Abstracting application deployment on Cloud infrastructures

Application deployment on Cloud

Common requirements from users
- The application and its supporting services should be highly available
- Horizontal scaling should be as painless as possible
- The entire application stack should be deployable on a range of different clouds with minimal friction
- Focus on high level aspects of the application
- Virtual infrastructure resources should be abstracted away
- Application complexity should not get in the way

Challenges with an IaaS-only approach
- Infrastructure-as-a-Service is a low-level abstraction
- Horizontal scaling is difficult and requires human intervention
- Re-deploying the application stack is error-prone and time consuming
- All aspects of the application lifecycle are not covered
- Virtual infrastructure resources are the highest abstraction level available
- Highly specialized skillset required to deploy a complex application

Goals
- Provide a suitable abstraction level which hides the IaaS details as much as possible
- Automate all aspects of the application lifecycle from deployment to scaling
- Make deployments reproducible across clouds with similar characteristics

Abstraction layer

We developed a two-level abstraction layer to simplify application deployment

Cloud side abstraction – Orchestration custom templates
- Abstraction from application details
- Application components and dependencies such as backends (e.g. databases, filesystem, cache system) are described by custom templates
- Control on the order of services instantiation
- Error chance limited
- Deployment time decisively reduced
- Significant advantage in deploy recurring infrastructures and testbeds

User side abstraction - Web interface
- Abstraction from application details
- Abstraction from Cloud infrastructure
- Easy-to-use instrument for not skilled users
- Possibility to scale the application

Use cases

Present
- Control system based on Highly Abstracted and Open Structure
- Orchestration through OpenStack Heat
- PHP based GUI exploiting Heat APIs
- Manually scaling backend components
- Deployment time reduced from days to minutes

Future
- EC Horizon 2020 project
- Standardization of the Orchestration service based on TOSCA templates
- Standard Web interfaces as Future Gateway programmable interface
- Dynamic deployment of a virtual site for interactive analysis
- Virtual batch systems on opportunistic cloud