

*Moving from 0.4-alpha8 to 0.4-beta2*

While we tried to use some documentation that was supposed to work, it failed. The failure gave little real indication as to what the problem was. The documentation selected that failed was <http://community.aegirproject.org/node/117>

From the IRC channel, we talked to mig5 and he suggested that we do it manually.

The manual process worked and actually, pretty well. I am not sure of the long term for this as the database will have DBnames that are created manually, instead of the 'site\_[nodeID]' naming convention. Beyond that, I have not run into any problems yet.

There are 2 different elements that need to be migrated. Platforms, and Sites.

## Platform migration

### Rsync Platform

Each and any platforms on the old Aegir box need to be copied to the new Aegir box.

Suggested commands on the new server are:

```
#> cd /var/aegir/platforms
```

```
#> rsync -aPv -e ssh aegir@old-server.com:/var/aegir/platforms/[platform name] .
```

### Clean the destination Platform

Since you copied the entire platform, it has the codebase for all the sites. We want to clean that out and process them individually... Working in the platform folder:

```
#> for d in `ls | grep -v "all\|default"`; do rm -fR $d; done
```

### Inform Aegir of new Platform

Login to the UI for the new Aegir

Add the new platform

As part of this process, the verify will be run.

## Site migration

Select the site(s) that you want to migrate.

### Create Backup

On the Old Aegir box, enter the Browser UI and find the Site. Select the 'backup' button and wait for the backup to complete.

### Transfer Backup

On the new Aegir box, if you don't already have it, create a tmp/migrate folder to hold the backups from the Old Aegir box. Then rsync to get all the latest backups.

```
#> cd /var/aegir/tmp/migrate
```

```
#> rsync -aPv -e ssh aegir@old-server.com:/var/aegir/backups/* .
```

### Create site

In the [platform]/sites folder you will create a folder for the site. Obviously, it needs to be the same name as it was on the old Aegir box.

```
#> su aegir
```

```
#> cd /var/aegir/[platform]
```

```
#> mkdir [site]
```

```
#> cd [site]
```

```
#> tar -xvzpf /var/aegir/tmp/migrate/[backupfile]
```

### Verify Rights

The owner and group rights should have come across properly... but check anyway.

Especially, the 'files' folder needs to have the group set to the group that apache is running as.

```
#> find files -type d -exec chgrp www-data {} \;
```

### Setup the Database

The database was backed up on the old site along with the codebase and files. It is named database.sql and is in the top of the folder structure. Assuming you know how to pass the -u [user] -p[password] parameters to mysql....

```
#> mysql -e "create database [dbname]"
```

```
#> cat database.sql | mysql [dbname]
```

### Modify settings.php

The settings.php file points to database information that is passed to it by the vhosts.d entries.

This does not work at this point, so must be commented out and replaced with a working \$db\_url variable. So, edit the settings.php file and comment all the references to \$\_SERVER and \$db\_url. Now create a \$db\_url similar to:

```
$db_url = "mysqli://user:password@localhost/dbname";
```

### Inform Aegir of new site

Informing Aegir of the new site is a simple process. Using the Browser UI, go to the platform that you just installed this into and hit the verify button. It will find the new site, and verify the site. When it is done, select the site and edit it. At this point, you can re-establish any aliases.

## Various notes

- ☞ Aliasing is not on by default. When you setup the new server, you need to run a one-time process to enable aliases and anything you might want that is not on by default. Found in admin/hosting/features.