Roberto Di Cosmo (coordinator)

18 Avril 2013
An Ideal Cloud Application

The promise:
- Automatic deployment
- Automatic (re)configuration
- Automatic scaling

The reality:
Long, detailed, manual descriptions of the full system are mandatory.

Aeolus plans to help fill the gap.

R. Di Cosmo (CNRS/INRIA/MDV/UPD)
An Ideal Cloud Application

The promise:
- automatic deployment

The reality:
- Long, detailed, manual descriptions of the full system are mandatory

Aeolus plans to help fill the gap
An Ideal Cloud Application

The promise:
- automatic deployment
- automatic (re)configuration
An Ideal Cloud Application

The promise:
- automatic deployment
- automatic (re)configuration
- automatic scaling

Aeolus plans to help fill the gap.
An Ideal Cloud Application

The promise:
- automatic deployment
- automatic (re)configuration
- automatic scaling

The reality:
Long, detailed, manual descriptions of the full system are mandatory
An Ideal Cloud Application

The promise:
- automatic deployment
- automatic (re)configuration
- automatic scaling

The reality:
Long, detailed, manual descriptions of the full system are mandatory

Aeolus plans to help fill the gap
Aeolus Focus

Key ingredients:
- A model of a cloud system,
- A high level description language for expressing reconfiguration requests,
- A low-level vendor-independent deployment description language,
- Advanced, specialised algorithms for optimised platform deployment and (re)configuration.

R. Di Cosmo (CNRS/INRIA/MDV/UPD)
Key ingredients:
- a model of a cloud system,
Aeolus Focus

Key ingredients:

- a model of a cloud system,
- a high level description language for expressing reconfiguration requests,
Aeolus Focus

Key ingredients:

- a model of a cloud system,
- a high level description language for expressing reconfiguration requests,
- a low-level vendor-independent deployment description language,
Aeolus Focus

Key ingredients:
- a model of a cloud system,
- a high level description language for expressing reconfiguration requests,
- a low-level vendor-independent deployment description language,
- advanced, specialised algorithms for optimised platform deployment and (re)configuration.
Aeolus Cloud model

Simple, but expressive model: finite internal state of software components, functional requirements and provides, nonfunctional requirements via capacity constraints;
Aeolus Cloud model

Simple, but expressive model: finite internal state of software components, functional requirements and provides, nonfunctional requirements via capacity constraints; with a very natural and concise graphical language.
Aeolus Cloud model

Simple, but expressive model: finite internal state of software components, functional requirements and provides, nonfunctional requirements via capacity constraints; with a very natural and concise graphical language.

A fragment of a realistic configuration for a WordPress web service.
Preliminary results: Zephyrus

An automatic tool for creating a full system configuration from software components and sophisticated user requests, satisfying all constraints.

Uses a sophisticated encoding into the *minizinc* language, and a specialised "candy" algorithm to compute component connections.
Future work

Zephyrus computes a final configuration; to reach it, one needs to compute a plan. This is a difficult task:

the basic model without conflicts or capacity constraints has a decidable reachability problem.
Future work

Zephyrus computes a final configuration; to reach it, one needs to compute a *plan*. This is a difficult task:

the basic model without conflicts or capacity constraints has a *decidable* reachability problem

the full model with conflicts and capacity constraints has an *undecidable* reachability problem
Future work

Zephyrus computes a final configuration; to reach it, one needs to compute a *plan*. This is a difficult task:

- **the basic model** without conflicts or capacity constraints has a *decidable* reachability problem
- **the full model** with conflicts and capacity constraints has an *undecidable* reachability problem

This explains why industry tools for automatic reconfiguration are quite limited, behind the marketing varnish: the problem itself is hard!
Future work

Zephyrus computes a final configuration; to reach it, one needs to compute a *plan*. This is a difficult task:

- **the basic model** without conflicts or capacity constraints has a *decidable* reachability problem
- **the full model** with conflicts and capacity constraints has an *undecidable* reachability problem

This explains why industry tools for automatic reconfiguration are quite limited, behind the marketing varnish: the problem itself is hard!

We are investigating interesting subclasses of the Aeolus model that allow efficient planning.
Industry use case: Pulse2 from Mandriva

IT management solution for big organizations: highly distributed architecture, can handle network link failures and heterogeneity. Here is a typical installation:
Industry use case: Pulse2 from Mandriva

IT management solution for big organizations: highly distributed architecture, can handle network link failures and heterogeneity. Here is a typical installation:

Main issue: decide how many package servers to deploy and where.
Optimal deployment of Pulse2

We have designed a simple, but expressive model for the deployment of the package servers.

The Package Server Location Problem (PSLP) is formulated as a multi-objective optimization problem for which the (contradictory) objectives are to:

- minimize investment costs,
- maximize the efficiency and the reliability of the service,
- allow spare capacity for robust operations.

PSLP is related to network design and facility location problems.
Optimal deployment of Pulse2

We have designed a simple, but expressive model for the deployment of the package servers.

The Package Server Location Problem (PSLP) is formulated as a multi-objective optimization problem for which the (contradictory) objectives are to:

- minimize investment costs,
- maximize the efficiency and the reliability of the service,
- allow spare capacity for robust operations.

PSLP is related to network design and facility location problems.

Solutions of a PSLP instance can be used to enrich configuration requests for Zephyrus.
Learn more, get involved

