# Summer Student Report - Summer 2013
# DBaaS with OpenStack Trove

Student: Andrea Giardini
Supervisors: Giacomo Tenaglia - Ewan Roche

# Project

The purpose of the project was to evaluate the Trove component for OpenStack, understand if it can be used with the CERN infrastructure and report the benefits and disadvantages of this software. Currently, databases for CERN projects are provided by a DbaaS software developed inside the IT-DB group. This solution works well with the actual infrastructure but it is not easy to maintain. With the migration of the CERN infrastructure to OpenStack the Database group started to evaluate the Trove component. Instead of mantaining an own DbaaS service it can be interesting to migrate everything to OpenStack and replace the actual DbaaS software with Trove. This way both virtual machines and databases will be managed by OpenStack itself.

# Trove - Database as a Service for OpenStack

> Trove is Database as a Service for OpenStack. It is designed to run entirely on OpenStack, with the goal of allowing users to quickly and easily utilize the features of a relational database without the burden of handling complex administrative tasks. Cloud users and database administrators can provision and manage multiple database instances as needed. Initially, the service will focus on providing resource isolation at high performance while automating complex administrative tasks including deployment, configuration, patching, backups, restores, and monitoring. [1]

As stated before the Trove component for OpenStack aims to provide a scalable and efficient service to supply database instances in the cloud, separating the administrative tasks from the applicative one. Users don't need to worry about system database administration or optimization because all the user related tasks can be done through the API access provided by the component. Common APIs are a unified entry point to access the databases. This way the user may access the database instance from a high-level point of view while the system administrator takes care of all the administrative tasks and server configuration.

# Work Summary

We started by installing a test environment, first on a virtual machine then on RAC50 (standard server with 2-sockets, 128GB Ram, 3x2TB HD - IT DB server), using the trove-integration script [2]. This script uses Devstack [3] to automatically build a complete test environment with OpenStack and Trove. The script is configured to automatically deploy the latest OpenStack version, but instead we needed a Grizzly environment for our tests: while in Devstack we have the possibily to select which version to deploy, this setting was completely ignored by the trove-integration script. So we had to write a new bug report [4], correct the script and submit our code to review [5] to have it also work with different OpenStack versions: at the moment it is possibile to deploy a complete Trove testing environment with a unique shell script. This can be very useful to easily test new components or features with multiple versions of OpenStack.

After that we encountered another problem with the instance creation: after building the instance the database did not become active, maintaining permanently the "Build" state. After some debugging we figured out that there was a problem with the home directory inside the cloud instance that had too restrictive permissions. So we created a new bug report [6] and a review [7] to solve the problem.

At this point everything was working fine on virtual machines. The deployment on the RAC50 hardware instead was not as easy as expected. The first issue that we faced was due to the operating system. Currently Trove supports only Ubuntu 12.04 LTS and the servers that we had (itrac50009 and itrac50064) were both running RHEL, unfortunately no Ubuntu images were available for the automatic installation so we had to manually install Ubuntu through debootstrap [8] as described in the Twiki page [9].

After installing Ubuntu on both of the machines we had the possibily to run the trove-integration script without any problems. We had to create a couple of ad-hoc iptables rules to let the virtual machine connect to the internet. At this point the test ambient was completely configured and running on RAC50: itrac50009 had the complete OpenStack suite with Trove and itrac50064 was configured as a compute node. However, the instances were able to communicate with the internet only with NAT, but this issue was expected. The CERN network has a very strict policy about MAC address registration so everything should work fine with the patched version of

Grizzly built by the CERN Cloud team.

During the configuration we filled a bug report to include the iptables commands in the trove-integration script [10] and another one for Devstack due to the installation of the wrong version of Django when Grizzly is selected [11].

In the last month we focused on providing an easy-to-use interface to manage the Trove instances through the traditional OpenStack dashboard. We found a draft for a Trove dashboard on GitHub that was not working, so we submitted some modifications and patches and, in collaboration with the author and another user, we got it working and fully functional. The list of the modifications that we made is publically available on GitHub [12]. Due to the lack of a functional dashboard inside the Horizon component we decided to try to merge our code with it, so we submitted a review for the Horizon component [13], in this way the dashboard will be available to everybody from the OpenStack Havana release within the Horizon component.

# Conclusions

The Trove project showed a great maturity and a very active community. A lot of features are ready to be merged and there is a lot of interest in this new OpenStack component. It integrates perfectly with the OpenStack environment providing APIs for an easy access, a dashboard for the databases management and script for testing. With the Trove component it is effectively possibile to centralize the management of virtual machines and database instances in a unique OpenStack cluster. It this way the CERN infrastructure team might be able to mantain both of those services without too much effort. Due to the API structure of the trove component it should be also very easy to move from the actual Dbod service to a trove-based service without any issue for the end user. Furthermore, we have to remember that the cloud image used to launch and configure the database instances is completely customizable, so with some scripts it should not be difficult to mount the corresponding NFS share and to install the database within it.

The Trove component still has some issues that need to be resolved. First of all the lack of support for RHEL hosts is blocking the implementation of the component in production: developers are actively working on it but, right now, still nothing is ready to be deployed. Secondly, we need support for different types of databases. Right now Mysql and Percona are supported but we also need to implement PostgreSQL and MongoDB because some users are actively using those kinds of databases. Thirdly, the lack of documentation: Trove completely lacks of every documentation, right now only a couple of wiki pages are available, due to this also the development of new tools is not so easy to accomplish.

# Acknowledgements

I spent a very good time here in CERN, it was an amazing experience and I learned a lot of things, the project was very interesting and I think that the environment here is probably unique in the world.

First of all, I would like to thank Giacomo for his amazing help and patience, he was a great supervisor to collaborate with: he taught me a lot of things and shared his own ideas about my project. Discussing together and evaluating how to proceed was really importart for me to learn how to organize my work and how to obtain the best results. I also would like to thank Maaike, Ewan, Artur and Arash, their help and tips were really helpful and their support and company were fundamental to make me feel as part of the group.

Finally, I would like to thank all the IT-DB group for this summer spent together, it was a great experience for me, as a student, to get in contact with such an amazing group, I hope to come back in the future and see you again.

# References


[1] https://wiki.openstack.org/wiki/Trove

[2] https://github.com/openstack/trove-integration/

[3] http://devstack.org/

[4] https://bugs.launchpad.net/trove-integration/+bug/1194793/

[5] https://review.openstack.org/#/c/35756/

[6] https://bugs.launchpad.net/trove-integration/+bug/1195786/

[7] https://review.openstack.org/#/c/35320/

[8] https://wiki.debian.org/it/Debootstrap/

[9] https://twiki.cern.ch/twiki/bin/viewauth/DB/Private/DBaaSwithOpenstackReddwarf (Accessible only to IT-DB members)

[10] https://bugs.launchpad.net/trove-integration/+bug/1194863

[11] https://bugs.launchpad.net/devstack/+bug/1209234

[12] https://github.com/rmyers/trove-dashboard/commits?author=AndreaGiardini

[13] https://review.openstack.org/#/c/42228/