FINCoS: Benchmark Tools for Event Processing Systems

— Invited Demo Paper —

Marcelo R. N. Mendes, Pedro Bizarro, and Paulo Marques
CISUC, University of Coimbra, Dep. Eng. Informática – Pólo II, Coimbra, Portugal
{mnunes, bizarro, pmarques}@dei.uc.pt

ABSTRACT

FINCoS is a set of benchmarking tools for load generation and performance measuring of event processing systems. It leverages the development of novel benchmarks by allowing researchers to create synthetic workloads, and enables users of the technology to evaluate candidate solutions using their own real datasets. An extensible set of adapters allows the framework to communicate with different CEP engines, and its architecture permits to distribute load generation across multiple nodes. In this paper we briefly review FINCoS, introducing its main characteristics and features, and discussing how it measures the performance of event processing platforms.

Categories and Subject Descriptors

C.4 [Performance of Systems]: Measurement techniques

General Terms

Experimentation, Measurement, Performance.

Keywords

Benchmarking, CEP, Event Processing, Framework, Tools.

1. INTRODUCTION

Recent years have witnessed the consolidation of *Complex Event Processing* (CEP) as an important research discipline and industrial trend. There is still, however, a lack of information about the performance of CEP engines as standard benchmarks are currently not available. The FINCoS framework aims at filling this gap by providing a flexible and neutral approach through which users, researchers and engineers can quickly run realistic performance tests on one or more event processing platforms without having to code themselves load generation, performance measurement and event conversion routines.

FINCoS development started in the end of 2007, as part of the BiCEP project [1]. At that time, there was very little information on how the CEP technology was being used in the real world and a great diversity of products, each with their own languages and implementation styles, which posed serious challenges for the development of novel benchmarks. Our goal then was to provide a flexible tool that allowed carrying out performance evaluation on CEP platforms independently on their structural differences or the workload employed. This would be beneficial for both users of the technology, which would be able to better evaluate the performance of their candidate platforms, and the academic community, which could more easily develop and experiment novel benchmarks for event processing systems.

The first version of the framework was released in 2008 [3], and since then it has been considerably extended and improved. Today, FINCoS can be used not only for stand-alone performance studies, but also as reusable load generation and measurement component in upcoming benchmarks kits. At the moment, the framework is in the final stages of the review process for being included in the SPEC Research Group tools repository [4].

2. OVERVIEW AND CORE FEATURES

The FINCoS framework is composed by five main components. *Drivers* simulate external event sources, submitting load to the system under test (SUT). On the opposite side, *Sinks* receive the results produced by the SUT, storing them in log files for subsequent answer validation and performance measurement. The communication with the CEP engine in both cases is made through an extensible set of *adapters*, which convert the events produced by the framework into a data format understood by the target system and vice-versa. A graphical application, denominated *Controller*, allows users to configure, execute, and monitor performance tests. The results of these performance runs can then be visualized both in real-time and after test completion, using the *Performance Monitor* component.

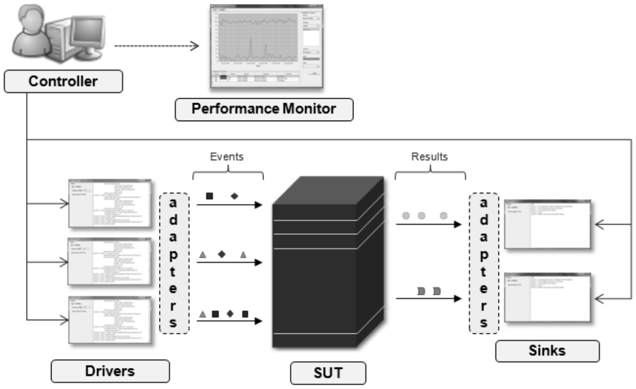

Figure 1: Overview of FINCoS components

FINCoS provides great flexibility and a wide range of options in the definition of experimental evaluations. For instance, the execution of *drivers* can be split into phases, each with its own

ICPE'13, April 21–24, 2013, Prague, Czech Republic.
ACM 978-1-4503-1636-1/13/04.

workload characteristics (e.g., event submission rate, types and datasets). This is useful not only for breaking performance tests into well-defined parts (e.g., warm-up and measurement interval) but also for evaluating the ability of event processing platforms in adapting to changes in the load conditions. In addition, users can choose if events should be generated by the framework itself or read from files containing real-world event data. The former shall be useful for researchers developing novel benchmarks while the latter should help customers trying to mimic their environments. The workload can also be seamlessly scaled by simply adding more *drivers* and *sinks* to the configuration.

Besides enabling users to define arbitrarily complex and realistic workloads, the framework was also designed to be portable across different CEP products. FINCoS allows running performance tests with any CEP engine that is able to exchange events through a JMS middleware. In addition, the framework supports direct communication with event processing platforms through custom adapters (using products APIs)[1].

3. PERFORMANCE MEASUREMENT

After test completion, the performance of the system under test is measured using the *performance monitor* application and the log files produced by *sinks* (the framework also allows measuring performance while tests are running at the cost of a slight overhead). The tool presents performance stats in both tabular and graphical formats – the former displays a snapshot of throughput and latency for each query running at the SUT, while the latter shows the evolution of these metrics over time.

Response time is measured by computing the difference between the time the SUT emitted a given result and the timestamp of the incoming event that triggered it. For that, output tuples produced by the CEP engine must explicitly include the timestamp of the causer event (the timestamp of the result itself is automatically collected and appended by the framework).

Since there is a great variance in the way the several event processing platforms operate, FINCoS provides some flexibility for computing response time. Figure 2 illustrates the three possible definitions of response time supported by the framework.

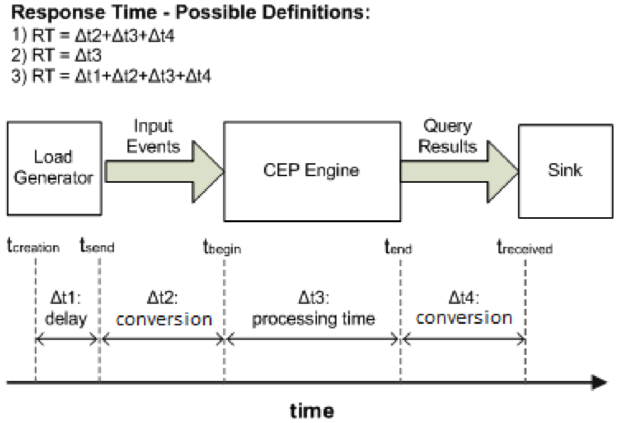

Figure 2: Latency measurement modes supported by FINCoS

The first, which we denominate *end-to-end* latency, represents the time it takes for an output tuple to arrive at a *sink* after the corresponding event that triggered it is sent by a *driver*. Note that in this definition the time for converting the event from the internal representation of the framework to a format understood by the SUT (and vice-versa) is accounted as part of the response time. Alternatively, the second definition can be used if the user wants to measure only the *processing time* of events inside the CEP engine. In this case, the events are timestamped inside the *adapters*, immediately before and after sending and receiving events to the SUT. A third option is available and is intended for accounting for delays introduced when the dispatch of events blocks on their processing at CEP engine (the framework then uses the event scheduled time instead of a measured time).

In addition to the point where events are timestamped, FINCoS allows users to choose the resolution used to compute latency – either milliseconds or nanoseconds. Generally, the nanoseconds resolution should be preferred as it is more accurate and many CEP systems offer sub-millisecond processing latencies. Note, however, that response times can be measured in nanoseconds only if *drivers* and *sinks* run in the same machine. It also incurs in more overhead than a millisecond resolution, so users should balance the need for accurate response time measurement and high event submission rates.

4. SUMMARY

The FINCoS framework provides a starting point for users and researchers who desire to carry out performance evaluations on event processing platforms. Fully-customizable workloads can be configured and tests can be performed on virtually any event processing platform, via standard JMS middlewares or directly, through customized adapters. These two characteristics are also important for the development of novel benchmarks, as FINCoS can be reused as a portable component in multiple benchmark kits – this shall be the case for the benchmarks we are currently developing in the scope of the BiCEP project [1].

As future work, we plan to extend FINCoS to support automation of performance tests (the graphical nature of the framework greatly facilitates the definition and monitoring of experiments, but might become troublesome for large experiment sets). We also plan to extend performance measurement, collecting system-level metrics, such as memory consumption and CPU utilization, in addition to the already provided response time and throughput.

Acknowledgements. This research has been supported in part by the Portuguese Science and Technology Foundation (FCT), under grant Nº 45121/2008.

5. REFERENCES

[1] BiCEP project web site: http://bicep.dei.uc.pt

[2] FINCoS web site: http://code.google.com/p/fincos/

[3] Mendes, M.R.N., Bizarro, P., Marques, P. A Framework for Performance Evaluation of Complex Event Processing Systems. In Proc. of the 2nd Conf. on Distributed Event-Based Systems (Rome, Italy, July 2008), 313-316. DOI= http://dx.doi.org/10.1145/1385989.1386030.

[4] SPEC Research Group tools repository: http://research.spec.org/tools.html

[1] FINCoS is distributed with out-of-the-box support for the open-source engine Esper.