Performance Evaluation and Benchmarking of Event-Based Systems

SPEC Distinguished Dissertation Award 2011 (Invited Abstract)

Kai Sachs
SAP AG, Germany
kai.sachs@sap.com
Thesis Supervisor:
Alejandro Buchmann, TU Darmstadt, Germany
buchmann@dvs.tu-darmstadt.de

Categories and Subject Descriptors
C.4 [Computer Systems Organization]: Performance of Systems—Modeling techniques, Design studies, Measurement techniques

Keywords
Benchmark, Event-based systems, Message-oriented middleware, SPECjms2007

Event-based systems (EBS) are increasingly used as underlying technology in many mission critical areas and large-scale environments, such as environmental monitoring and location-based services [3]. Moreover, novel event-based applications are typically highly distributed and data intensive with stringent requirements for performance and scalability. Since their reliability is crucial for the whole IT infrastructure, a certain Quality-of-Service (QoS) level has to be ensured. The motivation for our work was to support the development and maintenance of EBS that meet their QoS requirements. Given that EBS differ from traditional software in fundamental aspects such as their underlying communications paradigm, specific solutions and concepts are needed. System architects and deployers need tools and methodologies, which allow them to evaluate and forecast system performance and behavior in certain situations to identify potential performance problems and bottlenecks. Common approaches are benchmarking and performance modeling. However, no general performance modeling methodologies focusing on EBS had been published. Furthermore, there was a lack of test harnesses and benchmarks using representative workloads for EBS. Consequently, we focused on the development of a performance modeling methodology of EBS as well as on approaches to benchmark them. We summarize now our main contributions and proposed approaches.

To comprehend our contributions, an understanding of the fundamental ideas of EBS is essential. Generally, EBS are software systems in which an observed event triggers a reaction. For more details and definitions of events and related concepts refer to [3]. We evaluated the variety of underlying technologies with a focus on distributed EBS and message-oriented middlewares (MOMs) and provided a survey of products and standards [7]. In our review of existing work, we identified a lack of benchmarks and performance modeling approaches for EBS [5]. To support a structural evaluation of benchmarks, we introduced five categories of requirements [9]: (i) Representativeness: the benchmark has to be based on a representative workload. (ii) Comprehensiveness: exercise all platform features typically used in applications. (iii) Focus: place the emphasis on the technology server and minimize the impact of other services, e.g., databases. (iv) Configurability: provide a configurable tool for performance analysis. (v) Scalability: provide ways to scale the workload in a flexible manner.

None of the existing benchmarks met all our requirements. Therefore, we saw a strong need for independent and standardized benchmarks for EBS fulfilling the requirements. To address this need we developed the first industry standard benchmark for EBS jointly with the SPEC (Standard Performance Evaluation Corporation). As underlying technology platform we chose Java Message Service (JMS). This was motivated by the fact that MOMs are widely used in industry and the quasi-standard for MOMs is JMS. Our efforts resulted in the SPECjms2007 standard benchmark [9]. Its main contributions were twofold: based on the feedback of industrial partners, we specified a comprehensive standardized workload with different scaling options and implemented the benchmark using a newly developed complex and flexible framework.

Using the SPECjms2007 benchmark we introduced a methodology for performance evaluation of message-oriented middleware platforms and showed how the workload can be tailored to evaluate selected performance aspects [10]. We demonstrated our methodology in a case study of a leading JMS platform and conducted in-depth performance analyses of the platform for a number of different workload and configuration scenarios.

The SPECjms2007 business scenario was specified independently from the underlying technology. Therefore, its usage is not limited to a specific type of EBS. We illustrated how the standardized workload can be applied to other EBS using the example of jms2009-PS, a benchmark for publish/subscribe-based communication [8]. This benchmark provides a flexible framework for performance analysis with a strong focus on research. The proposed benchmarks are now the de facto standard benchmarks for evaluating
messaging platforms and have already been used success-
fully by several industrial and research organizations as a
basis for further research on performance analysis of EBS.

To the best of our knowledge, no work introducing a gen-
eral methodology for modeling EBS had been published. As
a consequence, we investigated whether and how traditional
performance modeling approaches are suitable to model the
specifica of EBS. We introduced a formal definition of EBS
and their performance aspects, which allows us, e.g., to de-
scribe workload properties and routing behavior in a struc-
tured way [6]. Resulting from our analysis of existing mod-
ing techniques, we proposed an approach to characterize
the workload and to model the performance aspects of EBS.
We used operational analysis techniques to describe the sys-

tem traffic and derived an approximation for the mean event
delivery latency. We then showed how more detailed per-
formance models based on queueing Petri nets (QPNs) [1] could
be built and used to provide more accurate performance pre-
diction. We chose QPNs as performance modeling technique
because of their modeling power and expressiveness. Our ap-
proach allows evaluating and predicting the performance of
an EBS and provides detailed system models. It can be used
for an in-depth performance analysis and to identify poten-
tial bottlenecks. A further contribution is a terminology for
performance modeling patterns targeting common aspects
of event-based applications using QPNs [7, 11].

To additionally improve the modeling power of QPNs, we
suggested several extensions of the standard QPNs, which
allow building models in a more flexible and general way
and address several limitations of QPNs [7]. By introducing
a level of abstraction, it is possible to distinguish between
logical and physical layers in our models. This enables to
flexibly map logical to physical resources and thus makes
it easy to customize the model to a specific deployment.
The different layers allow reusing one logical model in sev-
eral physical models or to map several logical models to one
physical model. Furthermore, we addressed two limiting as-
pcts of standard QPNs: constant cardinalities and the lack
of transition priorities. By introducing non-constant cardi-
nalities of transitions we increased the modeling flexibility
and minimized the number of transition modes. The miss-
ing support of transition priorities in standard QPNs lim-
its the control of the transition firing order. We addressed
this restriction by incorporating priorities for transitions into
QPNs and discussed several ways to implement them. Our
extensions were integrated in the QPME / SimQPN software
tools [4] or are planned for the upcoming release.

Finally, we validated the approach in two case studies.
We applied our methodology to model EBS and predicted
their performance and system behavior under load success-
fully. As part of the first case study we extended SIENA,
a well-known distributed EBS [2], with a runtime measure-
ment framework and predicted the runtime behavior includ-
ing delivery latency for a basic workload [6] with a sin-
gle event type. In the second case study, we developed a
comprehensive model of the complete SPECjms2007 work-
load including the persistent layer, point-to-point and pub-
lish/subscribe communication [11]. To model the workload
we applied our performance modeling patterns as well as our
proposed QPN extensions.

We evaluated its accuracy in a commercial middleware
environment. To validate our modeling technique we in-
vestigated deployments of the benchmark in representative
environments comparing the model predictions against mea-
surements on the real systems. A number of different sce-
narios with varying workload intensity (up to 30,000 mes-
ges / 4,500 transaction per second) and interaction mixes
were taken into account. By means of the proposed mod-
els we were able to predict the performance accurately. No
models of realistic systems of the size and complexity of the
one considered in this case study exist in the literature.

Both case studies demonstrated the effectiveness and prac-
ticality of the proposed modeling and prediction methodol-
gy in the context of a real-world scenario. The advan-
tage of the approach is that it is both practical and general,
and it can be readily applied for performance evaluation of
distributed EBS and MOM. The technique can be ex-
ploded as a powerful tool for performance prediction and
capacity planning during the software engineering lifecycle
of message-oriented event-driven systems.

Our results open up new avenues of research in the area of
event-based systems. Our performance modeling methodol-
gen can be used to build self-adaptive EBS using automatic
model extraction techniques. Such systems could dynami-
cally adjust their configuration to ensure that QoS require-
ments are continuously met.

1. REFERENCES

[1] F. Bause. QN + PN = QPN - Combining Queueing
    Networks and Petri Nets. Tech. report no.461, Dept.
    of CS, University of Dortmund, Germany, 1993.
    and Evaluation of a Wide-Area Event Notification
    applications and enabling technologies. In Proc. of the
    Modeling Tool Based on Queueing Petri Nets. ACM
    performance modeling of event-based systems. In -
    A methodology for performance modeling of
distributed Event-Based systems. In Proc. of the IEEE
    ISORC, 2008.
    Event-Based Systems. PhD thesis, TU Darmstadt,
    2010.
    Benchmarking publish/subscribe-based messaging
    systems. In DASFAA 2010 - Int. Workshops:
    Workload characterization of the SPECjms2007
    Performance evaluation of message-oriented
    middleware using the SPECjms2007 benchmark.
    modeling and analysis of message-oriented
    event-driven systems. Software and Systems Modeling,
    2012. DOI: 10.1007/s10270-012-0228-1.