OpenStack, the fastest-growing open cloud platform, is a highly scalable operating system for public and private clouds and is winning the support of service providers and organizations of all sizes. Trove is the "Database as a Service" (DBaaS) for OpenStack.

This Percona eBook introduces OpenStack, Trove and DBaaS; offers practical advice on overcoming issues associated with growth when using MySQL and OpenStack; provides a closer look at Percona Server, OpenStack and the Tesora DBaaS platform; and introduces OpenStack Live 2015, an annual conference that debuts April 13-14 2015 in Santa Clara, Calif.
Table of Contents

Chapter 1: DBaaS, OpenStack and Trove 101: Introduction to the basics ........................................ 3
Chapter 2: OpenStack Trove: The benefits of this DBaaS ................................................................. 7
Chapter 3: MySQL & OpenStack: How to overcome issues as your dataset grows ............................ 9
Chapter 4: Percona Server, OpenStack and the Tesora DBaaS Platform ........................................ 11
Chapter 5: OpenStack Live 2015: Annual conference debuts April 13-14, 2015 .............................. 12

About Percona

Percona was founded in August 2006 and now employs a global network of experts with a staff of more than 120 people. Our customer list is large and diverse, including Fortune 50 corporations, popular websites, and small startups. We have over 2,000 customers and, although we do not reveal all of their names, chances are we're working with nearly every large MySQL user you've heard about. To put Percona's MySQL expertise to work for you, please contact us.

Contact Us 24 Hours A Day

Is this an emergency? Get immediate assistance from Percona Support 24/7. Click here

Skype: oncall.percona
GTalk: oncall@percona.com
AIM (AOL Instant Messenger): oncallpercona
Telephone direct-to-engineer: +1-877-862-4316 or
UK Toll Free: +44-800-088-5561
Telephone to live operator: +1-888-488-8556
Customer portal: https://customers.percona.com/

Sales North (888) 316-9775 or
America (208) 473-2904
Sales +44-208-133-0309 (UK)
Europe 0-800-051-8984 (UK)
0-800-181-0665 (GER)
Training (855) 55TRAIN or
(925) 271-5054
We’ll be publishing a series of posts on OpenStack and Trove over the next few months on the MySQL Performance Blog, diving into usage and purpose. For readers who are already familiar with these technologies, there should be no doubt as to why we are incredibly excited about them, but for those who aren’t, consider this a small introduction to the basics and concepts.

What is Database as a Service (DBaaS)?
In a nutshell, DBaaS – as it is frequently referred to – is a loose moniker to the concept of providing a managed cloud-based database environment accessible by users, applications or developers. Its aim is to provide a full-fledged database environment, while minimizing the administrative turmoil and pains of managing the surrounding infrastructure.

Real life example: Imagine you are working on a new application that has to be accessible from multiple regions. Building and maintaining a large multiregion setup can be very expensive. Furthermore, it introduces additional complexity and strain on your system engineers once timezones start to come into play. The challenge of having to manage machines in multiple datacenters won’t simplify your release cycle, nor increase your engineers’ happiness.

Let’s take a look at some of the questions DBaaS could answer in a situation like this:

- **How do I need to size my machines, and where should I locate them?**
  Small environments require less computing power and can be a good starting point, although this also means they may not be as well-prepared for future growth. Buying larger-scale and more expensive hardware and hosting can be very expensive and can be a big stumbling block for a brand new development project. Hosting machines in multiple DC’s could also introduce administrative difficulties, like having different SLA’s and potential issues setting up WAN or VPN communications. DBaaS introduces an abstraction layer, so these consideration aren’t yours, but those of the company offering it, while you get to reap all the rewards.

- **Who will manage my environment from an operational standpoint?**
  Staffing considerations and taking on the required knowledge to properly maintain a production database are often either temporarily swept under the rug or, when the situation turns out badly, a cause for the untimely demise of quite a few young projects. Rather than think about how long ago you should have applied that security patch, wouldn’t it be nice to just focus on managing the data itself, and be otherwise confident that the layers beyond it are managed responsibly?

- **Have a sudden need to scale out?**
  Once you’re up and running, enjoying the success of a growing use base, your environment will need to scale accordingly. Rather than think long and hard on the many options available, as well
as the logistics attached to those changes, your DBaaS provider could handle this transparently.

**Popular public options:** Here are a few names of public services you may have come across already that fall under the DBaaS moniker:

– Amazon RDS  
– Rackspace cloud databases  
– Microsoft SQLAzure  
– Heroku  
– Clustrix DBaaS

What differentiates these services from a standard remote database is the abstraction layer that fully automates their backend, while still offering an environment that is familiar to what your development team is used to (be it MySQL, MongoDB, Microsoft SQLServer, or otherwise). A big tradeoff to using these services is that you are effectively trusting an external company with all of your data, which might make your legal team a bit nervous.

**Private cloud options?**  
What if you could offer your team the best of both worlds? Or even provide a similar type of service to your own customers? Over the years, a lot of platforms have been popping up to allow effective management and automation of virtual environments such as these, allowing you to effectively “roll your own” DBaaS. To get there, there are two important layers to consider:

- Infrastructure Management, also referred to as Infrastructure-as-a-Service (IaaS), focusing on the logistics of spinning up virtual machines and keeping their required software packages running.  
- Database Management, previously referred to DBaaS, transparently coordinating multiple database instances to work together and present themselves as a single, coherent data repository.

Examples of IaaS products:  
– OpenStack  
– OpenQRM

Example of DBaaS:  
– Trove

**Main Advantages of DBaaS**
For reference, the main reasons why you might want to consider using an existing DBaaS are as follows:

– *Reduced Database management costs*
DBaaS removes the amount of maintenance you need to perform on isolated DB instances. You offload the system administration of hardware, OS and database to either a dedicated service provider, or in the case where you are rolling your own, allow your database team to more efficiently manage and scale the platform (public vs private DBaaS).

- *Simplifies certain security aspects*

If you are opting to use a DBaaS platform, the responsibility of worrying about this or that patch being applied falls to your service provider, and you can generally assume that they’ll keep your platform secure from the software perspective.

- *Centralized management*

One system to rule them all. A guarantee of no nasty surprises concerning that one ancient server that should have been replaced years ago, but you never got around to it. As a user of DBaaS, all you need to worry about is how you interface with the database itself.

- *Easy provisioning*

Scaling of the environment happens transparently, with minimal additional management.

- *Choice of backends*

Typically, DBaaS providers offer you the choice of a multitude of database flavors, so you can mix and match according to your needs.

**Main Disadvantages**

- *Reduced visibility of the backend*

Releasing control of the backend requires a good amount of trust in your DBaaS provider. There is limited or no visibility into how backups are run and maintained, which configuration modifications are applied, or even when and which updates will be implemented. Just as you offload your responsibilities, you in turn need to rely on an SLA contract.

- *Potentially harder to recover from catastrophic failures*

Similarly to the above, unless your service providers have maintained thorough backups on your behalf, the lack of direct access to the host machines means that it could be much harder to recover from database failure.

- *Reduced performance for specific applications*
There’s a good chance that you are working on a shared environment. This means the amount of workload-specific performance tuning options is limited.

- Privacy and Security concerns

Although it is much easier to maintain and patch your environment. Having a centralized system also means you’re more prone to potential attacks targeting your dataset. Whichever provider you go with, make sure you are intimately aware of the measures they take to protect you from that, and what is expected from your side to help keep it safe.

Conclusion: While DBaaS is an interesting concept that introduces a completely new way of approaching an application’s database infrastructure, and can bring enterprises easily scalable, and financially flexible platforms, it should not be considered a silver bullet. Some big tradeoffs need to be considered carefully from the business perspective, and any move there should be accompanied with careful planning and investigation of options.

Embracing the immense flexibility these platforms offer, though, opens up a lot of interesting perspectives too. More and more companies are looking at ways to roll their own “as-a-Service”, provisioning completely automated hosted platforms for customers on-demand, and abstracting their management layers to allow them to be serviced by smaller, highly focused technical teams.
OpenStack's Trove: The benefits of this database as a service (DBaaS)

By Tim Sharp

In Chapter 1 my colleague Dimitri Vanoverbeke discussed at a high level the concepts of database as a service (DBaaS), OpenStack and OpenStack’s implementation of a DBaaS, Trove. Today I’d like to delve a bit further into Trove and discuss where it fits in, and who benefits.

Just to recap, Trove is OpenStack’s implementation of a database as a service for its cloud infrastructure as a service (IaaS). And as the declares, the Trove project seeks to provide a scalable and reliable cloud database service providing functionality for both relational and non-relational database engines. With the current release of Icehouse, the technology has begun to show maturity providing both stability and a rich feature set.

In my opinion, there are two primary markets that will benefit from Trove: the first being service providers such as RackSpace who provide cloud-based services similar to Amazon’s AWS. These are companies that wish to expand beyond the basic cloud services of storage and networking and provide their customer base with a richer cloud experience by providing higher level services such as DBaaS functionality. The other players are those companies that wish to “cloudify” their own internal systems. The reasons for this decision are varied, ranging from the desire to maintain complete control over all the architecture and the cloud components to legal constraints limiting the use of public cloud infrastructures.

With Trove, much of the management of your database system is taken care of by automating a significant portion of the configuration and initial setup steps necessitated when launching a new server. This includes deployment, configuration, patching, backups, restores, and monitoring that can be administered from either a CLI interface, RESTful API’s or OpenStack’s Horizon dashboard. At this point, what Trove doesn’t provide is failover, replication and clustering. This functionality is slated to be implemented in the Kilo release of OpenStack due out in April/2015.

The process flow is relatively simple. The OpenStack Administrator first configures the basic infrastructure by installing the database service. He or she would then create an image for each type of database they wish to support such as MySQL or MongoDB. They would then import the images and offer them to their tenants. From the end users perspective only a few commandes are necessary to get up and running. First issuing the <trove create> command to create a database service instance, followed by <trove list> command to get the ID of the instance and finally trove show command to get the IP address of it.

For example to create a database, you first start off by creating a database instance. This is an isolated database environment with compute and storage resources in a single tenant environment on a shared physical host machine. You can run a database instance with a variety of database engines such as MySQL or MongoDB.
From the Trove client I can issue the following command to create a database instance called PS_troveinstance, with a volume size of 2 GB, a user called PS_user, a password PS_password and the MySQL datastore (or database engine):

$ trove create --size 2 --users PS_user:PS_password --datastore MySQL PS_troveinstance

Next I issue the following command to get the ID of the database instance:

$ trove list PS_troveinstance

And finally, to create a database called PS_trovedb, I execute:

$ trove database-create PS_troveinstance PS_trovedb

Alternatively, I could have just combined the above commands as:

$ trove create --size 2 --database PS_trovedb users PS_user:PS_password --datastore MySQL PS_troveinstance

And thus we now have a MySQL database server containing a database called PS_trovedb.

In our next post on OpenStack/Trove, we’ll dig even further and discuss the software and hardware requirements, and how to actually set up Trove.

Many OpenStack operators use Percona open source software including the MySQL drop-in compatible Percona Server and Galera-based Percona XtraDB Cluster as well as tools such as Percona XtraBackup and Percona Toolkit.
MySQL & OpenStack: How to overcome issues as your dataset grows

By Stephane Combaudon

MySQL is the database of choice for most OpenStack components (Ceilometer is a notable exception). If you start with a small deployment, it will probably run like a charm. But as soon as the dataset grows, you will suddenly face several challenges.

Where is MySQL used in OpenStack?

Have a look at the logical diagram of OpenStack below (Click image for a larger view).

The diagram is a bit outdated: Neutron appears as Quantum and newer components like Heat are not pictured. But it shows that a database has to be used to store metadata or runtime information. And although many databases are supported, MySQL is the most common choice. Of course MySQL can also be used in instances running inside an OpenStack cloud.
What are the most common issues?

As with many applications, when you start small, the database is running well and maintenance operations are fast and easy to perform. But with a dataset that grows, you will find that the following operations are becoming increasingly difficult:

1. Having good backups: mysqldump is the standard backup tool for small deployments. While backups of instances having 100GB of data is still quite fast, restore is single-threaded and will take hours. You will probably need to use other tools such as Percona XtraBackup, but what are the tradeoffs?

2. Changing the schema: whenever you have to add an index, change a datatype or add a column, it can trigger a table rebuild which will prevent writes to proceed on the table. While the rebuild is fast when the table has only a few hundreds of MBs of data, ALTER TABLE statements can easily take hours or days for very large tables. Using pt-online-schema-change from Percona Toolkit is a good workaround, but it doesn’t mean that you can blindly run it without any precaution.

3. Making the database highly available: whenever the database is down, the whole platform is down or runs in a degraded state. So you need to plan for a high availability solution. One option is to use Galera, but that can introduce subtle issues.

4. Monitoring the health of your database instances: MySQL exposes hundreds of metrics, how do you know which ones to looked at to quickly identify potential issues?

1. and 2. are not likely to be issues for the MySQL instance backing your OpenStack cloud as it will be very small, but they can be big hurdles for guest databases that can grow very large depending on the application.

3. and 4. are highly desirable no matter the size of the database.

If MySQL is showing bad performance in your OpenStack deployment, Percona is here to help. Just give us a call anytime, 24/7. I also invite you and your team to attend the inaugural OpenStack Live 2015 conference, which runs April 13-14, 2015 in Santa Clara, Calif. It runs alongside the Percona Live MySQL Conference and Expo (April 13-16) at the Hyatt Regency Santa Clara and the Santa Clara Convention Center.
Percona Server and Percona XtraDB Cluster provide high-performance alternatives for MySQL users. We have also seen rapidly growing interest in these solutions in the OpenStack community where higher performance and high availability are crucial. Many OpenStack users are adopting these solutions but we’ve also seen demand from companies creating OpenStack distros. For example, Percona XtraDB Cluster is now certified for the RHEL OSP (OpenStack Platform) and is included in the Ubuntu 14.04 LTS release. Tesora recently joined this rapidly growing list when they announced the Tesora DBaaS Platform Enterprise Edition which includes Percona Server.

The Tesora platform is an enterprise-ready database as a service (DBaaS) platform based on the OpenStack Trove project. Percona Server was certified by Tesora in August as part of their Tesora DBaaS Platform certification program.

What does this mean for you if you are one of the users who have downloaded Percona Server more than 1,000,000 times?

Many enterprises are evaluating how to deliver robust, high-performance MySQL-as-a-service. OpenStack Trove is is an open-source platform that addresses this challenge. OpenStack operators can build and offer multiple databases as a service (such as MySQL, MongoDB, or Redis). These users can create, operate, maintain and delete different kinds of SQL or NoSQL databases on demand. Users don’t need to worry about the administrative details of the database technology nor complexities such as availability, resiliency, backups, recovery and security.

The Tesora DBaaS Platform Enterprise Edition builds upon OpenStack Trove and makes the “provisioning, deployment, configuration, tuning, monitoring and administration” simpler for operators. The platform includes support for multiple database backends including Percona Server which is certified on the Tesora DBaaS platform. Administrators and DBAs looking to build and offer a database as a service using the Tesora DBaaS Platform Enterprise Edition can rest assured that Percona Server is ready to meet their needs.

OpenStack is a rapidly evolving open-source platform that depends heavily on MySQL for optimal performance. Percona is participating in the Trove project and is sharing our knowledge on OpenStack through webinars (such as “OpenStack: A MySQL DBA Perspective”), blog posts (such as “OpenStack users shed light on Percona XtraDB Cluster deadlock issues”), professional services (such as Percona Consulting Services) and the recently announced OpenStack Live Conference which will be held April 13-14 2015 at the Santa Clara Convention Center in Silicon Valley. More on that in Chapter 5.

We look forward to working with Tesora as they build out their platform as well as the rest of the OpenStack community as this exciting technology continues to mature. I hope you can join us in Santa Clara for the OpenStack Live Conference this April – submit your speaker proposal now (deadline Nov. 17) or purchase your ticket at in advance at Early Bird prices.
OpenStack Live 2015: Annual conference debuts April 13-14, 2015

By Terry Erisman

OpenStack Live is a new annual conference that will run in parallel with the Percona Live MySQL Conference & Expo at the Santa Clara Convention Center in Silicon Valley. The inaugural event, OpenStack Live 2015, is April 13-14, 2015. We are lining up a strong Conference Committee and are now accepting tutorial and breakout session speaking proposals through November 9.

OpenStack Live will emphasize the essential elements of making OpenStack work better with emphasis on the critical role of MySQL and the value of Trove. You’ll hear about the hottest current topics, learn about operating a high-performing OpenStack deployment, and listen to top industry leaders describe the future of the OpenStack ecosystem. We are seeking speaking proposals on the following topics:

- Performance Optimization of OpenStack
- OpenStack Operations
- OpenStack Trove
- Replication and Backup for OpenStack
- High Availability for OpenStack
- OpenStack User Stories
- Monitoring and Tools for OpenStack

The conference features a full day of keynotes, breakout sessions, and Birds of a Feather sessions on April 14 preceded by an optional day of tutorials on April 13. A Monday reception will be held on the exhibit floor and joint lunches with both conferences offer you the opportunity to network with both the OpenStack and MySQL communities from both conferences. The OpenStack Live conference is a great event for users of any level.

As a bonus, OpenStack Live attendees may attend any Percona Live MySQL Conference session during the days of the OpenStack event. Conference only passes are available for April 14 and conference and tutorial passes are available for both April 13 and 14.

If you are using OpenStack and have a story to share – or a skill to teach – then now is the time to put pen to paper (or fingers to keyboard) and write your speaking proposal for either breakout or tutorial sessions (or both). Submissions will be reviewed by the OpenStack Live Conference Committee, which includes:

- Mark Atwood: Director – Open Source Evangelism for HP Cloud Services
- Rich Bowen: OpenStack Community Liaison at Red Hat
- Jason Rouault: Senior Director OpenStack Cloud at Time Warner Cable
- Peter Boros: Principal Architect at Percona
Presenting at OpenStack Live 2015 is your chance to put your ideas, case studies, best practices and technical knowledge in front of an intelligent, engaged audience of OpenStack users. If selected as a speaker by our Conference Committee, you will receive a complimentary full conference pass.

Public speaking not your thing or just want to learn about the latest and greatest OpenStack technologies, deployments and projects? Then [register now](http://example.com) and save big with our early bird discount. OpenStack Live 2015 is an ideal opportunity for organizations to connect with the community of OpenStack enthusiasts from Silicon Valley and around the world. The Percona Live MySQL Conference this past April had over 1,100 registered attendees from 40 countries and the OpenStack Open Source Appreciation Day on the Monday before the conference was fully booked so don’t delay, register today to save your seat!

We are currently accepting sponsors. You can learn more about sponsorship opportunities [here](http://example.com).

I hope to see you at OpenStack Live 2015 next April! And speakers, remember the deadline to submit your proposals is November 17. In the meantime you can learn more by visiting the official [OpenStack Live 2015 website](http://example.com).
About the authors

Field application engineer Dimitri Vanoverbeke is an open source veteran consultant, familiar with a broad range of open source solutions and adheres to the devops philosophy. He also enjoys traveling, cultural activities, basketball and the great outdoors. Dimitri lives with his girlfriend in the beautiful city of Ghent, Belgium.

Field application engineer Tim Sharp joined Percona in the summer of 2013 as a MySQL Technical Account Manager. His areas of knowledge include embedded database technologies, customer service and the traditional Linux/Apache/MySQL/PHP stack. Tim lives on Vancouver Island with his partner Aeron, a Bernese Mountain Dog named Maggy and a kitten called Peaches.

Percona architect Stéphane Combaudon joined Percona in July 2012 after working as a MySQL DBA for leading French companies. In real life, he lives in Paris with his wife and their twin daughters. When not in front of a computer or not spending time with his family, he likes playing chess and hiking.

Tushar Katarki is a director of product management at Percona. He has 15+ years of experience as a software developer, software architect, project manager and product manager. Prior to Percona he help positions at Red Hat, Polycom, Oracle, Sun and Sycamore Networks. Tushar is married and lives with his wife and two children in the Boston area.

Percona CMO Terry Erisman is an industry veteran with more than 20 years of experience that includes helping start seven companies, launch more than 25 new products, and driving high revenue for a multitude of award-winning companies in the SaaS, open source, and enterprise software sectors. He lives in the heart of Silicon Valley.

Visit Percona’s ever-growing MySQL eBook library for more great content at http://www.percona.com/resources/mysql-ebooks