

## Tobias Lütke's thoughts /

# Easy migration between Databases

Posted by **tobi** — 06:01 AM *May 29*

Recently we did a major switch in **Database** architectures here at Jaded Pixel and needed a simple way to move from one architecture to another.

The first thing I tried was to try to get the dump utility of the **database** we were using to produce sensible and portable SQL inserts. This effort fell flat on the face because of subtle differences in the database's string escaping and handling of booleans.

So after a quick inquiry in the Rails core channel [Rick Olson](#) recommended dumping the data to YAML and reloading it on the other side. Simple enough, the total amount of data to transfer was well below 100mb so this seemed like a sensible approach.

Get the [backup.rake](#) and add it to your lib/tasks/ directory.

Here is the basic process:

1. Connect to your server and use `RAILS_ENV=production rake db:backup:write` to get a yaml representation of all your data

```
rake db:backup:write
(in /Users/tobi/Code/Ruby/shopify)
Writing addresses...
Writing articles...
Writing blogs...
Writing carts...
Writing collections...
...
```

2. Update your `datababe.yml` to point to your new **database**.
3. Run `RAILS_ENV=production rake db:backup:read` to fill your new **database** with all written data.

Careful `db:backup:read` will **delete all data** in the target **database**. Use only with extreme caution.

## Comments

- [Sebastian Gräßl](#) 29 May 12:32

very cool and usefull. it is also usefull to just backup data. maybe as a cron.

i guess it would be also cool to have such a backup feature in shopify and not only to backup the data also to plug it right into vision to work with the target product group

- rick 29 May 12:39

Sebastian: I think using your database's own backup facilities would be best if that's all you need. I imagine it'd be much faster and more reliable for backup/restore.

- [Sebastian Gräßl](#) 29 May 12:46

@rick: yes, for backup the db it would be better to use the db's facility, but for the Vision db's it usefull

- [tobias Luetke](#) 29 May 12:46

Indeed. The YAML feature is very slow. I'm sure you could speed it up by some orders of magnitude by using `connection.select_values` instead of using the models directly; hey—It works and did its job for us!

- [Sebastian Gräßl](#) 29 May 14:05

to which db did you switch?

- Dennis Theisen 29 May 14:46

Hi Tobi! Long time no see. I could't find your email address anywhere, so my only option was to post here... Willste mich vielleicht ma kurz anmailen?

- [tobias Luetke](#) 29 May 15:07

Postgresql -> Mysql. A rather unpopular choice it might seem but there were excellent reasons..

- [Patrick Lenz](#) 29 May 15:27

Shouldn't one of the tasks above be `db:backup:read`?

- [Tobias Luetke](#) 29 May 17:00

updated that.

- Pat 29 May 18:24

Hey Tobi, any chance you will be able to post about why you switched? I've found that for most people, it doesn't matter which db they use...but you said you had excellent reasons, so I'd be interested in knowing what they are.

- [tobi](#) 29 May 20:21

Well i shouldn't have said excellent. It was a good enough reason for us. The magic word is replication. In the postgres world this is done externally as postgres itself has no replication abilities.

No matter how hard we tried we could not get a stable replication setup build. The main issue we kept running in was that no replication solution for postgres was able to replicate schema changes. This means that deploying migrations was complicated by orders of magnitude because now we had to distinguish between code migrations and schema migrations. Schema migrations had to run on all boxes at the same time, couldn't be undone and all databases had to be locked for this to happen. Code migrations could only run on the master machine.

Data integrity is what we need most from our **database** so this was a major issue. In shopify we have some analytic batch jobs which take a very long time to run and its imperative that these jobs need replicated databases of their own. So we made the decision of going back to mysql where we just had to set 3 config variables and every statement was replicated on all slaves

which cared for it. Pragmatism to the extreme.

- [Pingu](#) 30 May 05:51

Would you guys be interested in Dump slurp YAML plugin?

<http://nubyonrails.com/articles/2005/12/27/dump-or-slurp-yaml-reference-data>

- Brittain 30 May 15:07

Step 3 still references db:backup:write instead of db:backup:read

- [tobi](#) 30 May 15:29

Thanks, fixed this.

- Thorsten 02 Jun 00:21

Would you care commenting on what **database migration** you had to do and why?

- [Peter Morris](#) 21 Jun 05:22

Hi,

can you tell me which versions of rails your task works with?

I tried it on a project using 1.0 and received an error regarding symbol / proc mismatch...

Cheers, Peter.

**Commenting are now closed...**

## About

**Tobias Lütke** is cofounder and CEO of [Shopify](#), alumni of the Ruby on Rails core team and a startup advisor. Tobi wrote popular open source software such as ActiveMerchant, Liquid, Delayed::Job, Typo etc.

## Popular posts

## Categories

- [Home](#)
- [Code](#)
- [Rails](#)
- [Gallery](#)
- [about](#)
- [Testing](#)
- [test](#)
- [Talks](#)
- [Comedy](#)
- [Excellence in Advertisement](#)
- [Search](#)

- [Archives](#)

## Projects and Libraries

- [Shopify - Online store](#)
- [Liquid](#)
- [Active Merchant](#)

## Friends

- [Ottawa Startup](#)
- [Daniel Weinand](#)
- [Cody Fauser](#)
- [Andrew Geddes](#)

1449 readers  
BY FEEDBURNER

## Syndicate

-  [Articles](#)

All rights reserved.