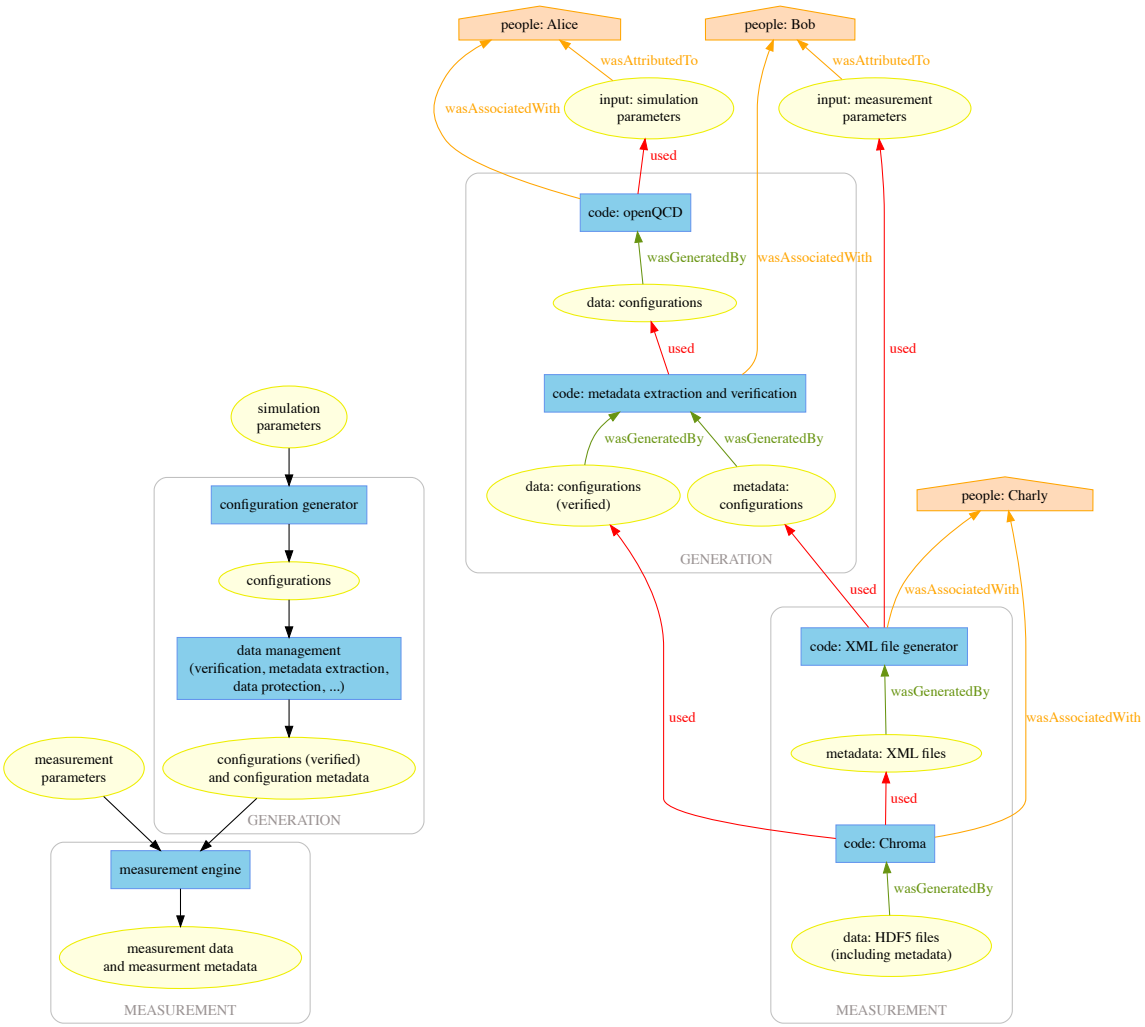
Our work is described within the W3C PROV standard [8, 9]. The basic features of this standard can be represented as a graph, which is shown in Fig. 1. There are three types of nodes in the graph: *entities* that can be derived from other entities, *activities* that generate or use entities, and *agents* that perform/control activities or produce entities. Examples are given for each case. The relations between the nodes are described by edges as shown in the figure. For example, an entity *wasGeneratedBy* an activity, or an activity *wasAssociatedWith* an agent. Note the conventions for the direction of the arrows.

In the next section we will describe LQCD workflows, where we distinguish two forms of workflow provenance: *retrospective provenance* describes information about past workflow executions and data derivations, while *prospective provenance* captures the structure of the workflow and provides a recipe for future workflow executions.

3. Lattice QCD workflows with and without provenance

The tasks that appear in generic LQCD calculations can be roughly divided into three classes: *generation* (of gauge-field ensembles), *measurement* (i.e., computation of correlation functions), and *analysis* (i.e., computation of observables). Here, we consider the first two parts, which are more compute-intensive and more generic than the analysis part. In Fig. 2a a generic LQCD workflow is visualized for the generation and measurement part, where the same shapes and color codes as in Fig. 1 are used to distinguish activities and entities. The simulation parameters consist of both physical parameters and algorithmic parameters, where often a subset of these parameters appears again in the set of measurement parameters. The activity labeled ‘‘data management’’ contains a number of steps. This includes data verification, backups and/or archiving the data, and processing the metadata. Note that the measurement part also includes data management, which is not shown to avoid cluttering the figure.

In the introduction we have addressed provenance-related questions such as **Q1** to **Q5**. In order to answer these questions we now need to add provenance information to our model. Note that we require different provenance elements to address the different questions. While questions **Q1** and **Q2** are more data-related, questions **Q3** to **Q5** are more workflow-related. In Fig. 2b we present our W3C PROV model that we have developed for the specific LQCD workflow used by our research



(a) Generic workflow without provenance. **(b)** Specific workflow formulated as a W3C PROV model.

Figure 2: Representation of the first two parts (generation and measurement) of a Lattice QCD workflow (both figures from [6]).

group. There are four activities (blue code boxes) that read the input parameters, manage the data, and generate the final HDF5 files. The seven entities (yellow boxes) represent input parameters, data, and metadata. In our case there are three agents (orange boxes) who execute the activities and are responsible for the input parameters. Note that we have dropped backup and archiving in the generation part in order to keep the figure simple. Also, we have split the measurement activity of Fig. 2a into two activities to better reflect the actual workflow.

Again, the directed edges describe the relationships between the agents, activities, and entities. However, we note that one edge type is missing. Since the flow of data entities is unique, w-edges (*wasDerivedFrom*) are simulated by a chain of u-edges (*used*) and g-edges (*wasGeneratedBy*).

We now investigate how our provenance model addresses the questions **Q1** to **Q5**. Questions **Q1** and **Q3** require retrospective provenance, questions **Q4** and **Q5** are prospective provenance queries, and question **Q2** combines both types of provenance. Question **Q3** can be answered directly from the

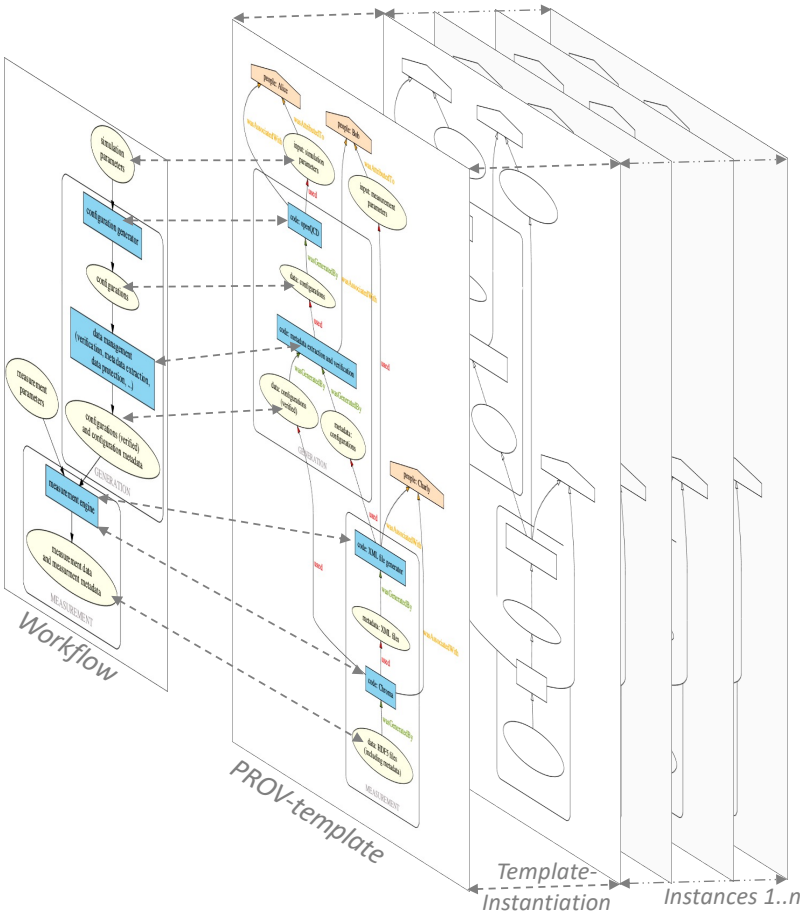


Figure 3: Proposed hybrid provenance model: The conceptual workflow (left) naturally maps to a W3C PROV template model (middle). Every workflow execution creates PROV-compatible retrospective provenance graphs (right), i.e., instances $1, \ldots, n$ of the provenance template in the middle (figure from [6]).

provenance graph using the name of the person. Similarly, questions **Q4** and **Q5** can be answered by inspecting the provenance graph and identifying the corresponding activities, entities, and agents. Question **Q1** is more complex. If we realize that the output data are incorrect, the data derivation chains must be traced from the results back to the sources (upstream propagation). If we find an error in an activity or entity, such as the silent data corruption mentioned earlier, the erroneous activity or entity needs to be fixed, and subsequent entities need to be recomputed or corrected (downstream propagation). In both cases we need to trace the dependencies along the derivation chains in the provenance graph. Finally, question **Q2** requires prospective provenance if we are only interested in the dependency structure at the conceptual level, while we need retrospective provenance if we are interested in the effect of a parameter change on a dataset in a previous workflow run.

4. Towards layered provenance

In Fig. 2a we have presented the generic workflow, which can be viewed as template. Neglecting further provenance information, the graph in Fig. 2b can be regarded as an instance of the graph

in Fig. 2a in which the generic elements are specialized to the activities and entities relevant to our research group. In the same fashion the provenance graph of Fig. 2b can be viewed as a *provenance template graph*. For each workflow execution an own provenance instance graph is generated, in which schema-level elements (e.g., $\mathrm{data{:}HDF5\ files}$) are replaced by references to concrete instance objects (e.g., $\mathrm{X251r000n1000\_run3.hd5}$).

These observations lead to the following proposal. The W3C PROV model needs to be extended to include both a *template-level* provenance graph and the corresponding *instance-level* provenance graphs including concrete values of all input parameters, names, timestamps, version numbers, etc. Furthermore, we propose a workflow layer from which the provenance template graph is derived. This proposal is visualized in the graph in Fig. 3. Any model based on this proposal should then meet the following requirements:

1. The community-level workflow structure should be linkable to provenance template graphs (research group level).
2. The instance-level provenance graphs of the hundreds or thousands of runs (with varying parameter settings, etc.) should be automatically linked to a provenance template.
3. A domain-aware provenance model should allow users to distinguish different types of data, e.g., using namespaces.

Based on this proposal, the next step is to develop a concrete implementation, where one can think of several possibilities. For the moment, we restrict ourselves to outline of what a possible LQCD prototype could look like. From existing LQCD workflows there is already a large amount of provenance information captured in log files. This information could be collected by a Python-based provenance harvesting tool and then deposited in a provenance store, i.e., a suitable database. In many cases provenance information can also be obtained from file or folder names. The remaining provenance information required by our model but not yet available through harvesting would have to be generated by, e.g., a light-weight provenance recorder, code instrumentation, or writing additional information to log files.

5. Summary and outlook

We are proposing a provenance model for the generation and measurement parts of the LQCD workflow based on the W3C PROV standard. From a generic LQCD workflow (see Fig. 2a) we have derived a PROV template (Fig. 2b), where we propose a layered model consisting of a workflow layer, a provenance template layer, and an instance layer (Fig. 3). This model allows us to answer many provenance-related questions such as **Q1** to **Q5**.

This is a first step to bring together the LQCD and provenance communities to define comprehensive standards for data management, including provenance metadata, and to co-develop W3C PROV extensions suitable for LQCD. We plan to refine our initial model proposal in the future and intend to implement a prototype that will allow us to evaluate its efficacy and practicality. Furthermore, we plan to apply our provenance model to the third part (analysis) of the overall LQCD workflow, which is less generic and more collaboration-specific.

Acknowledgments

We thank Sara Collins, Christoph Lehner, Nils Meyer, and Stefan Solbrig for stimulating discussions. This work was supported in part by DFG project 460248186 (PUNCH4NFDI).

References

[1] M.D. Wilkinson et al., *The FAIR Guiding Principles for scientific data management and stewardship*, *Scientific Data* **3** (2016) 160018.

[2] A.C. Irving, R.D. Kenway, C.M. Maynard and T. Yoshie, *Progress in building an International Lattice Data Grid*, *Nucl. Phys. B Proc. Suppl.* **129** (2004) 159 [hep-lat/0309029].

[3] F. Di Renzo, *The International Lattice Data Grid (ILDG 2.0)*, *PoS* **LATTICE2023** (2024) 112.

[4] C.M. Maynard and D. Pleiter, *QCDml: First milestone for building an International Lattice Data Grid*, *Nucl. Phys. B Proc. Suppl.* **140** (2005) 213 [hep-lat/0409055].

[5] C.C. Chang, C. Körber and A. Walker-Loud, *EspressoDB: A scientific database for managing high-performance computing workflows*, *J. Open Source Softw.* **5** (2020) 2007 [1912.03580].

[6] T. Auge, G. Bali, M. Klettke, B. Ludäscher, W. Söldner, S. Weishäupl and T. Wettig, *Provenance for Lattice QCD workflows*, *WWW '23 Companion: Companion Proceedings of the ACM Web Conference 2023* (2023) 1524 [2303.12640].

[7] Y. Gil et al., *PROV Model Primer*, 2013.

[8] L. Moreau et al., *The open provenance model core specification (v1.1)*, *Future Gener. Comput. Syst.* **27** (2011) 743.

[9] P. Groth and L. Moreau, *PROV-Overview*, 2013.