Performance and Deployment Evaluation of a Parallel Application in an on-premises Cloud Environment

Giacomo Mc Evoy[1] Bruno Schulze[1] Eduardo L.M. Garcia[1]

[1]National Laboratory for Scientific Computing (LNCC) - Brazil

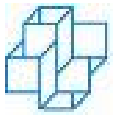
7th International Workshop on Middleware for Grids, Clouds and e-Science - MGC 2009

Laboratório Nacional de Computação Científica

Outline

Laboratório Nacional de Computação Científica

Outline



Motivation

- Provide computational resources to compute–intensive applications in a seamless fashion.
  - Grids offer scalability by increasing the number of compute nodes, but require environment setup and have the limitations of using shared resources.
  - Clouds offer additional scalability by resizing virtualized resources. Leverages virtualization techniques to provide a customized environment and access to resources appear privileged.

Objectives

- Evaluate the performance impact of a distributed application in an on–premises Cloud environment.
- Understand the deployment pattern of that application with the Cloud as a scaling mechanism.
- Elaborate the case of providing contextualization to many Globus containers in a single deployment.

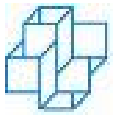
Some Definitions

- **Cloud system:** Dr. Buyya's definition:
  "A Cloud is a type of parallel and distributed system consisting of a collection of inter-connected and virtualized computers that are dynamically provisioned and presented as one or more unified computing resource(s) based on service-level agreements established through negotiation between the service provider and consumers."
  - Used as Infrastructure as a Service (IaaS)

Some Definitions (2)

- **Virtual Appliance:** Application + JeOS (Just enough Operating System) for it to run optimally in a virtualized environment. Appliance decoupled from the hardware.
- **Contextualization:** Configuration of context-sensitive applications after the deployment of comprising virtual appliances.

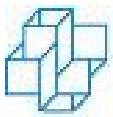
Outline



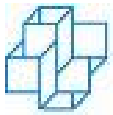
Laboratório Nacional de Computação Científica

The on–premises Cloud

- **Hardware**
  - Testing environemnt composed of heterogeneous machines.
- **Software**
  - Eucalyptus
    - An open-source cloud-computing framework.
    - Used to instantiate multiple VM instances from a single Virtual Appliance.
  - Nimbus Context Broker (NCB)
    - Deploys virtual clusters with a ready–to–use Globus container.
    - Used to obtain the hostname and IP address of each instantiated VM. Also populates the */etc/hosts* file in each VM to acknowledge the other nodes.

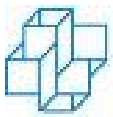


Outline



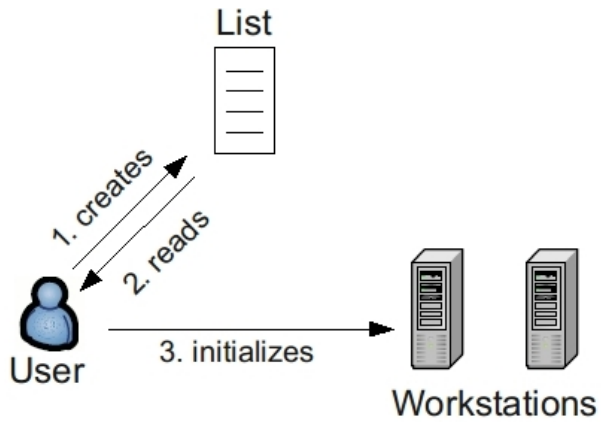
Description of Application

- Parallel numerical simulation optimization that uses an evolutionary algorithm.
  - Uses the Master–Worker architecture. The application handles scheduling.
  - Compute–intensive application, communication costs are a secondary concern.
  - Comprised of jobs (evolution steps) that are in turn composed of many decoupled tasks (individuals).
- Leverages two complementary communication mechanisms:
  - Object messages via Java RMI.
  - XML messages via Globus 4 Toolkit.



How was the Application originally deployed?

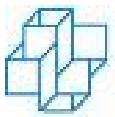
- User chooses Worker nodes, a list of Worker nodes is built manually.
- An initialization script creates the Worker instances.
  - SSH for RMI instances.
  - GridFTP and *globus-remote-deploy-gar* for Globus4 instances.



Laboratório Nacional de Computação Científica

How is the Application deployed in the Cloud?

- The Master node is chosen by the user. Worker nodes are created within the Cloud by spawning several instances of the same virtual appliance using the Eucalyptus' interface.
- The list of Worker nodes is automatically built by retrieving the Worker identities upon instantiation of the VMs. This is possible using the contextualization feature of the NCB.
- The initialization script runs in order to perform the proper file staging and Worker instantiation.
- This approach can be combined with the traditional deployment.



How is the Application deployed in the Cloud? (2)

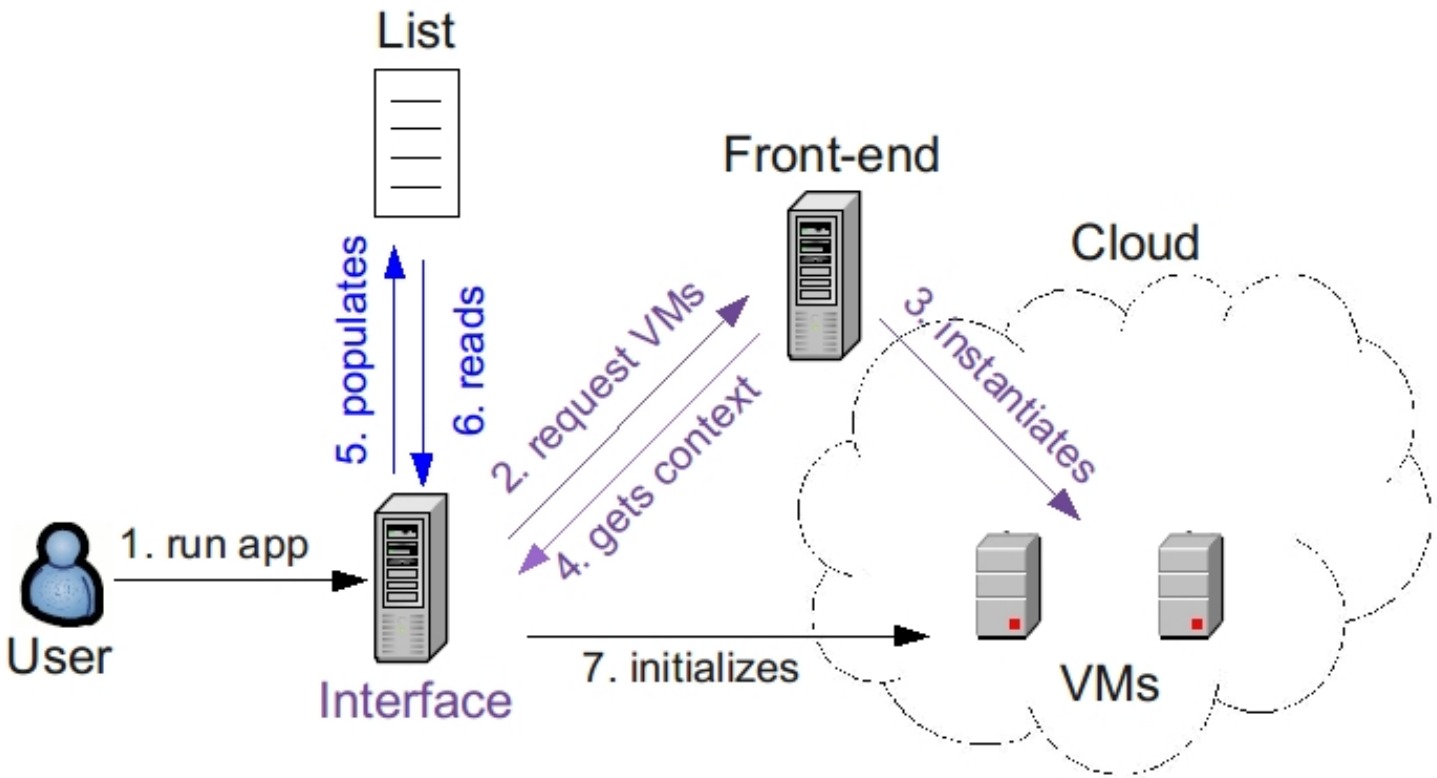
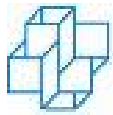


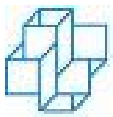
Figure: Deployment Diagram using contextualization

Outline



Single Node Evaluation

- Measures and compares compute and communication costs in a Quad–core server with four Worker instances.
- Three scenarios: native, VM and Cloud. Each scenario was evaluated for each flavor (RMI and GT4).
- Quantifies impact of virtualization ($\sim$5%) and overhead of Cloud Node Manager (additional $\sim$5%). The communication cost was doubled with virtualization.



Single Node Evaluation (2)

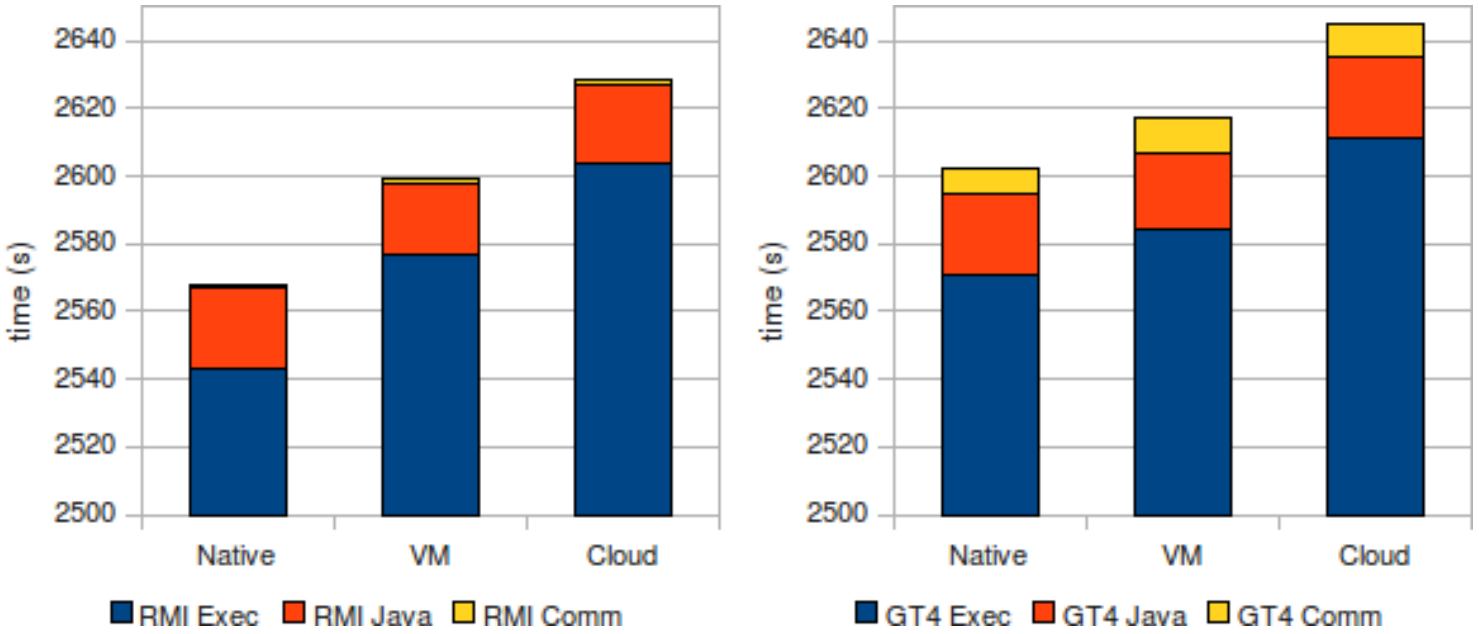


Figure: Performance metrics for a single Node* using RMI and GT4.
*AMD Phenom 9650 Quad-Core at 2.33 GHz and 8GB of RAM running four Worker instances

Laboratório Nacional de Computação Científica

Cloud Evaluation

- Calculates and compares speed-up of Working nodes.
- Same nodes were sequentally added when scaling, to avoid impact of heterogeneity.
- Compares impact of having many competing processes versus many VM instances running in the hardware.

Cloud Evaluation (2)

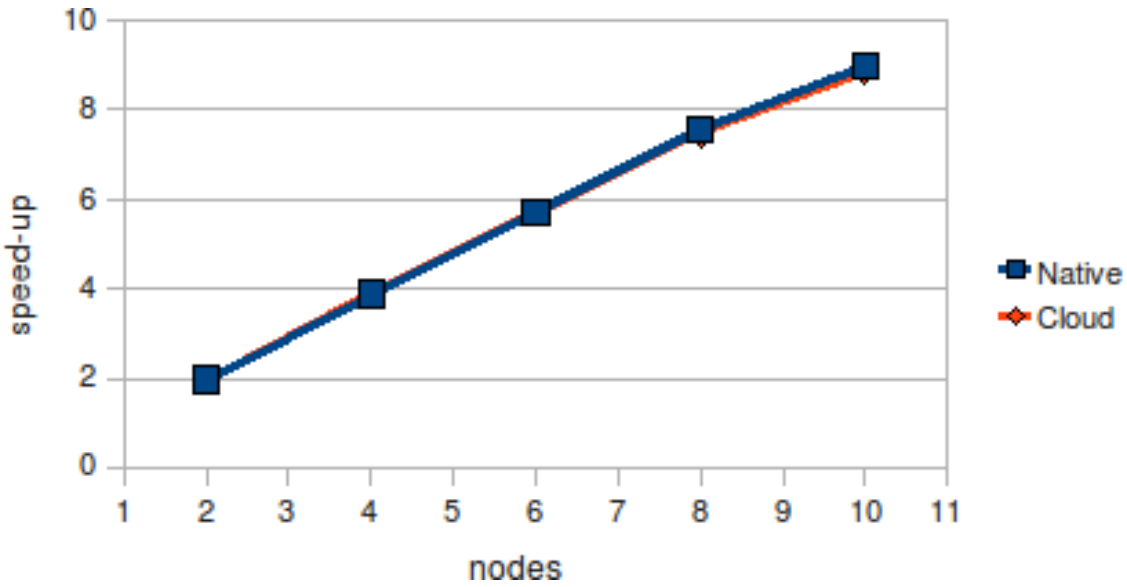
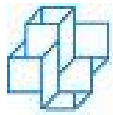


Figure: Scalability of Application in Native and Cloud scenarios

Laboratório Nacional de Computação Científica

Outline



Limitations of NCB

- The VM image provided by the NCB that configures the Globus Container assumes that the Container will be unique in the deployment.
  - The Container creates its own Certificate Authority (CA) to self–sign a host certicate.
  - Non–viable approach if deploying multiple Globus VM instances.
- The NCB provides no support to add nodes to the context after deployment.

Fixing the Security Configuration

- First possibility: dedicated CA
  - A dedicated CA can be made available specifically to sign host certificates from any VM created inside the private Cloud. Also needs to sign a certificate to the user.
  - Uses the Cloud's *metadata server* to inject information about the CA into the VM upon instantiation.
  - User's certificate is restricted (not host certificate) but Container can be tweaked to elevate the user's certificate permissions.
  - All VMs end up in the same Virtual Organization (VO).

Dedicated CA Proposal

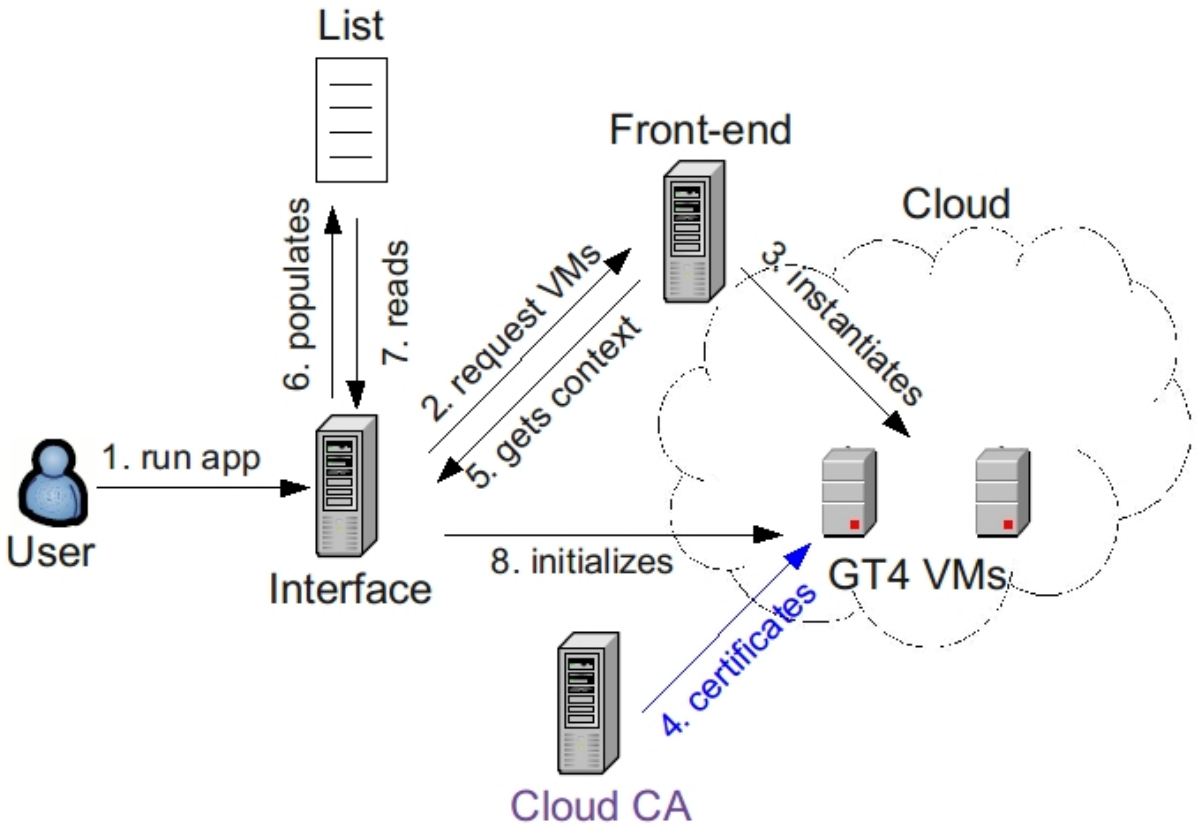


Figure: Deployment Diagram using a dedicated CA

Laboratório Nacional de Computação Científica

Fixing the Security Configuration (2)

- Second possibility: On–the–fly *localCA* inside one of the VMs
  - A VM is chosen to become the CA for the current deployment and generate a user certificate for all VMs.
  - Uses the contextualization mechanism of the NCB to establish the CA node and to notify the other nodes: the *localCA* role.
  - The VO is limited to the current deployment.

Local CA Proposal

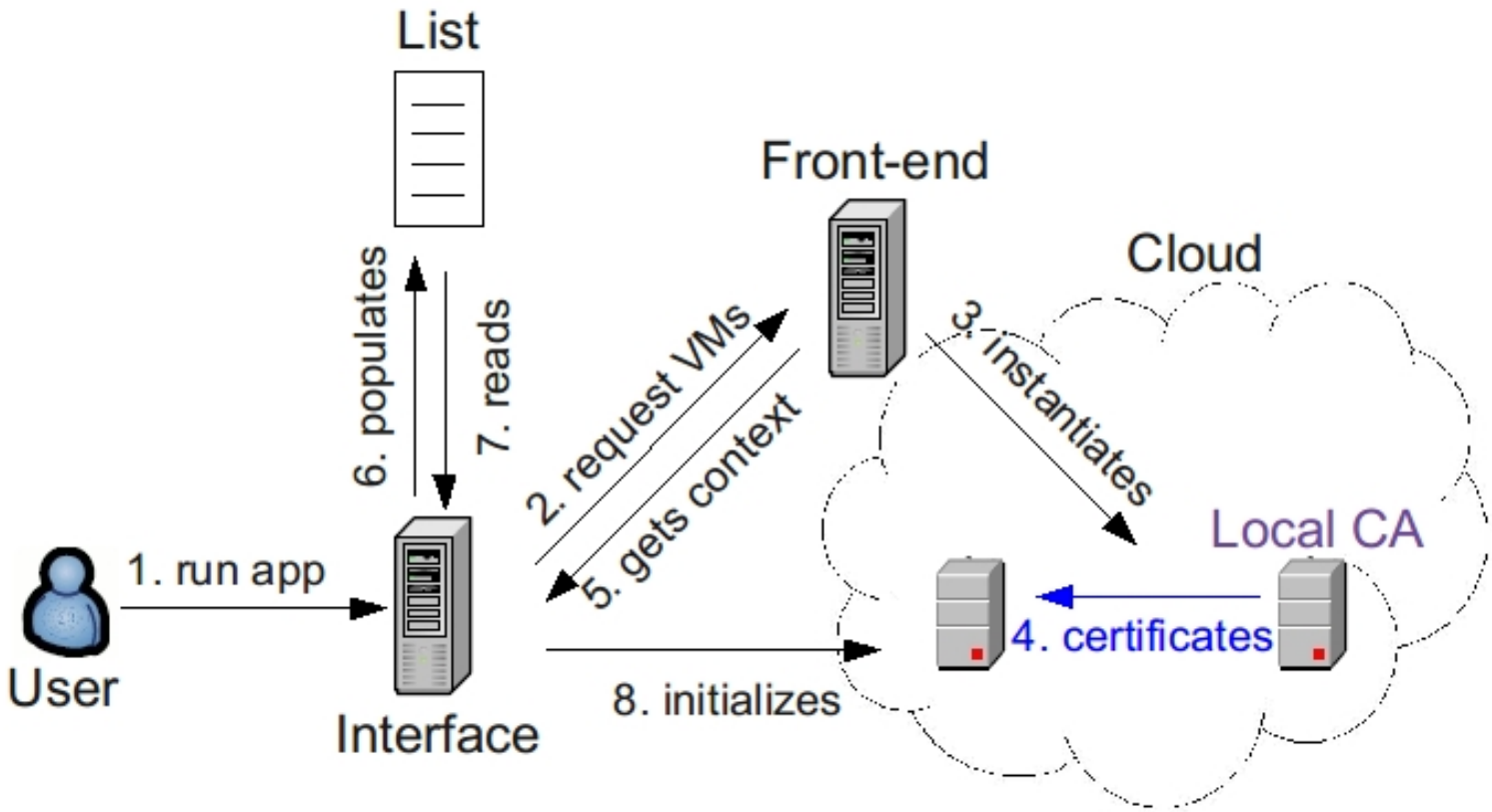
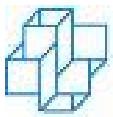


Figure: Deployment Diagram using a VM as CA

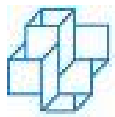
Providing Recontextualization

- Solution only works on the Master–Worker deployment pattern, because the Working nodes are decoupled.
- Establish a front–end that decides whether to use the NCB for context creation (first time) or reuse the information of the context on subsequent instantiations.
- If using the *localCA* approach, the *localCA* can store the context information for the current deployment. Only the context of the *localCA* is updated (low coupling).
- The Worker node list of the application is updated the same way. New Worker nodes are acknowledged on the next job, when the list is read again.

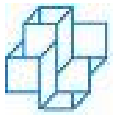


Properties of the Deployment with Globus

- **Simplifications**
  - User does not need to specify low–level details such as which VM image to use, or which roles each instance should receive.
  - User relieved of managing his credentials against multiple Globus VM instances.
  - Application staging solved automatically to the point of file staging with GridFTP.
- **Restrictions**
  - Platform expects a node to remain active in order to maintain the *localCA* role and allow further addition of Workers.
  - Recontextualization restricted to reconfigure only the instance that controls the context.
  - User loses the ability to customize VM instantiation.



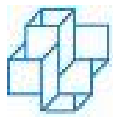
Outline



Laboratório Nacional de Computação Científica

Conclusion and Future Work

- The performance hits of both virtualization Eucalyptus Cloud Infrastructure were measured and found to be acceptable
  - Initial evaluation of scalability was encouraging but experiments with additional cores are needed.
  - Cloud Fabric will consist of several high–end servers (Intel Xeon Dual Processor Quad–Core).
- The limitations of the Nimbus Context Broker found when trying to deploy our application were fixed by our proposals.
  - *LocalCA* approach more flexible, but was not fully implemented.
  - Recontextualization still a problem for the general case.
- Deployment pattern suggests a Globus–based Cloud Platform
  - File staging, execution and security were automatized.
  - Continue work for other deployment patterns.



Thank you!
Questions?

Giacomo Mc Evoy
giacomo@lncc.br