Performance Management Group

fortiss

München, 2015-11-06

Modeling Big Data Systems by Extending the Palladio Component Model

6th Symposium on Software Performance (SSP) 2015

Johannes Kroß[1], Andreas Brunnert[1], Helmut Krcmar[2]

[1] fortiss GmbH, [2] Technische Universität München

fortiss GmbH
An-Institut Technische Universität München

Agenda

- Motivation
- Development Process and Characteristics of Big Data Systems
- Palladio Component Model (PCM) Meta-model Extension
- Related Work
- Conclusion and Future Work

Agenda

- Motivation
- Development Process and Characteristics of Big Data Systems
- Palladio Component Model (PCM) Meta-model Extension
- Related Work
- Conclusion and Future Work

Motivation

- Various big data technologies with different characteristics
- Casado and Younas (2015) list two main techniques that are common for big data systems, namely, batch and stream processing

Motivation

- The added value of big data systems for organizations depends on the performance of such systems (Barbierato et al. 2014)
- Performance models allow for proactive evaluations of these systems
- Existing performance meta-models for big data systems, however, focus on either ...
  - … one processing paradigm such as stream processing
    e.g., Ginis and Strom (2013)
  - … or one technology such as Apache Hadoop MapReduce
    e.g., Ge et al. (2013)
- We propose a general performance meta-model to specify shared characteristics of big data systems

Agenda

- Motivation
- Development Process and Characteristics of Big Data Systems
- Palladio Component Model (PCM) Meta-model Extension
- Related Work
- Conclusion and Future Work

Development Process of Big Data Systems

Component developers

- Batch processing (e.g., using Apache MapReduce)

```
public void map(Object key, Text value, ..)..{
  StringTokenizer itr = new StringTokenizer(value.toString());
  while (itr.hasMoreTokens()) {
    word.set(itr.nextToken());
    context.write(word, one); } }
```

```
public void reduce(Text key, Iterable<IntWritable> values,..)..{
  int sum = 0;
  for (IntWritable val : values) {
    sum += val.get(); }
  result.set(sum);
  context.write(key, result); }
```

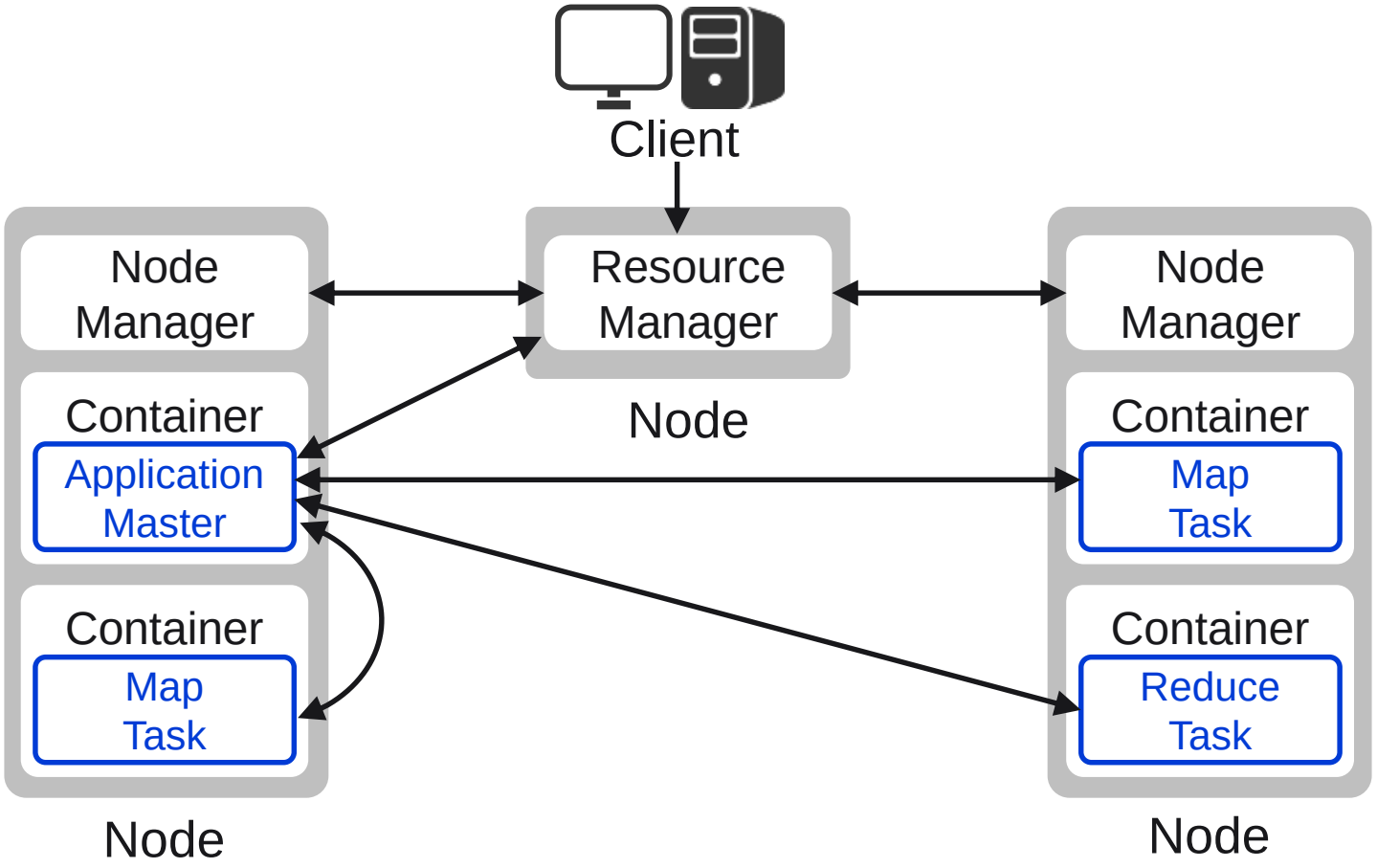
- Stream processing (e.g., using Apache Storm)

```
public void execute(Tuple tuple, BasicOutputCollector collector) {
  String word = tuple.getString(0);
  Integer count = counts.get(word);
  if (count == null) count = 0;
  count++;
  counts.put(word, count);
  collector.emit(new Values(word, count)); }
```

Development Process of Big Data Systems

System deployers

- Resource environment (e.g., Apache YARN)



Characteristics of Big Data Systems

- We derive the following requirements of big data systems that we propose to implement based on the finding of previous work (Kroß et al. 2015)

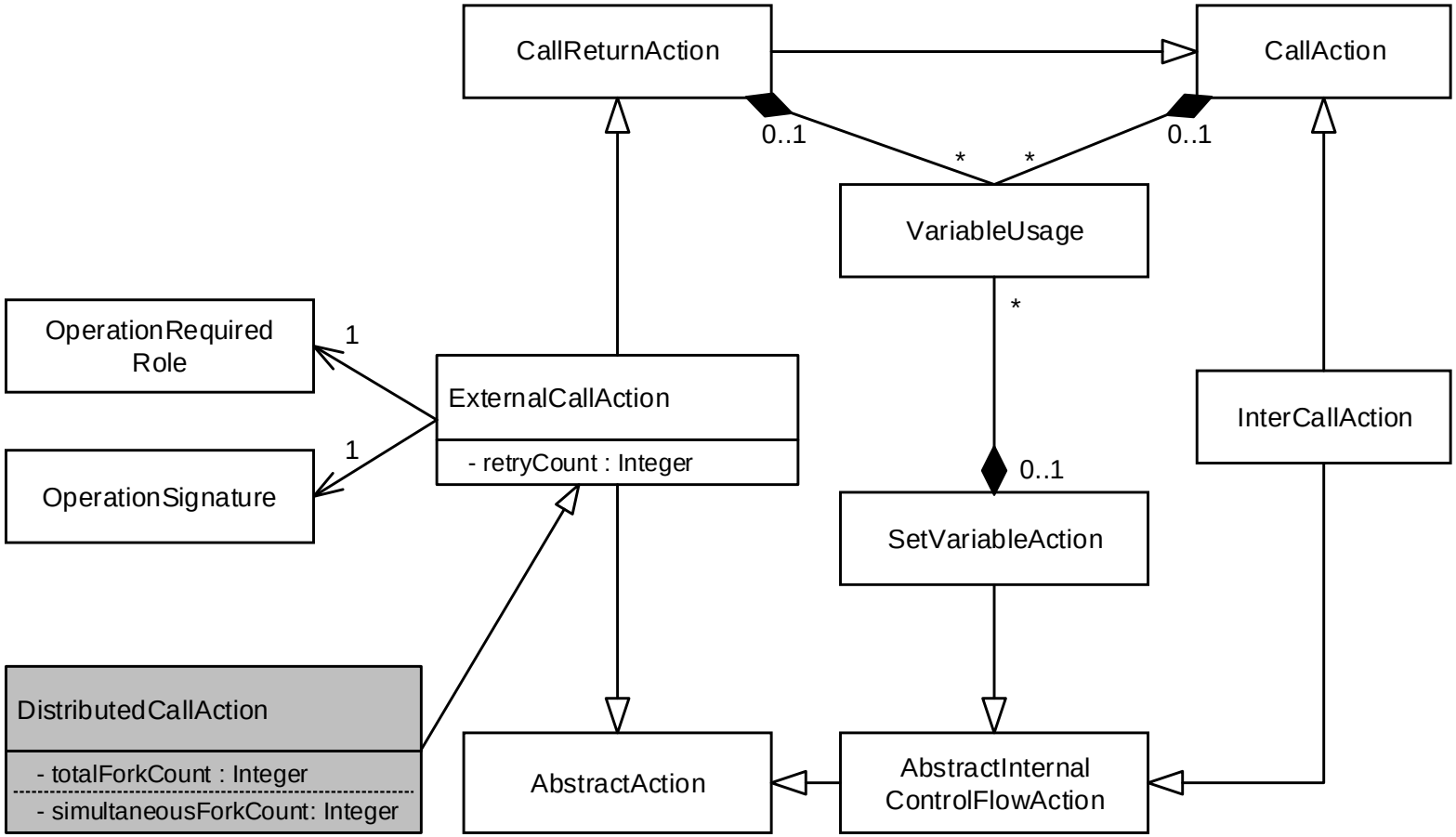
1. **Distribution and parallelization of operations**
   - Component developers specify reusable software components consisting of operations using software frameworks like Apache Spark.
   - In doing so, they may specify, but also may not know the definite number of simultaneous and/or total executions of an operation.
2. **Clustering of resource containers**
   - System deployers specify resource containers with resource roles (e.g., master or worker nodes), link them to a mutual network and logically group them to a computer cluster.

Agenda

- Motivation
- Development Process and Characteristics of Big Data Systems
- Palladio Component Model (PCM) Meta-model Extension
- Related Work
- Conclusion and Future Work

PCM Meta-model Extension

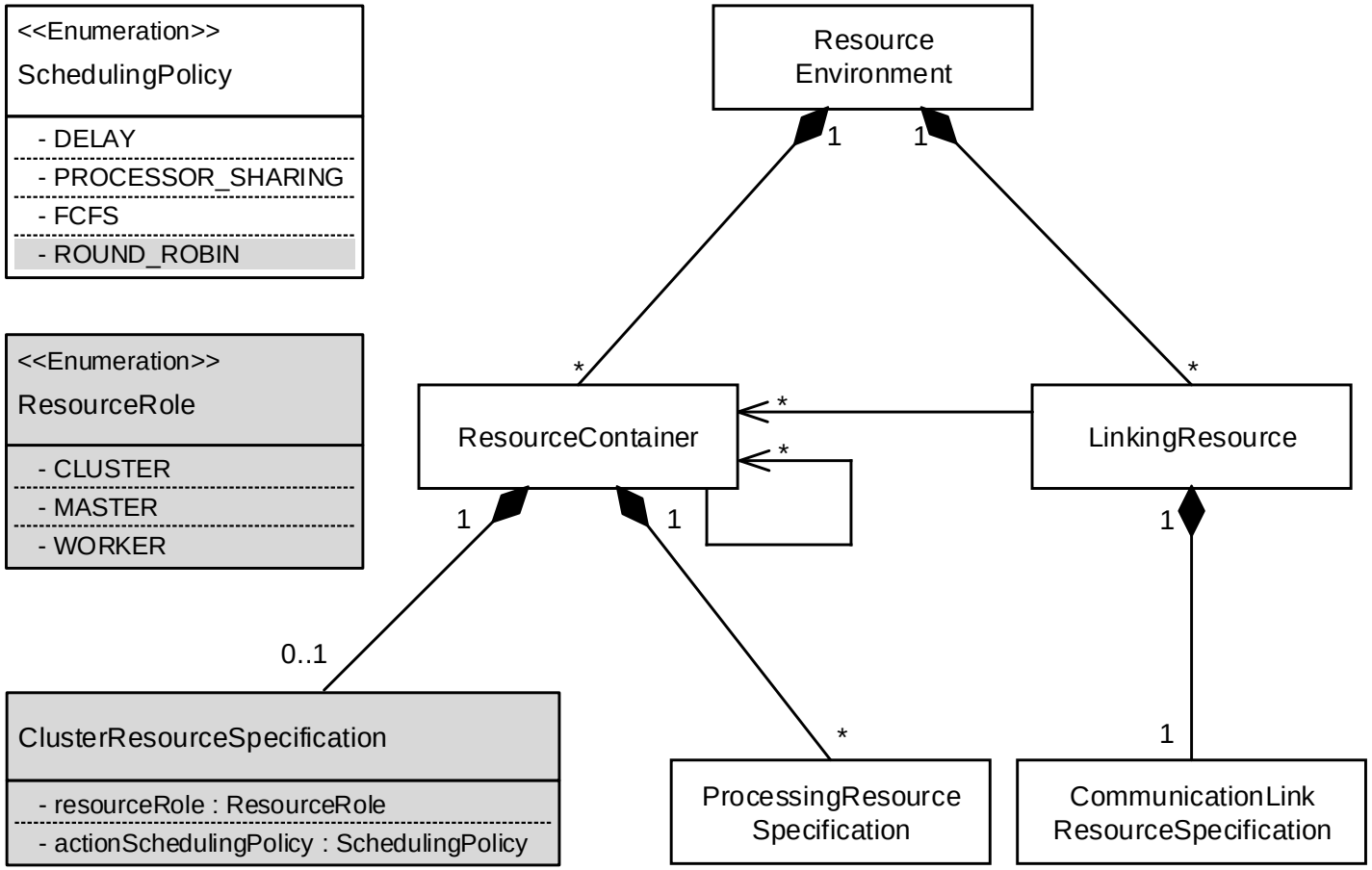
Service effect specification (SEFF) actions



PCM Version 3.4.1

PCM Meta-model Extension

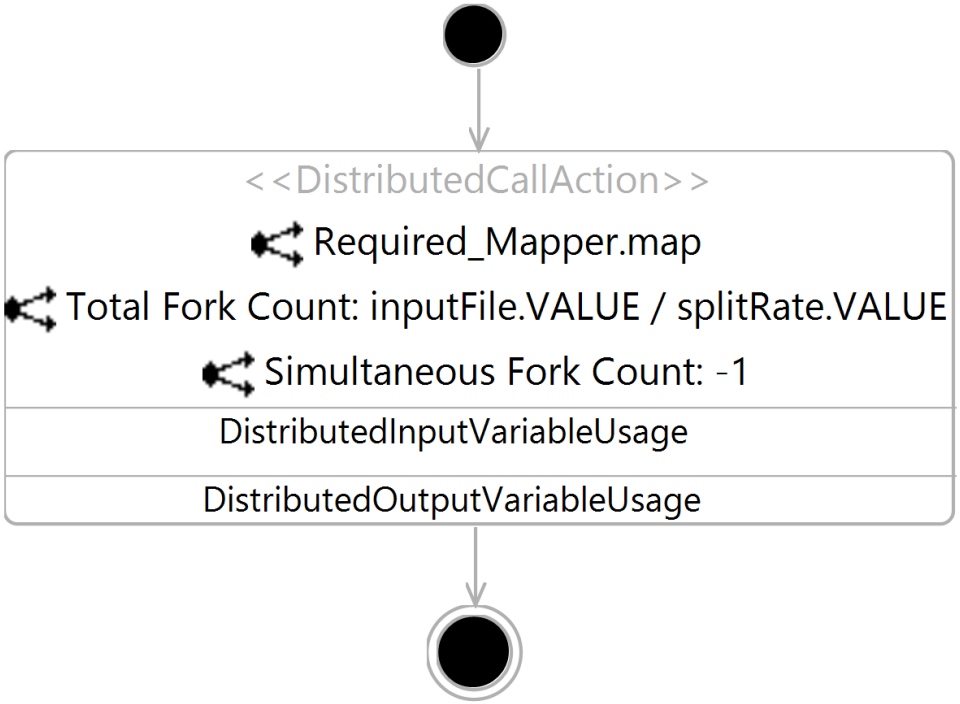
Resource environment



PCM Version 3.4.1

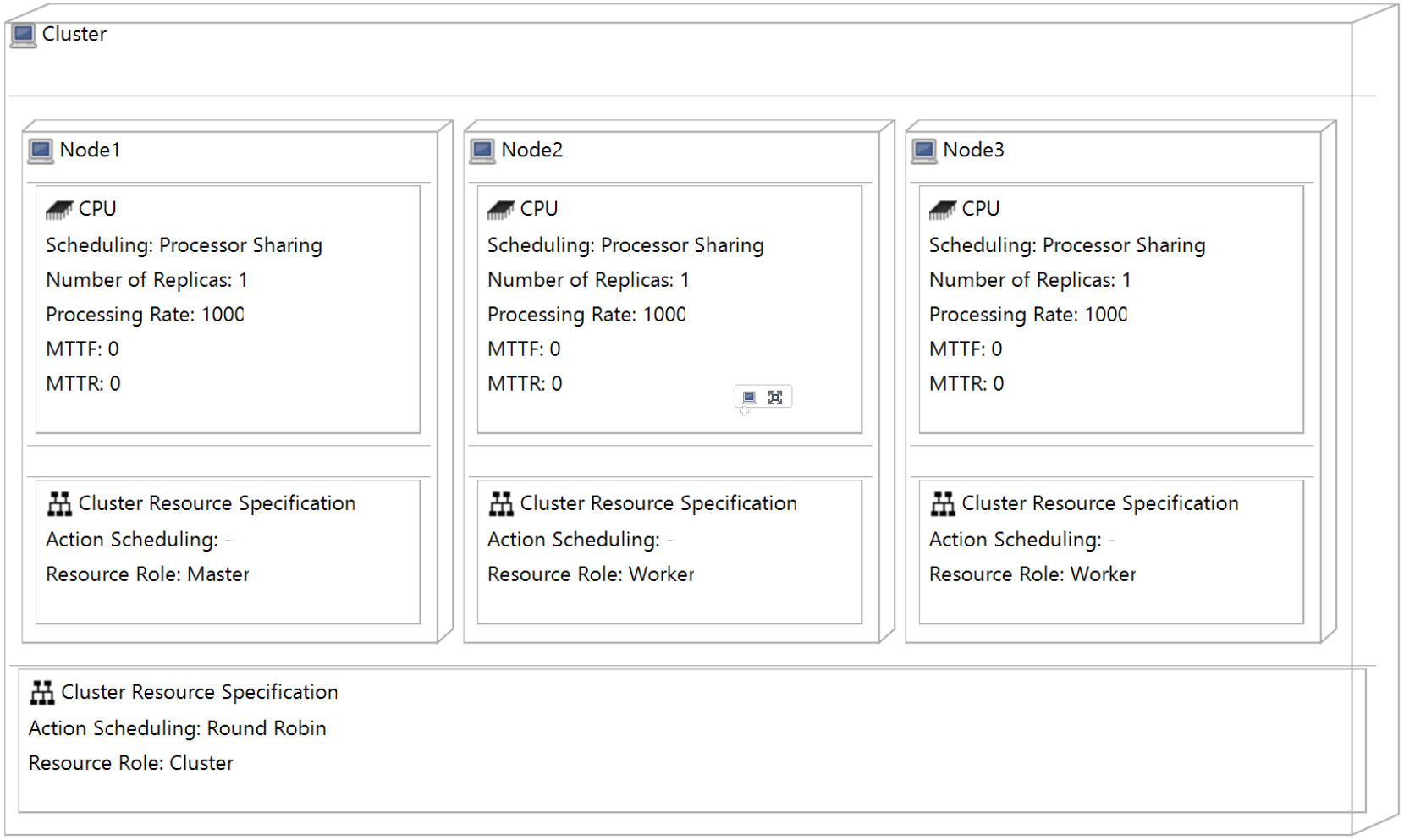
PCM Meta-model Extension

Service effect specification (SEFF) diagram



PCM Meta-model Extension

Resource environment diagram



Agenda

- Motivation
- Development Process and Characteristics of Big Data Systems
- Palladio Component Model (PCM) Meta-model Extension
- Related Work
- Conclusion and Future Work

Related Work

- Ginis and Strom (2013) present a method for predicting the response time of stream processes in distributed systems
- Verma et al. (2011) introduce the ARIA framework which specifies on strategy scheduling of single Apache MapReduce jobs
- Vianna et al. (2013) propose an analytical performance model which focuses on the pipeline between map and reduce jobs
- Barbierato et al. (2013) and Ge et al. (2013) present modeling techniques for Apache MapReduce which allow to estimate response times only
- Castiglione et al. (2014) use Markovian agents and mean field analysis to model big data batch applications and to provide information about performance of cloud-based data processing architectures

Agenda

- Motivation
- Development Process and Characteristics of Big Data Systems
- Palladio Component Model (PCM) Meta-model Extension
- Related Work
- Conclusion and Future Work

Conclusion and Future Work

- We introduced a modeling approach that allows to model essential characteristics of data processing as found in big data systems
- We presented to meta-model extensions for PCM ..

  …to model a computer cluster and

  …to apply distributed and parallel operations on this cluster
- We plan to ...

  … complete extending the simulation framework SimuCom

  … fully evaluate our extensions for up- and downscaling scenarios

  … automatically derive performance models based on measurement data

References

- Barbierato, E., Gribaudo, M., Iacono, M.: Performance evaluation of nosql big-data applications using multi-formalism models. Future Generation Computer Systems 37(0), 345-353 (2014)
- Casado, R., Younas, M.: Emerging trends and technologies in big data processing. Concurrency and Computation: Practice and Experience 27(8), 2078-2091 (2015)
- Castiglione, A., Gribaudo, M., Iacono, M., Palmieri, F.: Modeling performances of concurrent big data applications. Software: Practice and Experience (2014)
- Ge, S., Zide, M., Huet, F., Magoules, F., Lei, Y., Xuelian, L.: A Hadoop MapReduce performance prediction method. In: Proceedings of the IEEE 10th International Conference on High Performance Computing and Communications & 2013 IEEE International Conference on Embedded and Ubiquitous Computing, 820-825 (2013)
- Ginis, R., Strom, R.E.: Method for predicting performance of distributed stream processing systems. *US Patent 8,499,069*, url: https://www.google.com/patents/US8499069 (2013)
- Kroß, J., Brunnert, A., Prehofer C., Runkler, T., Krcmar, H.: Stream processing on demand for lambda architectures. *Computer Performance Engineering* (Vol. 9272) Eds.: M. Beltrán, W. Knottenbelt, and J. Bradley, pp. 243-257. Springer International Publishing (2015)
- Verma, A., Cherkasova, L., Campbell, R.H.: Aria: automatic resource inference and allocation for mapreduce environments. In: Proceedings of the 8th ACM International Conference on Autonomic Computing. pp. 235-244. ACM, New York, NY, USA (2011)
- Vianna, E., Comarela, G., Pontes, T., Almeida, J., Almeida, V., Wilkinson, K., Kuno, H., Dayal, U.: Analytical performance models for mapreduce workloads. International Journal of Parallel Programming 41(4), 495-525 (2013)

CONTACT US

Johannes Kroß

kross@fortiss.org

performancegroup@fortiss.org

pmw.fortiss.org