Design and Execution of make-like Distributed Analyses

R F von Cube, M Erdmann, B Fischer, R Fischer, M Rieger

III. Physics Institute A, RWTH Aachen University, Germany

E-mail: rfischer@physik.rwth-aachen.de

Abstract. In particle physics, workflow management systems are primarily used as tailored solutions in dedicated areas such as Monte Carlo production. However, physicists performing data analyses are usually required to steer their individual workflows manually, which is time-consuming and often leads to undocumented relations between particular workloads. We present a generic analysis design pattern that copes with the sophisticated demands of end-to-end HEP analyses. The approach presents a paradigm shift from executing parts of the analysis to defining the analysis. The clear interface and dependencies between individual workloads then enables a make-like execution.

Our tools allow to specify arbitrary workloads and dependencies between them in a lightweight and scalable structure. Further features are multi-user support, automated dependency resolution and error handling, central scheduling, and status visualization. The WLCG infrastructure is supported including CREAM-CE, DCAP, SRM and GSIFTP. Due to the open structure, additional computing resources, such as local computing clusters or Dropbox storage, can be easily added and supported. Computing jobs execute their payload, which may be any executable or script, in a dedicated software environment. Software packages are installed as required, and input data is retrieved on demand.

The management system is explored by a team performing ttbb and ttH cross section measurements.

1. Introduction

Modern high-energy physics data analyses are growing in complexity and scale, compare figure 1. Adding to the complexity are for instance the number of advanced analysis techniques like multivariate classifiers, sometimes requiring graphics processing units, the number of simultaneous analyses, e.g. for various final state channels or event categories. The scale increases as more data get recorded, and more simulated events are needed to describe the data with proper statistics. Issues of scale are typically addressed via high throughput computing concepts involving a large number of computing jobs and many computing sites, e.g via the Worldwide LHC Computing Grid (WLCG)[1]. The user is presented with the task to execute the analysis under these conditions. This involves a lot of bookkeeping to make sure that each required file is at the right place at the right time. Due to the complexity, often only the physicist knows the exact order in which certain scripts need to be executed, which files need to be copied, or where to perform the analysis. New team members or students are often challenged to learn many technical details on how to run the analysis. Finally, if people leave a group, they often take key knowledge with them. The focus of analysis documentation in publications or

conference talks is mostly on what was performed, not on how it was done on a technical level. These factors contribute to making it difficult to extend or replicate an existing analysis.

In this paper, we explore the application of workflow management systems for user analyses. If the user defines what is the input data, what does the input depend on, and how it should be processed, the system should be able to execute the task. The goal is to run the analysis `make`-like [2] in a distributed manner.

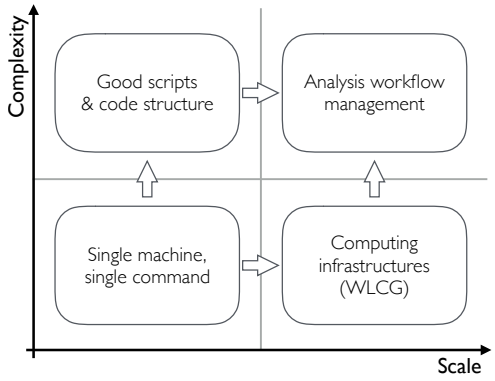

Figure 1. The figure shows different coping mechanisms for user analysis challenges with respect to their scale and their complexity.

2. Workflow Systems in High-Energy Physics

A data analysis comprises certain steps. Typical examples of analysis steps in high-energy physics are event selection, training of classifiers, statistical inference, plotting, or auxiliary estimations of scale factors or other relevant information. Often one part of the analysis consists of a number of steps. An example could be a data driven background estimation that first needs to create data tuples with relevant information, then performs a fit in a sideband, and subsequently creates control plots.

In more general terms, this can be described as a workflow, i.e. one or multiple related data analyses, that comprise smaller workloads. The workflows are related via a clear interface. For example, all dependencies have to be run first and the output of prior workloads needs to be accessible by subsequent workloads. In order to execute a workload, computing resources are needed, in particular, computing and storage resources. This perspective on data analyses shows how analogous it is to workflow management systems.

Figure 2. Typical Monte Carlo event simulation workflow.

Therefore, it is natural to look for existing workflow management systems related to high-energy physics. The typical example that can be found is illustrated in figure 2. It shows a chain from event generation to the event reconstruction that is typically used to created data sets of simulated collision events. A couple of observations can be made from these workflows. The type of depicted simulation workflow is one-dimensional, static, and recurring. Except for variations in certain steps of the chain, i.e. a alternative event generators, the software requirements are

homogeneous. Because simulation is so important to analyse recorded data and because the whole generation requires enormous computing and storage resources, special infrastructures are built for it. Among these are databases, storage systems, workload management systems.

User analyses on the other hand are not one-dimensional. One workload might need input from several others and the output might be needed by multiple subsequent workloads. They are not static since they need to adapt to newly discovered challenges during the analysis, to new recipes developed within large collaborations, and to requirements made during a review process. The scale of analyses is much smaller than for event simulation, while the requirements are more diverse. In summary, existing workflow management systems are not applicable to user data analyses.

3. Workflow Management for User Analyses

In this section two approaches to create workflow management systems for user analyses are presented. The first one is the report-based approach. It was tailored from scratch for high-energy physics. The main idea is that each time any payload is executed a report with metadata, i.e. regarding created output files, is created. This system should perform well if the storage system is slow especially with respect to file lookups.

The seconds approach is target based. Instead of creating a reports with metadata, the system checks the existence of necessary input files prior to scheduling a workload. This system should perform well for quick access to storage. In addition, this approach was chosen to test whether existing workflow management systems can be adapted to the high-energy physics environment. It is based on the Luigi pipelining tool[3].

Both systems are written in Python and will be presented in the following sections.

3.1. Report-based Workflow Management

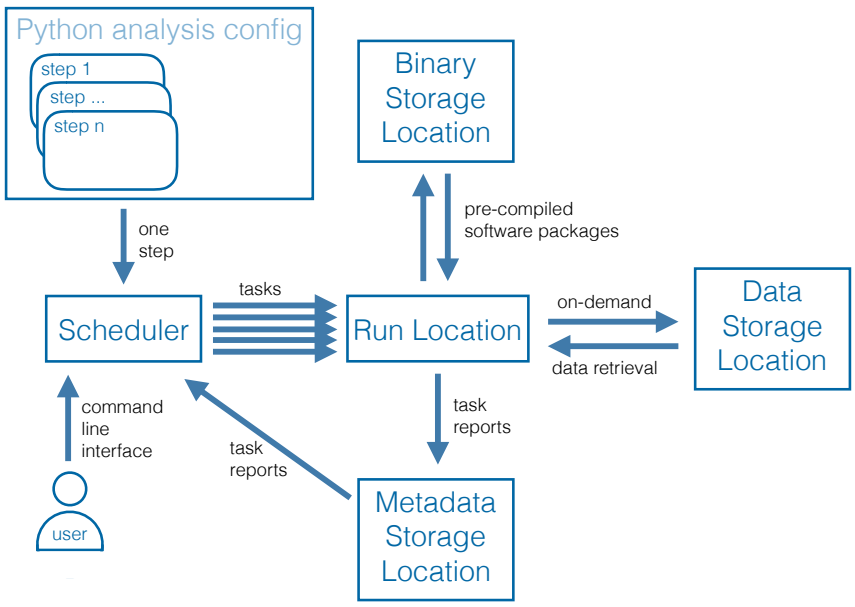

Figure 3. Scheme depicting the interplay between various components of the report-based approach in order to execute an analysis step.

The execution scheme is shown in Figure 3. Each step is executed in one or multiple tasks. The Scheduler is the component that determines which tasks are unfinished. Unfinished tasks

are sent to the Run Location configured for this step. Multiple tasks can be combined into one computing job if the user reckons the execution duration for one task is too short for one job. Once the job starts at the Run Location, the software environment is created using setup scripts and the Software Manager. After the environment is set up, the required input data is downloaded. If all required software and files are available the payload is executed. After a task is finished, the step stores metadata and provenance information in JSON [4] task reports on a Storage Location for meta data.

The next time the Scheduler identifies unfinished tasks it first downloads newly created task reports. Reports contain information on whether a task was performed successfully and if not what was the cause of the error. This allows to resubmit unsuccessful tasks already while the computing job is still running on other tasks. They also contain information on output Data Collections that were created including information on Storage Locations of output files. Subsequent steps can use these collections as their input Data Collection such that the Scheduler can schedule the execution of appropriate tasks for the subsequent step.

Because steps can spawn tens of thousands of tasks, reports are summarized into one file. Thereby, each task report only has to be read once. An example for useful metadata that can be evaluated by subsequent steps is the number of originally generated events for samples containing simulated events. This number can be used later to calculate event weights and to scale the samples appropriately.

The user can specify whether dependent steps need to be finished entirely or whether subsequent steps can already work with parts of the output data. For example, not all tasks of an event selection step have to be done for a following reconstruction step to start running on parts of the data. A step performing a statistical inference on the other hand might need all available reconstructed data to produce proper results.

3.2. Target-based Workflow Management

The target-based approach is depicted in figure 4. It shares many similarities with the report-based system. The main difference is that it does not create metadata reports. This reduces the complexity of the system because it does not need to manage reports or summarize them. The reduction in complexity is traded off against lookups whether required input files exist. The number of files that need to be checked can be of the order of tens of thousands.

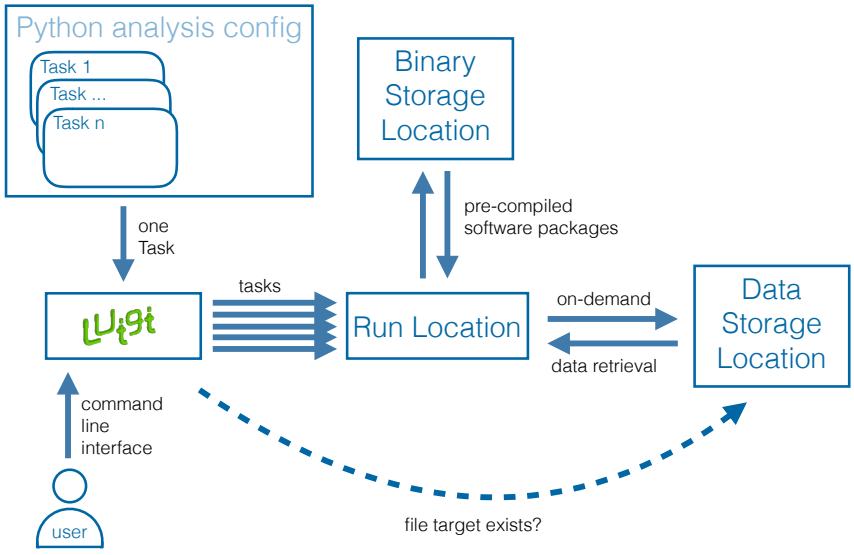

Figure 4. Schematic illustration of the target-based approach.

4. Conclusion

Both Workflow Management system presented here were tested alongside cross section measurements at the LHC for the top quark pair associated bottom quark pair production and the top quark pair associated Higgs production, respectively. Despite different requirements regarding central storage for data and metadata, both approaches worked well in conjunction with dCache system[5]. Many small reports files were handled equally well as the large number of file lookups.

Both systems provide building blocks for user analyses without limitations on the payload software that is used. Rather than providing an analysis framework, they can be understood as a design pattern for user analyses. In both cases all parameters needed for analyses are transparently encoded and thereby documented within the workflow descriptions. Thus, bookkeeping tasks and undocumented states of files and intermediate steps are prevented. This enables to reproduce the exact same results if the workflow is re-executed. From the point of view of the user, the systems allow to focus on physics topics rather than on technicalities of the execution. In summary, workflow management systems for user analysis present a paradigm shift from executing the analysis to defining it.

Acknowledgments

This work is supported by the Ministerium für Wissenschaft und Forschung, Nordrhein-Westfalen, the Bundesministerium für Bildung und Forschung (BMBF) and the Helmholz Alliance Physics at the Terascale.

References

[1] Bird I 2011 *Annual Review of Nuclear and Particle Science* **61** 99–118

[2] Free Software Foundation, Inc 2016 GNU Make URL `https://www.gnu.org/software/make`

[3] Spotify Luigi URL `https://github.com/spotify/luigi`

[4] 2013 The JSON Data Interchange Format Tech. Rep. Standard ECMA-404 1st Edition / October 2013 ECMA URL `http://www.ecma-international.org/publications/files/ECMA-ST/ECMA-404.pdf`

[5] Fuhrmann P, Behrmann G, Bernardt C *et al.* 2012 dCache, agile adoption of storage technology (Computing in High Energy and Nuclear Physics, New York City (USA), 21 May 2012 - 25 May 2012) URL `https://bib-pubdb2.desy.de/record/140491`