



Cloud-like Management of Grid Sites 2.0 Software

Henar Muñoz Frutos, Eduardo Huedo, Rubén Montero, Charles Loomis

► To cite this version:

Henar Muñoz Frutos, Eduardo Huedo, Rubén Montero, Charles Loomis. Cloud-like Management of Grid Sites 2.0 Software. 2012. hal-00696646v2

HAL Id: hal-00696646

<https://hal.science/hal-00696646v2>

Submitted on 15 May 2012

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Enhancing Grid Infrastructures with
Virtualization and Cloud Technologies

Cloud-like Management of Grid Sites 2.0 Software

Deliverable D6.5 (V1.0)
11 May 2012

Abstract

This document presents the features implemented for the automatic deployment and dynamic provision of grid services, and for the scalable cloud-like management of grid site resources in the second year of the project. These features, developed largely in Work Package 6 (WP6), are integrated into the StratusLab Toolkit by Work Package 4 (WP4).



StratusLab is co-funded by the
European Community's Seventh
Framework Programme (Capacities)
Grant Agreement INFSO-RI-261552.



The information contained in this document represents the views of the copyright holders as of the date such views are published.

THE INFORMATION CONTAINED IN THIS DOCUMENT IS PROVIDED BY THE COPYRIGHT HOLDERS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE MEMBERS OF THE STRATUSLAB COLLABORATION, INCLUDING THE COPYRIGHT HOLDERS, OR THE EUROPEAN COMMISSION BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THE INFORMATION CONTAINED IN THIS DOCUMENT, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright © 2012, Members of the StratusLab collaboration: Centre National de la Recherche Scientifique, Universidad Complutense de Madrid, Greek Research and Technology Network S.A., SixSq Sàrl, Telefónica Investigación y Desarrollo SA, and The Provost Fellows and Scholars of the College of the Holy and Undivided Trinity of Queen Elizabeth Near Dublin.

This work is licensed under a Creative Commons Attribution 3.0 Unported License
<http://creativecommons.org/licenses/by/3.0/>



Contributors

Name	Partner	Sections
Muñoz Frutos, Henar	TID	2, 3, 5
Huedo, Eduardo	UCM	2, 3, 4
Montero, Rubén S.	UCM	3, 4
Loomis, Charles	LAL	General and English Revision

Document History

Version	Date	Comment
0.1	18 Apr. 2012	Initial version for comment.
0.2	28 Apr. 2011	Contribution to implemented features
0.3	28 Apr. 2011	Contribution to no implemented features
0.4	30 May 2011	Executive Summary, Introduction and Conclusions
0.5	9 May 2011	Peer revision
0.6	10 May 2011	Contribution towards the peer revision comments
1.0	11 May 2011	Final version

Contents

List of Figures	5
List of Tables	6
1 Executive Summary	7
2 Introduction	8
2.1 Organization of Following Chapters	9
3 Implemented Features	10
3.1 Interoperability	10
3.1.1 Cloud like APIs	10
3.1.2 Virtual Appliance (Service) Language Definition	10
3.2 Service Manager	10
3.3 VM Manager	11
3.4 Network Manager	11
3.5 Storage Manager	11
3.6 Monitoring and Accounting	12
3.7 Inter-cloud Connector	12
4 Missing Features	16
5 Conclusions	17
References	20

List of Figures

2.1	StratusLab Architecture	9
-----	-----------------------------------	---

List of Tables

3.1	Implemented Features Summary	13
-----	--	----

1 Executive Summary

The present document describes the software implemented in WP6 during year 2, whose design was specified in the deliverable D6.4 *Cloud-like Management of Grid Sites 4.0 Design Report* [4].

Concretely, TCloud [5], OCCI [3] and Deltacloud [1] APIs have been incorporated into StratusLab as cloud-like APIs for accessing the Service Manager, Virtual Machine Manager and for storage and networking capabilities. The Open Virtualization Format (OVF) [2] was selected as the service definition language and used also for contextualization. The Claudia framework [6] was integrated to handle service management and scalability including new scalability rules identified by use cases. New storage and network capabilities have been developed in WP6 as advanced features integrated in OpenNebula [7]. Finally, an Inter-cloud Connector module has been introduced for working on cloud bursting, federation and brokering.

Most of these implemented features, developed in WP6, have been integrated into the StratusLab distribution with the help of WP4. This means that the implemented features have been stored in the StratusLab git repositories, accessible via the RPM package repository and tested via jobs in hudson.

This document presents the features implemented for the automatic deployment and dynamic provision of grid services, and for the scalable cloud-like management of grid site resources. It describes the implemented features and where the software, installation and configuration instructions and end-user documentation can be found.

2 Introduction

Grid applications deployed over IaaS clouds can benefit from a set of features that StratusLab provides in the release 2.0: i) interoperability of APIs and application definition, ii) new scaling rules for the Service Manager, iii) networking capabilities, iv) storage management, v) monitoring of the application and accounting and vi) federation via the Inter-cloud Connector.

With respect to the features represented in StratuLab architecture version 2.0, shown in Figure 2.1, the following document specifies the work carried out in WP6 regarding the following topics:

- The inclusion of cloud-like interfaces and a language based on standards to increase the interoperability in cloud service management.
- The introduction of a service manager, on top of the current virtual machine manager, which is able to control and configure grid services as a whole providing scalability mechanism at the service-level (as part of the service manager) to scale up and down grid service components according to some Key Performance Indicators (KPIs) and hardware usage.
- Advanced networking capabilities that provide added features such that users can more dynamically create and configure deployment-specific virtual networks, in order to provide finer control and isolation of their system deployed in the cloud.
- Storage capabilities such as the OpenNebula datastores. There are other components related to storage capabilities in the StratusLab architecture, like the Marketplace and the Persistent Disk Service, but they have not been implemented as part of WP6.
- Monitoring systems at the physical, virtual hardware and grid service layers for regular administration tasks, accounting purposes and for triggering scalability mechanisms.
- The Inter-cloud Connector as part of Stratuslab for providing federation capabilities interfacing a given cloud site with another.

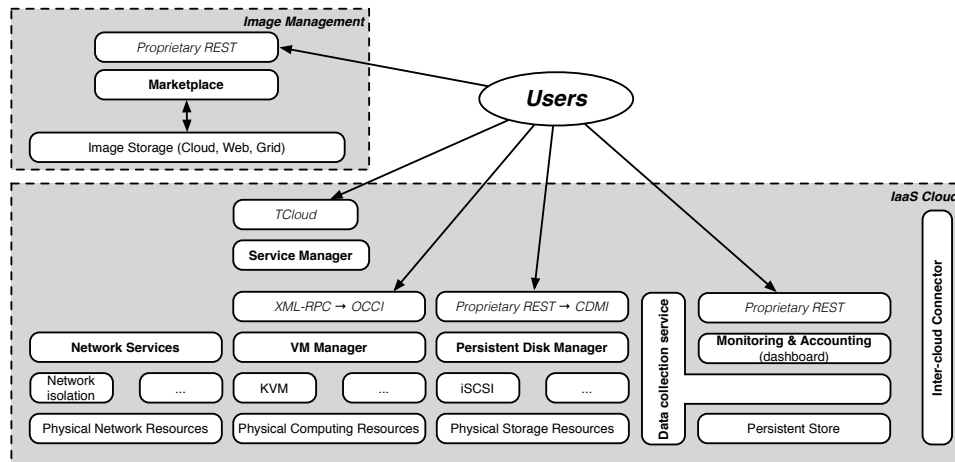


Figure 2.1: StratusLab Architecture

2.1 Organization of Following Chapters

The document is organized as follows: Chapter 3 provides a summary of the implemented features in WP6 with respect to the previous topics (defined in D6.4 [4]). Missing features are discussed in Chapter 4. Finally, the Chapter 5 provides some conclusions.

3 Implemented Features

This section provides an overview of the features implemented in StratusLab with respect to automatic deployment and dynamic provision of grid services, as well as scalable cloud-like management of grid site resources. Table 3.1 presents an overview of the developed functionalities. This table lists the functionality, the repository where the software is located, the RPM packages which contain the functionality, whether the functionality is part of the StratusLab 2.0 release, administrator documentation, user documentation, and the contribution from within WP6. The table summarizes the implemented features which are described in the following sections.

3.1 Interoperability

Interoperability allows users to access different sites uniformly and to reuse running virtual images. StratusLab is focused on the use of Cloud-like API and Virtual Appliance (Service) Language Definition.

3.1.1 Cloud like APIs

The Cloud-like APIs in StratusLab contain the TCloud API for the Service Manager and monitoring, OCCI for VM, network and image management, and Deltacloud (replacing the planned jClouds API) for VM and image management.

3.1.2 Virtual Appliance (Service) Language Definition

The Open Virtualization Format (OVF) [2] is the format used to specify virtual appliances. With respect to the standard, new extensions have been included in OVF to cover StratusLab requirements, like scalability and load balancing. StratusLab provides a library to process OVF files as part of the Claudia distribution software. The `ovf-manager` is the OVF processing library which is part of the Claudia distribution.

3.2 Service Manager

The Service Manager that has been integrated in StratusLab is named Claudia. Claudia acts as a middleware for IaaS clouds that will enable service-level provisioning, automated deployment, scalability for the service (together with new scalability rules) and integration with monitoring. New scalability rules have been

added in the Claudia implementation to specify the number of nodes to be scaled or some scaling down policies like the lazy scaling down.

3.3 VM Manager

As Virtual Machine Manager, StratusLab includes OpenNebula 3.2, which has been considerably improved during the project, including support for groups and ACLs, datacenter placement policies, improved security, new authentication systems, and VM templates.

Moreover, OpenNebula provides functionality to manage networks and storage, for monitoring and accounting, and to build hybrid or federated clouds. That functionality is described in the following sections.

3.4 Network Manager

StratusLab incorporates the abstraction needed to extend the VM lifecycle model to add networking aspects. This generic approach allows the management component of StratusLab, OpenNebula, to interact with any networking fabric. This functionality is provided through OpenNebula network drivers, that perform configuration operations before the VM is launched, after it is booted and when it is destroyed. The StratusLab distribution includes drivers to interact with common switching facilities: IEEE 802.1Q and Open vSwitch.

Additionally StratusLab provides support for enabling simple firewalling rules to allow a regular user to filter TCP, UDP or ICMP traffic. This firewall rules are set up on-the-fly, outside the VM, so it does not require any intervention of the user or special firewalling software to be installed in the VM.

3.5 Storage Manager

StratusLab features multiple components to manage storage, namely: the Marketplace, the Persistent Disk Service, and specialized drivers to integrate these components with OpenNebula.

The Marketplace serves as a searchable registry for shared machine and disk images. The Persistent Disk Service provides on-site management of persistent storage areas allowing users to create disks of a given size, to launch a machine with the disks attached, and to persist the disks after their use, so that the disks are available for use by another machine instance. These components have been developed outside WP6.

The storage capabilities of OpenNebula have been also improved in StratusLab, with the addition of the Datastore abstraction. A datastore is an abstraction of any storage medium for VM disk images. Datastores are distributed to the hosts with specific transfer (TM) drivers (shared, ssh, iscsi, vmware or qcow). This allows a single host to include multiple datastores of different types.

OpenNebula 3.4 includes several datastore types: file-system based, iSCSI/LCM and VMware. Moreover the temporal disk images or copies are placed in a special

datastore, the system datastore, that can be also distributed among the hosts with different transfer drivers. However, in StratusLab 2.0, which includes OpenNebula 3.2, the Marketplace and the Persistent Disk Service are still integrated using transfer drivers.

3.6 Monitoring and Accounting

StratusLab provides monitoring systems to constantly check the performance of the system. These monitoring systems are able to collect metrics about hardware, software and service, which are used later for scalability. The monitoring systems are based on probes (collectd software) introduced in the VM by contextualization, a collector and aggregator of this monitoring information, a database which stores all the data and an API.

For resource accounting, StratusLab relies on the underlying capabilities of OpenNebula. By querying the accounting database administrators can retrieve information regarding resource usage per user and can easily correlate it with the hosted services running inside the virtualized resources (e.g. a grid site on a StratusLab based cloud).

3.7 Inter-cloud Connector

StratusLab provides the Inter-cloud Connector, which allows the Multi-Cloud scenario, where multiple external and internal cloud computing services are deployed and managed to match business needs. The Inter-cloud Connector allows instantiation of VMs on public clouds (like Amazon EC2) as well as partner clouds (like other StratusLab sites). The component abstracts the external clouds APIs by providing a common interface to them that can be plugged into the VM Manager component allowing a hybrid or federated cloud approach (where the VM Manager decides to outsource computation to another public or partner cloud, respectively) or the Service Manager component with a cloud brokering approach (when the VM is deployed in different cloud providers according to user policies).

The following table shows the status of the Management of Grid Services and Resources towards the release 2.0.

Table 3.1: *Implemented Features Summary*

Topic	Repository	Packages	In 2.0?	Admin Doc.	User Doc.	WP6 work
TCloud API	git://github.com/StratusLab/claudia.git	tcloud-server-rpm	Yes	http://stratuslab.eu/doku.php/claudia	http://claudia.morfeo-project.org/wiki/index.php/TCloud_Server	Evolved existing code
OCCI API	git://git.opennebula.org/one.git	one-3.2-StratusLab	Yes	http://opennebula.org/documentation:rel3.4:occicg	http://opennebula.org/documentation:rel3.4:occiug	Evolved existing code
Deltacloud API	git://git.apache.org/deltacloud.git	deltacloud-core	No ¹	http://wiki.opennebula.org/deltacloud	http://wiki.opennebula.org/deltacloud#using_deltacloud	Evolved existing code
OVF format	git://github.com/StratusLab/claudia.git	clotho-rpm	Yes	http://claudia.morfeo-project.org/wiki/index.php/OVF_Manager	http://claudia.morfeo-project.org/wiki/index.php/OVF_Manager	Taken from Claudia
Service Manager (Claudia)	git://github.com/StratusLab/claudia.git	clotho-rpm, tcloud-server-rpm, claudia-commons, claudia-client-rpm	Yes	http://stratuslab.eu/doku.php/claudia	http://stratuslab.eu/doku.php/claudia#claudia_stratuslab_test	Evolving in WP6

¹It is part of Deltacloud.

Scaling policies	git://github.com/StratusLab/claudia.git	part of clotho-rpm	Yes	http://stratuslab.eu/doku.php/claudia		New ones in WP6
VM Manager (OpenNebula)	git://git.opennebula.org/one.git	one-3.2-StratusLab	Yes	http://opennebula.org/documentation:rel3.4:vmmg	http://opennebula.org/documentation:rel3.4:vm_guide	Evolved existing code
Network Manager (OpenNebula)	git://git.opennebula.org/one.git	one-3.2-StratusLab	Yes	http://opennebula.org/documentation:rel3.4:nm	http://opennebula.org/documentation:rel3.4:vgg	Developed in WP6
Storage Manager (OpenNebula)	git://git.opennebula.org/one.git	one-3.4	No ²	http://opennebula.org/documentation:rel3.4:sm	http://opennebula.org/documentation:rel3.4:img_guide	Developed in WP6
Monitoring	git://github.com/StratusLab/monitoring.git	collectd-base, collectd-basicprobes, collectd-java	Yes	http://stratuslab.eu/doku.php/monitoringsystems	http://stratuslab.eu/doku.php/monitoringsystems	Developed in WP6
Accounting	git://git.opennebula.org/one.git	one-3.2-StratusLab	Yes	Not applicable	http://opennebula.org/documentation:rel3.4:accounting	Evolved existing code
OpenNebula Connector	git://github.com/StratusLab/one.git	one-3.2-StratusLab	Yes			Developed in WP6

²It is part of OpenNebula 3.4, while StratusLab 2.0 includes OpenNebula 3.2.

Amazon EC2 Connector	git://git.opennebula.org/one.git	one-3.2-StratusLab	Yes	http://opennebula.org/documentation:rel3.4:ec2g	Not applicable	Evolved existing code
Claudia Broker	git://github.com/StratusLab/claudia.git	placement-rpm	Yes		Not applicable	Evolved existing code

4 Missing Features

The jClouds API was finally replaced by the Deltacloud API. Both of them can be defined as aggregator APIs, but there were some existing code for the latter one, which had to be adapted for OpenNebula 3.2 and 3.4.

Also, there were plans to evaluate and adopt solutions from other projects, like CDMI-Proxy from Venus-C for storage, accounting and billing system also from Venus-C, and OpenNAAS from Mantychore, providing IP Network as a Service. However, these solutions were released too late for StratusLab to evaluate them. For example, OpenNAAS 0.8 was released in April 2012, CDMI-Proxy 2.0 in April 2012, and the 1.0 version of the accounting solution in March 2012.

5 Conclusions

This document has provided an overview of the functionalities implemented within the scope of WP6 in the second period of the project.

Some of the topics specified in this document, developed in WP6, have been integrated as part of the Stratuslab v2.0 distribution. To obtain this objective, WP6 developments have been integrated as part of the StratusLab development infrastructure (git repositories, hudson jobs, production of RPM packages etc.). In addition, some command line client tools have been developed in WP4 for the automatic configuration and installation of these components as part of the StratusLab distribution. Finally, these developments have been deployed in Stratuslab testbeds and tested via use cases (part of WP2).

Glossary

APEL	Account Processor for Event Logs
Appliance	Virtual machine containing preconfigured software or services
Appliance Repository	Repository of existing appliances
CDDL	Configuration Description, Deployment, and Lifecycle Management
DGAS	Distributed Grid Accounting System
DHCP	Dynamic Host Configuration Protocol
DMTF	Distributed Management Task Force
Front-End	OpenNebula server machine, which hosts the VM manager
Hybrid Cloud	Cloud infrastructure that federates resources between organizations
IaaS	Infrastructure as a Service
IP	Infrastructure Provider
Instance	a deployed Virtual Machine
JRA	Joint Research Activity
KPI	Key Performance Indicator
Machine Image	Virtual machine file and metadata providing the source for Virtual Images or Instances
NFS	Network File System
Node	Physical host on which VMs are instantiated
OASIS	Organization for the Advancement of Structured Information Standards
OCCI	Open Cloud Computing Initiative
OGF	Open Grid Forum
OVF	Open Virtualization Format
Public Cloud	Cloud infrastructure accessible to people outside of the provider's organization
Private Cloud	Cloud infrastructure accessible only to the provider's users
Regression	Features previously working which breaks in a new release of the software containing this feature
Service Manager/SM	A toolkit to provides Service Providers to dynamically control the Service provisioning and scalability
Service Provider/SP	The provider who offers the application to be deploy in the Cloud
SMI	Service Manager Interface
SSD	Solution Deployment Descriptor

VApp	Virtual Appliance: pre-configured software stacks comprising one or more virtual machines to provide self-contained services
Virtual Machine / VM	Running and virtualized operating system
VMI	Virtual Manager Interface
VO	Virtual Organization
VOMS	Virtual Organization Membership Service
Web Monitor	Web application providing basic monitoring of a single StratusLab installation
Worker Node	Grid node on which jobs are executed

References

- [1] Apache Software Foundation. Apache Deltacloud. Online resource, 2012. <http://deltacloud.apache.org/>.
- [2] DMTF. Open virtualization format specification. Specification DSP0243 v1.0.0d. Technical report, Distributed Management Task Force, Sep 2008. <https://www.coin-or.org/OS/publications/optimizationServicesFramework2008.pdf>.
- [3] OCCI-WG. Open Cloud Computing Interface Specification. Technical report, Open Grid Forum, 2009. <http://forge.ogf.org/sf/go/doc15731>.
- [4] Stratuslab Consortium. Deliverable 6.4 Cloud-like Management of Grid Sites 2.0 Design Report. Online resource., 2011. <http://stratuslab.eu/lib/exe/fetch.php/documents:stratuslab-d6.4-v2.0.pdf>.
- [5] Telefónica. TCloud API Specification, Version 0.9.0. Online resource., 2010. http://www.tid.es/files/doc/apis/TCloud_API_Spec_v0.9.pdf.
- [6] TID. The Claudia project. Online resource., 2012. <http://claudia.morfeo-project.org>.
- [7] UCM. OpenNebula: The Open Source Solution for Data Center Virtualization. Online resource., 2012. <http://www.opennebula.org>.