Will Be Defined After the Brainstorming Phase*

Eirik Brandztæg[1,2] and Sébastien Mosser[1]

[1] SINTEF IKT, Oslo, Norway
[2] University of Oslo, Oslo, Norway
{firstname.lastname}@sintef.no

Abstract. ∼150 words expected. Will also be defined after the brainstorming phase. Must mention *(i)*the problem, *(ii)*the actual contribution and *(iii)*the obtained results.

1 Introduction

I'll write the introduction afterwards, when the content of the paper will be fixed.

- Cloud–computing research field [1]
- Model–driven engineering applied to the cloud

2 Running Example: Give it a name

Name: bank-demo, grails-bank-example?

- How it works
  - Simple user-account system
- Technologies
  - YALOA
  - Grails (Spring, GSP, GORM, Spring, Hibernate), Spring security, Spring webflow
- Why this demo
  - Simplicity, commonly used example
  - Enterprise technologies
  - Easily examplified over several nodes/instances

3 Give it a name too

Name: CloudML, cloudml-engine?

* This work is funded by the European commission through the REMICS project, contract number 257793, with the 7th Framework Program.

3.1 Showcase 1, the idea (multicloud template language)

- Model-driven approach (template in JSON)
  - Reusability
  - Design / Architecture decoupling
  - Less technical
- Small/Simple template
  - Pros
    - Less complexity
    - Robustness
    - Same between providers
  - Cons
    - Less Flexibility
    - Loss of features and services

3.2 Showcase 2, the engine

- Written in Scala, builds with Maven, works from Java
- Multicloud support
- Validates the example (Can create nodes)
- Can be hosted by third-parties with REST interface

Badish: No support for deployment - yet

If the contribution does not fit in a single section, it can be divided into two sections.

4 Validation & Experiments

- Three instances
  1. Frontend webapp
  2. Frontend webapp
  3. Backend database
- Setup works on
  - AWS EC2
  - Rackspace cloudservers

5 Related Works

- AWS CloudFormation (Amazon only)
- jclouds (Only through (more advance?) code)
- libcloud (Only through (more advance?) code)
- CA Applogic (only graphical, and inhouse)

6 Conclusions

I'll write the conclusions afterwards.

References

1. Armbrust, M., Fox, A., Griffith, R., Joseph, A.D., Katz, R.H., Konwinski, A., Lee, G., Patterson, D.A., Rabkin, A., Stoica, I., Zaharia, M.: Above the Clouds: A Berkeley View of Cloud Computing. Tech. Rep. UCB/EECS-2009-28, EECS Department, University of California, Berkeley (Feb 2009), http://www.eecs.berkeley.edu/Pubs/TechRpts/2009/EECS-2009-28.html