The WNoDeS Cloud Layer

Status and Perspectives

D. Salomoni, INFN-CNAF

Davide.Salomoni@cnaf.infn.it

Workshop CCR/INFNGrid 2010 – Catania

Outline

1. Clouds and Grids: A Very Short Introduction

2. Cloud Access in WNoDeS

3. Summary

Outline

1. Clouds and Grids: A Very Short Introduction
2. Cloud Access in WNoDeS
3. Summary

The compulsory slide on definitions

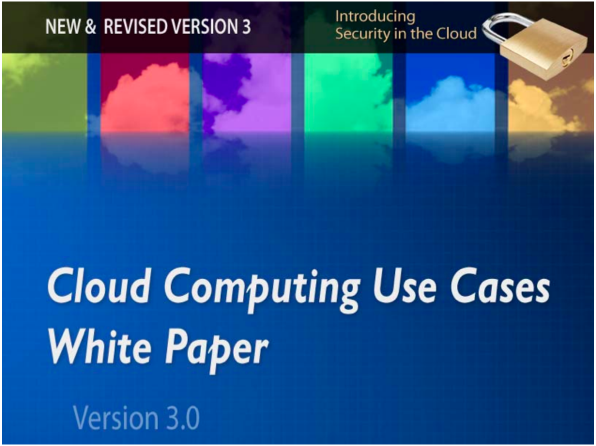
*The essence of the [definition] can be captured in a simple checklist, according to which a **Grid** is a system that:*

1. *coordinates resources that are not subject to centralized control. . .*
2. *. . . using standard, open, general-purpose protocols and interfaces. . .*
3. *. . . to deliver nontrivial qualities of service.*

(I. Foster, What is the Grid? A Three Point Checklist, 2002)

***Cloud computing** is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. (NIST Working Definition of Cloud Computing.)*

White Paper by the Cloud Computing Use Case Discussion Group



NEW & REVISED VERSION 3

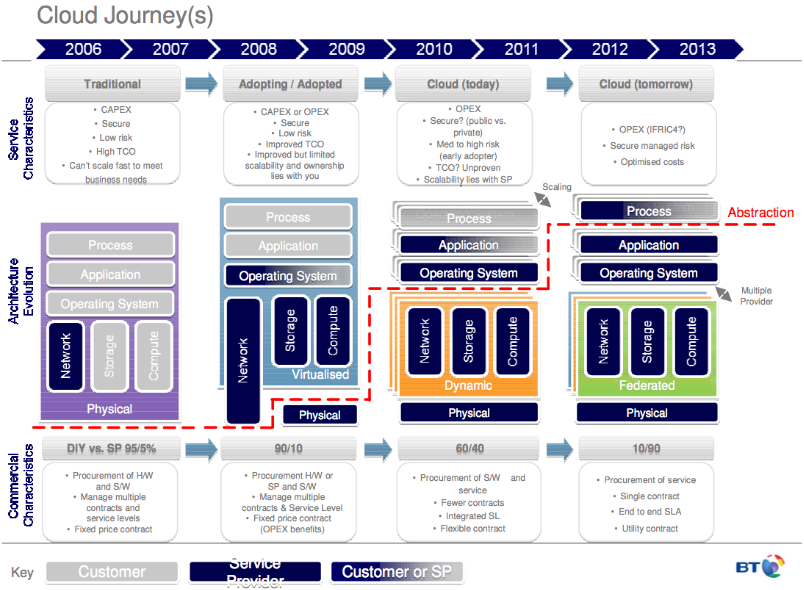
Introducing
Security in the Cloud

Cloud Computing Use Cases
White Paper

Version 3.0

Link: `http://goo.gl/aCN0`

The evolution of cloudy promises



Cloud Journey(s)

2006 2007 2008 2009 2010 2011 2012 2013

	Traditional	Adopting / Adopted	Cloud (today)	Cloud (tomorrow)
Service Characteristics	• CAPEX • Secure • Low risk • High TCO • Can't scale fast to meet business needs	• CAPEX or OPEX • Secure • Low risk • Improved TCO • Improved but limited scalability and ownership lies with you	• OPEX • Secure? (public vs. private) • Med to high risk (early adopter) • TCO? Unproven • Scalability lies with SP	• OPEX (IFRIC4?) • Secure managed risk • Optimised costs
Architecture Evolution	Process Application Operating System Network, Storage, Compute Physical	Process Application Operating System Network, Storage, Compute Virtualised Physical	Process Application Operating System Network, Storage, Compute Dynamic Physical	Process Application Operating System Network, Storage, Compute Federated Physical
Commercial Characteristics	DIY vs. SP 95/5% • Procurement of H/W and S/W • Manage multiple contracts and service levels • Fixed price contract	90/10 • Procurement H/W or SP and S/W • Manage multiple contracts & Service Level • Fixed price contract (OPEX benefits)	60/40 • Procurement of S/W and service • Fewer contracts • Integrated SL • Flexible contract	10/90 • Procurement of service • Single contract • End to end SLA • Utility contract

New requirements to existing Grid infrastructures

While Grid interfaces are widely used esp. by large user communities, Cloud computing offers significant advantages for many uses (among them, pay-as-you-go models, simplified access).

Ideally, though, one would like to adopt Cloud services so that:

- Resources are shared between access interfaces (Grid, Cloud, or else).
- Scalability is ensured.
- Existing services and agreements are not required to change.
- Resource center policies and know-how are honored.
- New services can attract both existing and new customers.

These are both key challenges and opportunities for existing Grid infrastructures.

New requirements to existing Grid infrastructures

While Grid interfaces are widely used esp. by large user communities, Cloud computing offers significant advantages for many uses (among them, pay-as-you-go models, simplified access).

Ideally, though, one would like to adopt Cloud services so that:

- Resources are shared between access interfaces (Grid, Cloud, or else).
- Scalability is ensured.
- Existing services and agreements are not required to change.
- Resource center policies and know-how are honored.
- New services can attract both existing and new customers.

These are both key challenges and opportunities for existing Grid infrastructures.

New requirements to existing Grid infrastructures

While Grid interfaces are widely used esp. by large user communities, Cloud computing offers significant advantages for many uses (among them, pay-as-you-go models, simplified access).

Ideally, though, one would like to adopt Cloud services so that:

- Resources are shared between access interfaces (Grid, Cloud, or else).
- Scalability is ensured.
- Existing services and agreements are not required to change.
- Resource center policies and know-how are honored.
- New services can attract both existing and new customers.

These are both key challenges and opportunities for existing Grid infrastructures.

New requirements to existing Grid infrastructures

While Grid interfaces are widely used esp. by large user communities, Cloud computing offers significant advantages for many uses (among them, pay-as-you-go models, simplified access).

Ideally, though, one would like to adopt Cloud services so that:

- Resources are shared between access interfaces (Grid, Cloud, or else).
- Scalability is ensured.
- Existing services and agreements are not required to change.
- Resource center policies and know-how are honored.
- New services can attract both existing and new customers.

These are both key challenges and opportunities for existing Grid infrastructures.

New requirements to existing Grid infrastructures

While Grid interfaces are widely used esp. by large user communities, Cloud computing offers significant advantages for many uses (among them, pay-as-you-go models, simplified access).

Ideally, though, one would like to adopt Cloud services so that:

- Resources are shared between access interfaces (Grid, Cloud, or else).
- Scalability is ensured.
- Existing services and agreements are not required to change.
- Resource center policies and know-how are honored.
- New services can attract both existing and new customers.

These are both key challenges and opportunities for existing Grid infrastructures.

New requirements to existing Grid infrastructures

While Grid interfaces are widely used esp. by large user communities, Cloud computing offers significant advantages for many uses (among them, pay-as-you-go models, simplified access).

Ideally, though, one would like to adopt Cloud services so that:

- Resources are shared between access interfaces (Grid, Cloud, or else).
- Scalability is ensured.
- Existing services and agreements are not required to change.
- Resource center policies and know-how are honored.
- New services can attract both existing and new customers.

These are both key challenges and opportunities for existing Grid infrastructures.

New requirements to existing Grid infrastructures

While Grid interfaces are widely used esp. by large user communities, Cloud computing offers significant advantages for many uses (among them, pay-as-you-go models, simplified access).

Ideally, though, one would like to adopt Cloud services so that:

- Resources are shared between access interfaces (Grid, Cloud, or else).
- Scalability is ensured.
- Existing services and agreements are not required to change.
- Resource center policies and know-how are honored.
- New services can attract both existing and new customers.

These are both key challenges and opportunities for existing Grid infrastructures.

Examples of services requested *today*

Some of the typical *new* service requests:

- Customer-definable software environments. This is a feature that finds several uses in "traditional" Grids as well.
- Setting up dynamic pools of virtual servers (e.g., user interfaces, or worker nodes for parallel interactive analysis). More generally, flexibly allocating hardware resources through complex advance-reservation requests.
- Instantiating pre-packaged, ready-to-go services.
- Truly distributed, on-demand, Cloud storage.
- Not everybody "speaks Grid": providing access to distributed, traditional Grid infrastructures as if they were not Grids, also to non-traditional users, like Public Administrations, or to the private sector.

The key problem is one of integration between several access interfaces (Grid, Cloud, or else).

Examples of services requested *today*

Some of the typical *new* service requests:

- Customer-definable software environments. This is a feature that finds several uses in "traditional" Grids as well.
- Setting up dynamic pools of virtual servers (e.g., user interfaces, or worker nodes for parallel interactive analysis). More generally, flexibly allocating hardware resources through complex advance-reservation requests.
- Instantiating pre-packaged, ready-to-go services.
- Truly distributed, on-demand, Cloud storage.
- Not everybody "speaks Grid": providing access to distributed, traditional Grid infrastructures as if they were not Grids, also to non-traditional users, like Public Administrations, or to the private sector.

The key problem is one of integration between several access interfaces (Grid, Cloud, or else).

Examples of services requested *today*

Some of the typical *new* service requests:

- Customer-definable software environments. This is a feature that finds several uses in "traditional" Grids as well.
- Setting up dynamic pools of virtual servers (e.g., user interfaces, or worker nodes for parallel interactive analysis). More generally, flexibly allocating hardware resources through complex advance-reservation requests.
- Instantiating pre-packaged, ready-to-go services.
- Truly distributed, on-demand, Cloud storage.
- Not everybody "speaks Grid": providing access to distributed, traditional Grid infrastructures as if they were not Grids, also to non-traditional users, like Public Administrations, or to the private sector.

The key problem is one of integration between several access interfaces (Grid, Cloud, or else).

Examples of services requested *today*

Some of the typical *new* service requests:

- Customer-definable software environments. This is a feature that finds several uses in "traditional" Grids as well.
- Setting up dynamic pools of virtual servers (e.g., user interfaces, or worker nodes for parallel interactive analysis). More generally, flexibly allocating hardware resources through complex advance-reservation requests.
- Instantiating pre-packaged, ready-to-go services.
- Truly distributed, on-demand, Cloud storage.
- Not everybody "speaks Grid": providing access to distributed, traditional Grid infrastructures as if they were not Grids, also to non-traditional users, like Public Administrations, or to the private sector.

The key problem is one of integration between several access interfaces (Grid, Cloud, or else).

Examples of services requested *today*

Some of the typical *new* service requests:

- Customer-definable software environments. This is a feature that finds several uses in "traditional" Grids as well.
- Setting up dynamic pools of virtual servers (e.g., user interfaces, or worker nodes for parallel interactive analysis). More generally, flexibly allocating hardware resources through complex advance-reservation requests.
- Instantiating pre-packaged, ready-to-go services.
- Truly distributed, on-demand, Cloud storage.
- Not everybody "speaks Grid": providing access to distributed, traditional Grid infrastructures as if they were not Grids, also to non-traditional users, like Public Administrations, or to the private sector.

The key problem is one of integration between several access interfaces (Grid, Cloud, or else).

Examples of services requested *today*

Some of the typical *new* service requests:

- Customer-definable software environments. This is a feature that finds several uses in "traditional" Grids as well.
- Setting up dynamic pools of virtual servers (e.g., user interfaces, or worker nodes for parallel interactive analysis). More generally, flexibly allocating hardware resources through complex advance-reservation requests.
- Instantiating pre-packaged, ready-to-go services.
- Truly distributed, on-demand, Cloud storage.
- Not everybody "speaks Grid": providing access to distributed, traditional Grid infrastructures as if they were not Grids, also to non-traditional users, like Public Administrations, or to the private sector.

The key problem is one of integration between several access interfaces (Grid, Cloud, or else).

Common grounds

Grids and Clouds (abstracting from the concept of "a Grid job", which one should regard as an implementation detail) basically target the use of resources.

The two terms come from different grounds, but really they are just different interfaces to access resources.

- Users may actually benefit joining an existing infrastructure, rather than building (or "buying") a new one.
    - This may actually not be a user’s choice.
- Sharing of data and resources across Grid/Cloud interfaces should be encouraged.
- Leveraging on multi-year investments and know-how on Grids to incrementally evolve and build new services is a strategic decision.
- Grids like EGEE are production infrastructures, serving the scientific needs of many (big and small) research communities.

The question is then how you do integrate Grids and Clouds.

Outline

1. Clouds and Grids: A Very Short Introduction
2. Cloud Access in WNoDeS
3. Summary

A practical Grid/Cloud integration example: WNoDeS

The **Worker Nodes on Demands Service** (WNoDeS) is a software INFN is developing. It is built around a tight integration with a LRMS (a "batch system") and is running in production at the INFN Tier-1 Computing Center. Its main characteristics are:

- Full integration with existing computing resource scheduling, policing, monitoring and accounting workflows.
- On-demand virtual resource provisioning and VLAN support to dynamically isolate Virtual Machines depending on service type / customer requests.
- Support for users to select and access WNoDeS-based resources through Grid, Cloud interfaces, or also through direct job submissions.

The WNoDeS focus is on *Everything as a Service*, where *Everything* may be hardware, software, data, platform, infrastructure.

A practical Grid/Cloud integration example: WNoDeS

The **W**orker **N**odes **o**n **De**mands **S**ervice (WNoDeS) is a software INFN is developing. It is built around a tight integration with a LRMS (a "batch system") and is running in production at the INFN Tier-1 Computing Center. Its main characteristics are:

- Full integration with existing computing resource scheduling, policing, monitoring and accounting workflows.
- On-demand virtual resource provisioning and VLAN support to dynamically isolate Virtual Machines depending on service type / customer requests.
- Support for users to select and access WNoDeS-based resources through Grid, Cloud interfaces, or also through direct job submissions.

The WNoDeS focus is on *Everything as a Service*, where *Everything* may be hardware, software, data, platform, infrastructure.

A practical Grid/Cloud integration example: WNoDeS

The **Worker Nodes on Demands Service** (WNoDeS) is a software INFN is developing. It is built around a tight integration with a LRMS (a "batch system") and is running in production at the INFN Tier-1 Computing Center. Its main characteristics are:

- Full integration with existing computing resource scheduling, policing, monitoring and accounting workflows.
- On-demand virtual resource provisioning and VLAN support to dynamically isolate Virtual Machines depending on service type / customer requests.
- Support for users to select and access WNoDeS-based resources through Grid, Cloud interfaces, or also through direct job submissions.

The WNoDeS focus is on *Everything as a Service*, where *Everything* may be hardware, software, data, platform, infrastructure.

A practical Grid/Cloud integration example: WNoDeS

The **Worker Nodes on Demands Service** (WNoDeS) is a software INFN is developing. It is built around a tight integration with a LRMS (a "batch system") and is running in production at the INFN Tier-1 Computing Center. Its main characteristics are:

- Full integration with existing computing resource scheduling, policing, monitoring and accounting workflows.
- On-demand virtual resource provisioning and VLAN support to dynamically isolate Virtual Machines depending on service type / customer requests.
- Support for users to select and access WNoDeS-based resources through Grid, Cloud interfaces, or also through direct job submissions.

The WNoDeS focus is on *Everything as a* Service, where *Everything* may be hardware, software, data, platform, infrastructure.

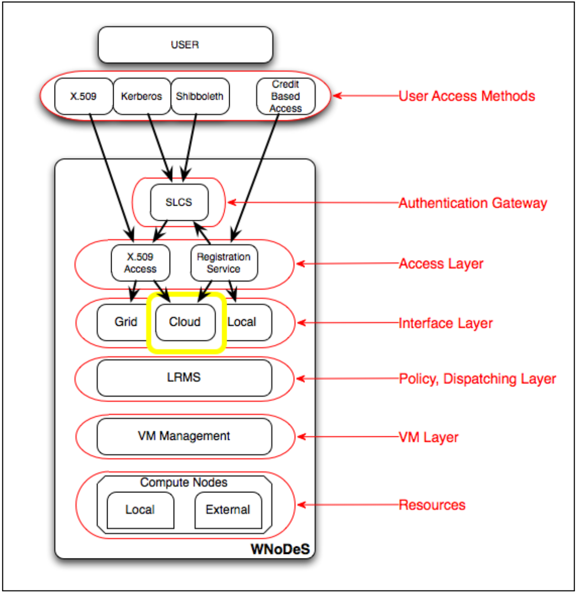
A practical Grid/Cloud integration example: WNoDeS

The **W**orker **N**odes **o**n **De**mands **S**ervice (WNoDeS) is a software INFN is developing. It is built around a tight integration with a LRMS (a "batch system") and is running in production at the INFN Tier-1 Computing Center. Its main characteristics are:

- Full integration with existing computing resource scheduling, policing, monitoring and accounting workflows.
- On-demand virtual resource provisioning and VLAN support to dynamically isolate Virtual Machines depending on service type / customer requests.
- Support for users to select and access WNoDeS-based resources through Grid, Cloud interfaces, or also through direct job submissions.

The WNoDeS focus is on *Everything as a Service*, where *Everything* may be hardware, software, data, platform, infrastructure.

WNoDeS: overall architectural framework



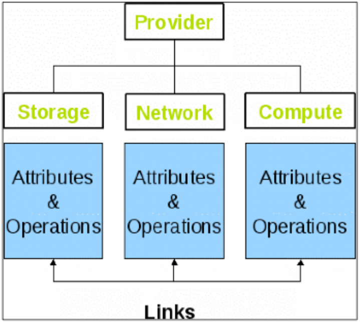
Key points

1. On-demand virtual service provisioning
2. Flexible, integrated scheduling policies
3. Multiple access interfaces
4. Multiple authentication methods
5. Integrated access to existing (e.g., Grid) infrastructures
6. Access to external resources

The OGF Open Cloud Computing Interface

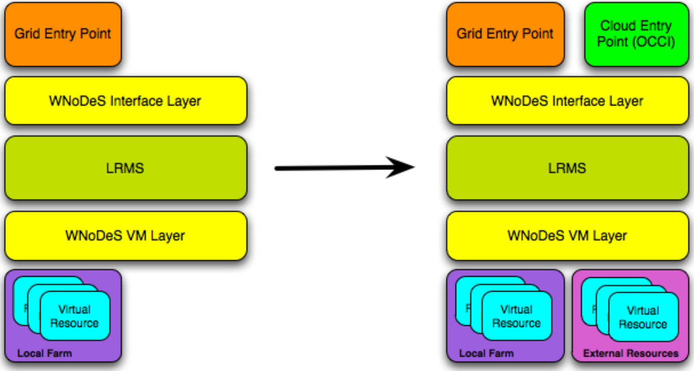
The Open Cloud Computing Interface (OCCI) API is being developed within the Open Grid Forum to access "Infrastructure as a Service" (IaaS) based Clouds.

It is a slim RESTful based API, allowing users to access and manage (computing, storage, network) resources using a Uniform Resource Identifier (URI).

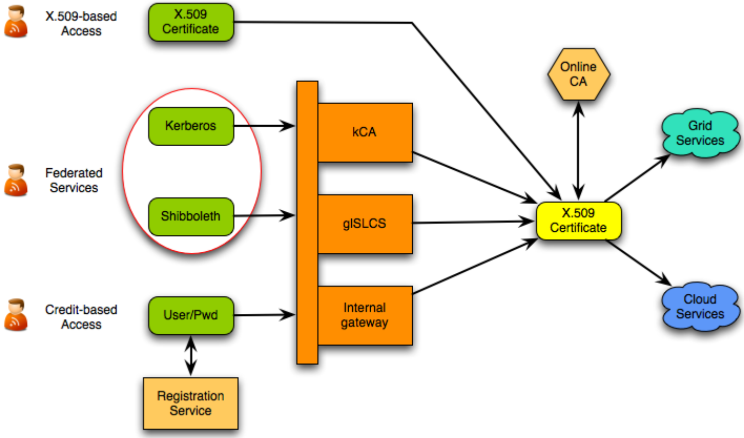


Integrating Cloud services

WNoDeS delivers access to Cloud services through the Open Cloud Computing Interface, implementing a subset of the OCCI API, using X.509 authentication and exposing a REST interface.



The WNoDeS authentication gateway



See talk by V. Ciaschini on the integration of VOMS/ARGUS/AA on Wednesday.

WNoDeS Cloud Access today

The WNoDeS Cloud RESTful web service is accessible via a basic alpha-stage Web application (see demo).

At the moment, upon a VM deployment request a "dummy job" is sent to the LRMS, and from there to a bait; eventually the dummy job runs on the allocated Cloud VM.

This has some disadvantages (like having the Cloud VM to be part of the LRMS cluster). The next WNoDeS version will support Cloud VM instantiations so that:

- A Cloud VM is totally oblivious of the LRMS.
- Control of the Cloud VMs is fully distributed, and consistency is ensured by the baits.

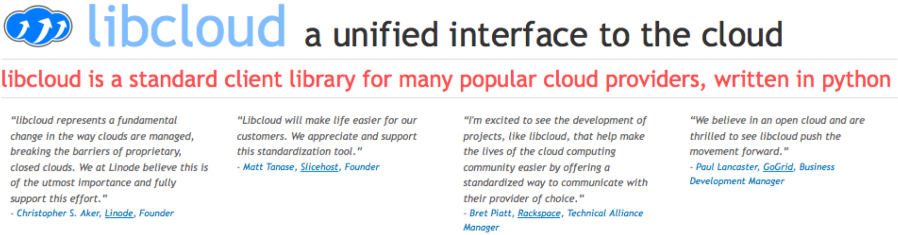
In case of Cloud allocations, only wallclock time is considered for accounting purposes.

WNoDeS Cloud Web App enhancements

Apart from the new architecture for instantiation of Cloud VMs, the following enhancements are targeted for the first WNoDeS public release (3Q10):

- Support for private/public `ssh` keys (using `libguestfs` to change VM images on the fly).
- Support for the OCCI commands deploy, start, stop, restart.
- VM image selection (only a pre-defined image is currently available.)
- VOMS and authentication gateway integration.
- Network parameter: requested subnet (public, private.)
- Network parameter: requested throughput.
- Virtual storage (details still to be worked out for this item.)

Future Cloud enhancement: `libcloud` support



libcloud a unified interface to the cloud

libcloud is a standard client library for many popular cloud providers, written in python

"libcloud represents a fundamental change in the way clouds are managed, breaking the barriers of proprietary, closed clouds. We at Linode believe this is of the utmost importance and fully support this effort."
- Christopher S. Aker, Linode, Founder

"Libcloud will make life easier for our customers. We appreciate and support this standardization tool."
- Matt Tanase, Slicehost, Founder

"I'm excited to see the development of projects, like libcloud, that help make the lives of the cloud computing community easier by offering a standardized way to communicate with their provider of choice."
- Bret Piatt, Rackspace, Technical Alliance Manager

"We believe in an open cloud and are thrilled to see libcloud push the movement forward."
- Paul Lancaster, GoGrid, Business Development Manager

`libcloud` will allow the possibility to access WNoDeS Cloud capabilities not only via the Web App, but also via Python scripts. It will then be possible to programmatically instantiate and manage Cloud VMs.

Link: `http://incubator.apache.org/libcloud`

libcloud simple usage example

```python
from libcloud.types import Provider
from libcloud.providers import get_driver
from libcloud.deployment import MultiStepDeployment, ScriptDeployment, SSHKeyDeployment

RACKSPACE_USER = 'your username'
RACKSPACE_KEY = 'your key'

Driver = get_driver(Provider.RACKSPACE)
conn = Driver(RACKSPACE_USER, RACKSPACE_KEY)

# read your public key in
sd = SSHKeyDeployment(open("~/.ssh/id_dsa.pub").read())
# a simple script to install puppet post boot, can be much more complicated.
script = ScriptDeployment("apt-get install puppet")
# a task that first installs the ssh key, and then runs the script
msd = MultiStepDeployment([sd, script])

images = conn.list_images()
sizes = conn.list_sizes()

# deploy_node takes the same base keyword arguments as create_node.
node = conn.deploy_node(name='test', image=images[0], size=size[0], deploy=msd)
# <Node: uuid=..., name=test, state=3, public_ip=['1.1.1.1'], provider=Rackspace ...>
# the node is now booted, with your ssh key and puppet installed.
```

Outline

1. Clouds and Grids: A Very Short Introduction
2. Cloud Access in WNoDeS
3. Summary

So, what is this for?

Beyond the use cases mentioned at the beginning, some examples:

- A prototype of a virtual analysis facility: First experiences (*S Bagnasco et al 2010 J. Phys.: Conf. Ser. 219 062033*), see `http://goo.gl/Fs0S`

  > *[. . .] Leveraging on the virtualization of highly performant multi-core machines it is possible to build a fully virtual analysis facility on the same Worker Nodes that compose an existing LCG Grid Farm. [. . .]*

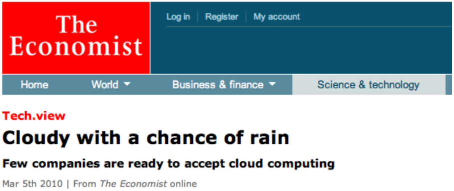
  (BTW: neither resource accounting, nor the static definition of the PROOF nodes, mentioned as problematic in the article above, are issues in WNoDeS)

- The WNoDeS software as a tool to support virtual pools of servers for interactive analysis and software development (*Submitted to CHEP10*)

Several work is still to be done especially in the I/O area, but premises are encouraging.

A couple of questions (1)

- Do we still need or want an e-Infrastructure for scientific computing? Do *you* care?



Tech.view

Cloudy with a chance of rain

Few companies are ready to accept cloud computing

Mar 5th 2010 | From *The Economist* online

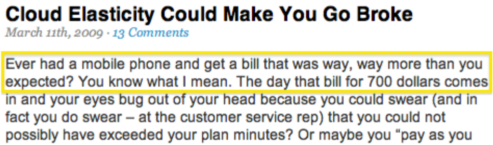
A recent poll by CommVault identified the following as the main obstacles or worries for cloud adoption:

- Security and privacy
- Reliability
- Cost
- Scalability (are there *really* infinite resources?)

As users (perhaps belonging to established communities), is it possible to formalize what our *difference* is? And as *providers*?

A couple of questions (2)

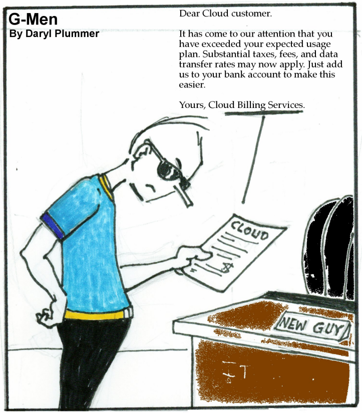
For example, elasticity is a great thing, but. . .



Cloud Elasticity Could Make You Go Broke

March 11th, 2009 · 13 Comments

Ever had a mobile phone and get a bill that was way, way more than you expected? You know what I mean. The day that bill for 700 dollars comes in and your eyes bug out of your head because you could swear (and in fact you do swear – at the customer service rep) that you could not possibly have exceeded your plan minutes? Or maybe you "pay as you


G-Men
By Daryl Plummer
Dear Cloud customer.
It has come to our attention that you have exceeded your expected usage plan. Substantial taxes, fees, and data transfer rates may now apply. Just add us to your bank account to make this easier.
Yours, Cloud Billing Services.
CLOUD
$
NEW GUY

(Daryl Plummer, Gartner) – i.e., **On-Demand Overspending**

That’s it – It’s Shoppable (although perhaps not yet shippable)

> *[Cloud computing is] nothing more than a faddish term for the established concept of computers linked by networks. A cloud is water vapor. (Larry Ellison, co-founder and CEO, Oracle Corporation, September 2009)*

> *The truth is rarely pure and never simple. (Oscar Wilde, The Importance of Being Earnest, 1895)*

Thanks!

e-mail: `Davide.Salomoni@cnaf.infn.it`