Virtual Environments: Easy Modeling of Interdependent Virtual Appliances in the Cloud

Xabriel J. Collazo-Mojica¹, S. Masoud Sadjadi¹, Fabio Kon², Dilma Da Silva³

1: Florida International University, USA
2: Universidade de São Paulo, Brazil
3: IBM T.J. Watson Research Center, USA
The Problem

- Realizing groups of interdependent virtual machines (VMs) in the cloud is a common task in today’s Internet.
  - This require a good understanding of:
    - Software installation for each piece of the stack
    - Network details such as IP addresses, ports, etc.
  - They are typically deployed in a cloud layer called Infrastructure as a Service (IaaS).
    - Each IaaS provider has different APIs!
The Problem (Cont)

- Research has pointed the need for better tools for composition in the cloud [1].
  - Clearly, an easier to understand model can help non-experts in cloud computing to develop solutions in this domain.
Motivation

• We envision that different IaaS APIs and providers will continue to proliferate.

• Non-expert users such as Web developers and CS students should not be bothered by the configuration details!

• We have identified specific concepts which could use better abstractions.
Proposed Solution

• We propose a modeling approach that is abstract enough to allow these interdependent VMs systems to be seen as “appliances providing specific services”.
  ▫ An appliance is a VM with well defined services [2].
• This abstraction allows solutions to be:
  ▫ easily designed
  ▫ fast to deploy
  ▫ unaffected by IaaS vendor lock-in
Proposed Solution (Cont)

• We call these models *Virtual Environments*.
  ▫ A *Virtual Environment* is a model of a group of interdependent virtual appliances that specifies certain constraints on the exposed services of each appliance.

• We have developed a prototype visual designer for easy composition of these environments.
The Flexible Tool: Defining VAs

Defining an Appliance:

Once defined, it looks like this:

“provides” a db service
The Flexible Tool: More VA examples

Example of an Ruby on Rails Appliance

Example of a Load Balancer Appliance

“requires” a db service

“requires” a http service
The Flexible Tool: endpoint connection example

QoS constraints could go here.
(I.e. min guaranteed throughput)
The Flexible Tool: A fully defined VE

Note this is a logical architecture!
The Flexible Tool: Visual Environment
Transforming the model

Transforming the model

QoS-Aware IaaS Cloud (work being done in our lab)
Most Relevant Related Work

- IaaS providers, such as Amazon Web Services [3] or GoGrid [4], provide raw virtualized computing power.
  - All the configuration needs to be done by the user.
  - We leverage this work by building another layer of abstraction on top of it.

- Platform as a Service (PaaS) providers, such as Google AppEngine [5], abstract away the underpinnings of a fully working web application.
  - Catch: Vendor lock-in.
  - Our approach will provide a fully working IaaS environment, but no app logic.
Most Relevant Related Work

- Commercial applications implementing a similar modeling approach are available [6, 7].
  - They only work on their proprietary cloud platforms.

- IBM has worked on a similar project [8].
  - Their implementation assumes that users are experts.
  - While they target enterprise customers, we target non-expert cloud users.
Concluding Remarks and Future Work

• In this paper, we presented the basics of our modeling ideas focusing on what matters to a non-expert end user.
  ▫ A detailed view of the underpinnings is future work.

• We envision fast deployment of working systems through an automatic configuration process.

• For future work
  ▫ Short term: fully working virtual environments.
  ▫ Long term: providing various IaaS deployment choices.
Acknowledgements

- We appreciate the discussions held with David Villegas.
- National Science Foundation under Grant No. OISE-0730065.
- US Department of Education under P200A090061.
- IBM.

Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation, the US Department of Education, or IBM.
References


References (Cont)

Questions?

Thanks so much for your time!