StratusLab

Enhancing Grid Infrastructures with
Virtualization and Cloud Technologies

Release of Cloud-like Management of Grid Services and Resources 1.0 Beta

Milestone MS14 (V1.2)
14 March 2011

Abstract

A Service Manager component was introduced on top of the IaaS cloud, so that grid services can be dynamically provisioned as a single entity, concealing low-level details of the physical infrastructure. This milestone was achieved with the integration of the Service Manager in the StratusLab distribution.

CAPACITIES

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.

Contributors

Name	Partner	Sections
Jose Lopez	TID	All

Document History

Version	Date	Comment
0.1	1 Mar. 2011	Skeleton of milestone.
0.2	11 Mar. 2011	Draft for internal review.
1.0	11 Mar. 2011	Version after internal review.
1.1	14 Mar. 2011	Updated version.
1.2	14 Mar. 2011	Final version after reviewer comments.

1 Introduction

Grid applications deployed over IaaS clouds can benefit from a Service Management layer on top of the Virtual Infrastructure Management layer. This solution conceals low-level details from the user and provides a wider range of scalability mechanisms and a broader set of actions over the cloud than a regular IaaS cloud.

The next sections deal with the main capabilities of the Service Manager, how the integration of the Service Manager in the Stratuslab distribution has been done so far, and what steps are needed for the Service Manager to become part of the StratusLab v1.0 release.

2 Cloud-like Management of Grid Services and Resources

2.1 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 the following capabilities:

Service-level provisioning The basic unit for provision is the service to be deployed. A service is a set of Virtual Machines playing different roles, with their specific needs for storage and connectivity. Note the special use of the term *service* in this context, not to be confused with its common meaning as a piece of server software (e.g. a Unix system service).

Automated deployments This is achieved with the creation and contextualization of Virtual Machines, and the orchestration provided by the Virtual Infrastructure Manager (OpenNebula).

Scalability The use of resources is optimized by scaling up/down the service applying elasticity rules at the service level.

Standard APIs Claudia is accessed via TCloud and OpenNebula is accessed via an XMLRPC interface and eventually OCCI.

Standard Service Definition Language Services, Virtual Machines and networks are described using Open Virtualization Format (OVF).

Monitoring OpenNebula was integrated with the Ganglia monitoring tool and fault tolerance mechanisms were developed to reduce service downtime.

2.2 Integration of the Service Manager

Claudia was integrated in the StratusLab release through the same continuous-integration process used by the rest of the components. Regarding the development procedures, the source code is uploaded to a Git repository, integrated using Hudson and packaged in a Yum repository. Claudia is composed of several components: Clotho (lifecycle manager), TCloud Server (communications hub implementing the TCloud API) and the Claudia Client (command-line client to deploy services, among other actions). Additional modules exist dealing with a driver for Clotho, a driver for OpenNebula, an Open Virtualization Format (OVF) manager, and common libraries. The packages are built using Maven (triggered by Hudson jobs). Three rpm packages are generated for Clotho, TCloud Server and Claudia Client.

Additional work was needed for the manual installation and configuration of Claudia using the StratusLab tool stratus-install. This task consisted of tailor-made Python scripts that dealt with the installation of the Claudia RPM packages, the creation of Claudia configuration files based on the configuration read from the StratusLab main configuration file, and the launch of the Claudia components. The configuration step must be finished because not all the configuration parameters needed by Claudia are present in the StratusLab configuration file; these must be added.

2.3 Steps towards the release

In order for Claudia to become part of the StratusLab v1.0 release, some additional steps are needed. First of all, integration with manual installation is not completely done and the configuration procedure that reads the StratusLab main configuration file must be fine tuned. Once this is achieved, Claudia must be integrated with Quattor to allow for automatic installation. Automatic tests have to be added to Hudson to complete the continuous-integration cycle. Finally, a use case deploying a grid application must be demonstrated.