

A Framework for BPMS Performance and Cost Evaluation on the Cloud

Guillaume Rosinosky, Samir Youcef, François
Charoy

guillaume.rosinosky@bonitasoft.com
@grosinosky



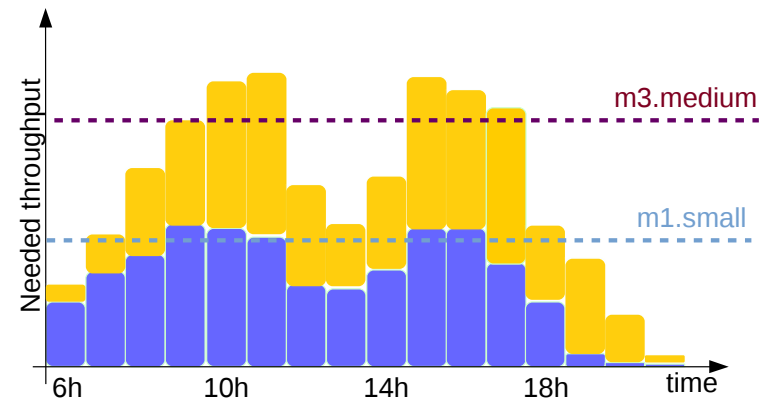
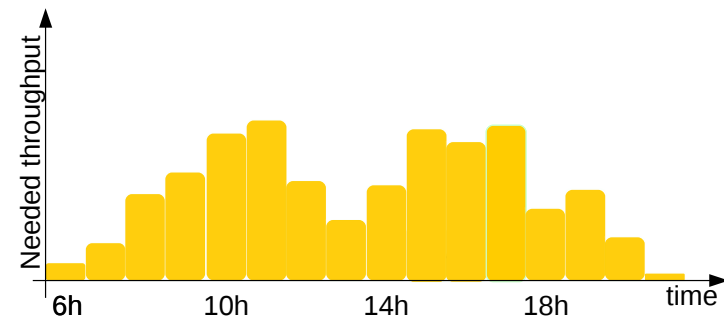
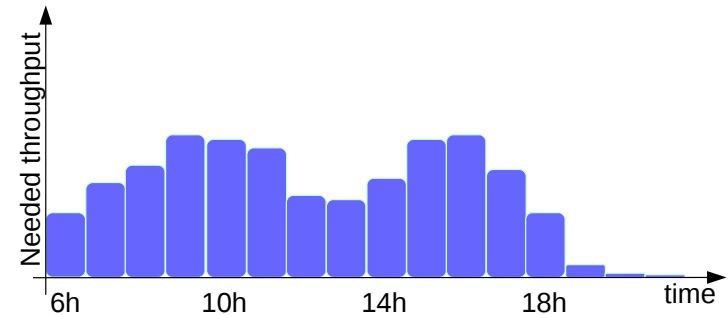


Introduction

- Goal : find elastic methods for BPMaaS resource allocation
- CIFRE thesis (mixed Bonitasoft/INRIA Nancy)
- Dual problematic
- Bonitasoft now not very interested in allocation resource methods but the tools can be useful...
-

Problematic (research)

- Find elasticity methods for BPMaaS
 - Resource allocation & scheduling
 - Price minimization for defined quality
- Hypothesis
 - BPM task number throughput as a quality metric
- Need to evaluate
 - Resource size
 - BPM task throughput justification
 - Later : test load balancer coupled with algorithm





Problematic (entreprise)

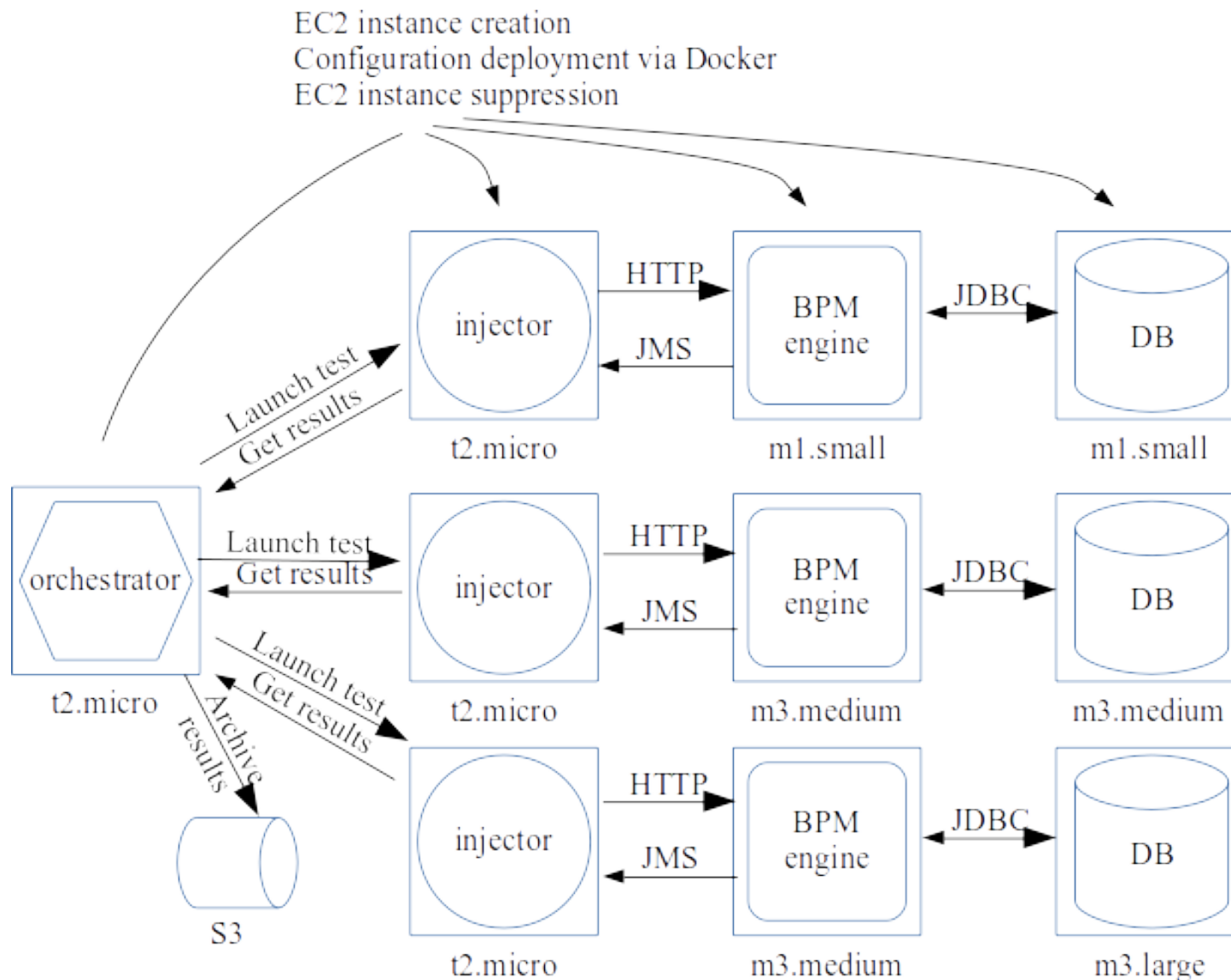
- Integration tests
 - Evaluate non regression between versions
- Performances for customers
 - Sizing, cloud configuration recommendation
- Evaluate different configurations
 - On premises
 - Cloud (Amazon, but next other vendors)
- Ability to retrieve logs



Problematic

- Have an Open Source framework able to :
 - Allocate cloud resources
 - Deploy needed software on it
 - Database
 - BPMS
 - Load balancer
 - Testing tool
 - Deallocate cloud resources
- Get metrics for analysis
- Archive

Problematic





Existing work

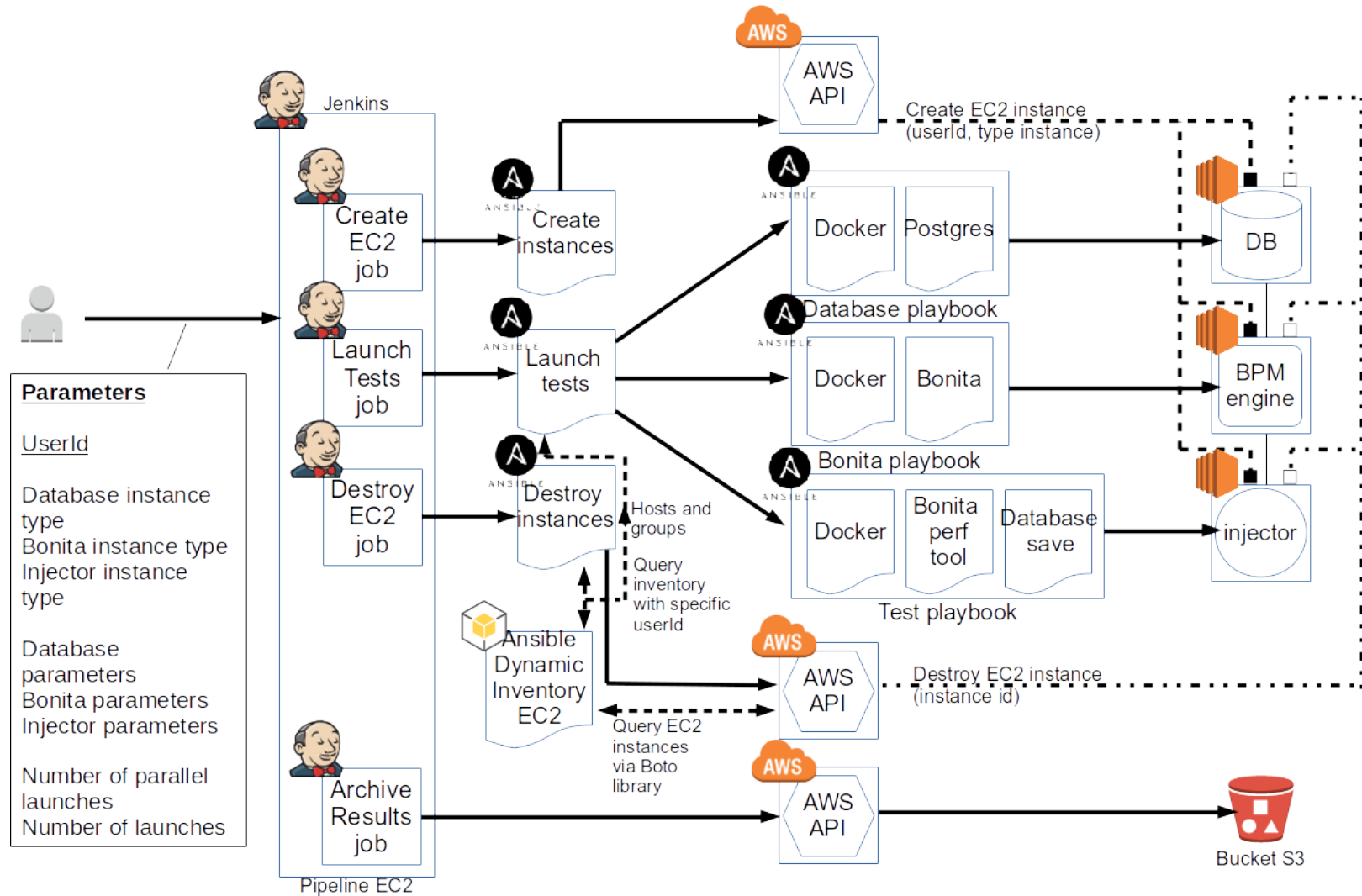
- BPMS framework
 - Benchflow
 - Based on Faban, process centric
 - Done on premise, without orchestrator
 - Betsy
 - BPMN or BPEL compliance
- Generic frameworks
 - Commercial such as Smart CloudBench



Used tools

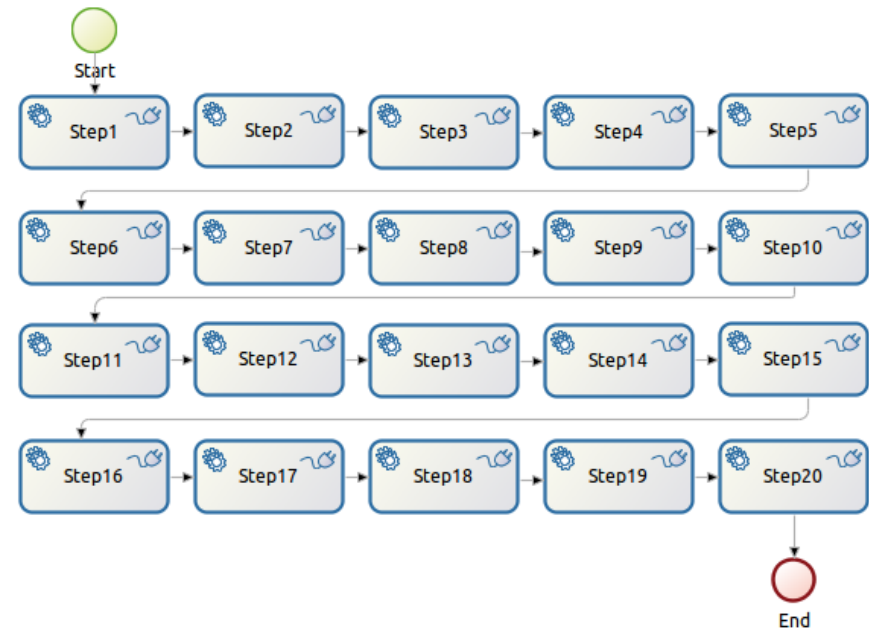
- Docker : containers
 - « VM without overhead » + configuration
 - Repeatable and isolated runs
 - => Database, Bonita, load balancer, injector images
- Ansible : scripts and orchestration tool
 - Automation tool for cloud provisioning, configuration management and application deployment
 - Used for cloud provisioning, resources initialization, test launches
- Jenkins : continuous integration automation server
 - Web UI used for software project builds and deployment,
 - UI, test archive and scripting

General schema

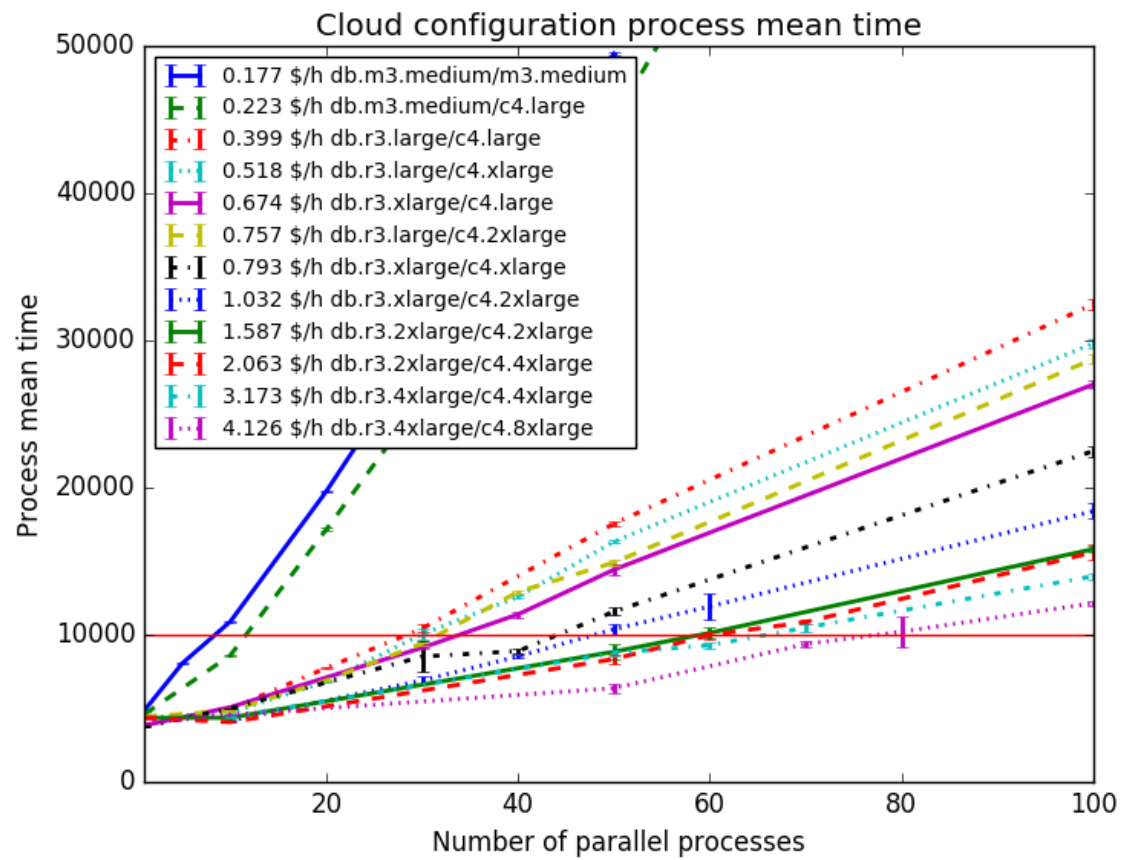


Tests

- Software
 - Bonita Benchmark tool
 - Bonita 7.2.3 community edition official Docker
 - Postgresql official Docker
- Hardware : Amazon Web Services
 - Test c4 family and m3.medium (BPMS), r3 family and m3.medium (database)
- Used process : « standard process » (20 consecutive automated tasks), launched 3000 times
- Tests : 6 consecutives tests on each configuration

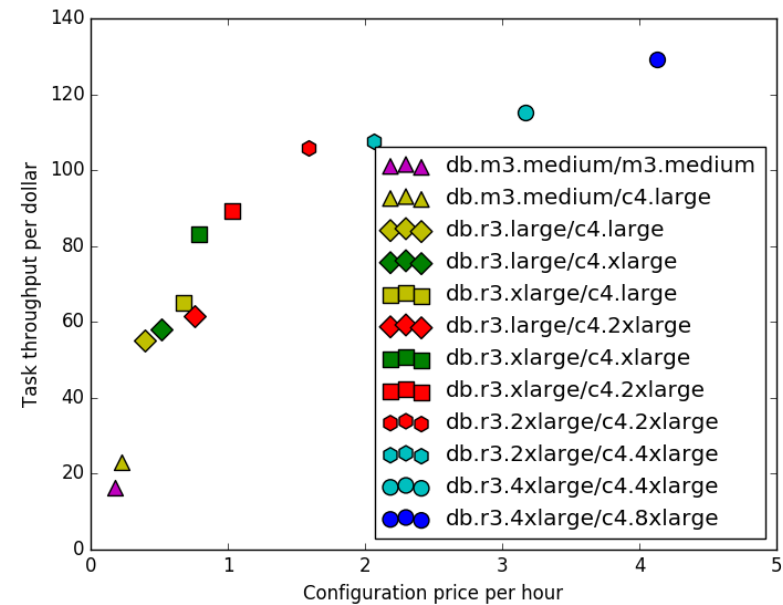


Results



Results

DB inst. type	AS inst. type	price	task TP	task TP per \$
db.m3.medium	m3.medium	0.177	16.400	92.656
db.m3.medium	c4.large	0.223	23.157	103.845
db.r3.large	c4.large	0.399	55.164	138.255
db.r3.large	c4.xlarge	0.518	58.067	112.100
db.r3.xlarge	c4.large	0.674	65.113	96.607
db.r3.large	c4.2xlarge	0.757	61.474	81.208
db.r3.xlarge	c4.xlarge	0.793	83.236	104.963
db.r3.xlarge	c4.2xlarge	1.032	89.149	86.384
db.r3.2xlarge	c4.2xlarge	1.587	105.794	66.663
db.r3.2xlarge	c4.4xlarge	2.063	107.585	52.150
db.r3.4xlarge	c4.4xlarge	3.173	115.283	36.332
db.r3.4xlarge	c4.8xlarge	4.126	129.279	31.332





Conclusion

- Effective tool for performance tests
- Research part
 - used for papers, works well
 - more generalization needed
- Enterprise part :
 - used internally for performance tests,
 - used for clustered version tests



Perspectives

- Add hardware metrics
- Add other cloud providers
- Add other BPMS
- Combine with REST client for BPMS tests
- « Dockerize » orchestrator



References and links

-
- Tools
 - Docker : <https://www.docker.com/>
 - Ansible : <https://www.ansible.com/>
 - Jenkins : <https://jenkins.io/>
 - BonitaBPM : <http://www.bonitasoft.com/>
 - Postgresql : <https://www.postgresql.org/>



Thank you :)

- Thanks to Bonitasoft, and AWS
- **Mail :** guillaume.rosinosky@bonitasoft.com
- ... questions ?



Demo

Let's take a look here