

## Upgrade from WebGUI 6.2.11 to 7.3.33

tar xvzf 6.8a4des0 [0,0]

Contents: [Dobrica PavlinuÅ¸iÄ¸ 's random unstructured stuff]

- [Dobrica PavlinuÅ¸iÄ¸ 's random unstructured stuff \(Install dependencies\)](#)
- [Dobrica PavlinuÅ¸iÄ¸ 's random unstructured stuff \(Basic configuration\)](#)
  - ◆ [Dobrica PavlinuÅ¸iÄ¸ 's random unstructured stuff \(Mysql binary log\)](#)
- [Dobrica PavlinuÅ¸iÄ¸ 's random unstructured stuff \(Cleanup target machine\)](#)
- [Dobrica PavlinuÅ¸iÄ¸ 's random unstructured stuff \(Copy existing installation\)](#)
  - ◆ [Dobrica PavlinuÅ¸iÄ¸ 's random unstructured stuff \(Copy MySQL databases\)](#)
  - ◆ [Dobrica PavlinuÅ¸iÄ¸ 's random unstructured stuff \(Copy existing WebGUI installation\)](#)
  - ◆ [Dobrica PavlinuÅ¸iÄ¸ 's random unstructured stuff \(Finish database setup\)](#)
- [Dobrica PavlinuÅ¸iÄ¸ 's random unstructured stuff \(Upgrade steps\)](#)
  - ◆ [Dobrica PavlinuÅ¸iÄ¸ 's random unstructured stuff \(Upgrade scripts\)](#)
  - ◆ [Dobrica PavlinuÅ¸iÄ¸ 's random unstructured stuff \(Helper scripts\)](#)
- [Dobrica PavlinuÅ¸iÄ¸ 's random unstructured stuff \(Tweaks\)](#)
  - ◆ [Dobrica PavlinuÅ¸iÄ¸ 's random unstructured stuff \(MySQL encoding\)](#)
  - ◆ [Dobrica PavlinuÅ¸iÄ¸ 's random unstructured stuff \(Fix configuration\)](#)
  - ◆ [Dobrica PavlinuÅ¸iÄ¸ 's random unstructured stuff \(Code cleanup\)](#)
  - ◆ [Dobrica PavlinuÅ¸iÄ¸ 's random unstructured stuff \(Fix permissions\)](#)
  - ◆ [Dobrica PavlinuÅ¸iÄ¸ 's random unstructured stuff \(Spectre\)](#)
    - ◇ [Dobrica PavlinuÅ¸iÄ¸ 's random unstructured stuff \(Cleanup versions\)](#)
    - ◇ [Dobrica PavlinuÅ¸iÄ¸ 's random unstructured stuff \(Delete workflow instances\)](#)
    - ◇ [Dobrica PavlinuÅ¸iÄ¸ 's random unstructured stuff \(/etc/resolv.conf\)](#)
- [Dobrica PavlinuÅ¸iÄ¸ 's random unstructured stuff \(Critical version you have to upgrade to\)](#)
  - ◆ [Dobrica PavlinuÅ¸iÄ¸ 's random unstructured stuff \(6.6.4\)](#)
  - ◆ [Dobrica PavlinuÅ¸iÄ¸ 's random unstructured stuff \(6.8.10\)](#)
  - ◆ [Dobrica PavlinuÅ¸iÄ¸ 's random unstructured stuff \(6.99.5\)](#)
  - ◆ [Dobrica PavlinuÅ¸iÄ¸ 's random unstructured stuff \(7.3.22\)](#)
- [Dobrica PavlinuÅ¸iÄ¸ 's random unstructured stuff \(Version related notes\)](#)
  - ◆ [Dobrica PavlinuÅ¸iÄ¸ 's random unstructured stuff \(7.3.4\)](#)
  - ◆ [Dobrica PavlinuÅ¸iÄ¸ 's random unstructured stuff \(7.4.17\)](#)
  - ◆ [Dobrica PavlinuÅ¸iÄ¸ 's random unstructured stuff \(7.4.33\)](#)
- [Dobrica PavlinuÅ¸iÄ¸ 's random unstructured stuff \(Hint\)](#)

## Install dependencies

```
apt-get install apache2-mpm-prefork
apt-get install libapache2-mod-perl2
apt-get install perlmagick imagemagick
```

Don't bother with installation of perl modules from packages. Debian has modules which are sometimes a bit too old for cutting-edge WebGUI, so just let CPAN install it for you.

# Basic configuration

All operations on both hosts (mjesecc, zemlja) are done as `root` user. This is not strictly necessary (especially for WebGUI tasks), but it's easier in combination with MySQL setup. We want to get running quickly, YMMV.

## Mysql binary log

Comment out following in `/etc/mysql/my.cnf`

```
log_bin
expire_logs_days
max_binlog_size
```

To preserve disk space if you are not using replication!

And restart server:

```
/etc/init.d/mysql restart
rm /var/log/mysql/mysql-bin.*
```

## Cleanup target machine

If this is not first installation attempt, you might want to cleanup target machine first.

```
# root@mjesec
cd /data/WebGUI
grep dsn etc/*.conf | cut -d: -f4- \
  | xargs -i echo 'drop database {};' \
  | ssh zemlja 'cat > /data/drop_all_dbs.sql'
```

Now switch to target machine and finish cleanup:

```
# root@zemlja
cd /data
xargs -i mysql -e '{} ' < /data/drop_all_dbs.sql
rm -Rf WebGUI
```

## Copy existing installation

In this step, we will copy existing WebGUI 6.2.11 installation to new (virtual) host (zemlja) to test upgrade.

## Copy MySQL databases

```
# root@mjesec
cd /data/WebGUI
grep dsn etc/*.conf | cut -d: -f4- \
  | xargs mysqldump --databases \
  | ssh zemlja mysql mysql
```

## Copy existing WebGUI installation

```
# root@mjesec
cd /data
tar cfp - WebGUI/ | ssh zemlja 'cd /data && tar xvpf -'
```

This is much faster than scp in my experience...

## Finish database setup

Now, on zemlja, let's grant webgui user access to those databases:

```
# root@zemlja
cd /data/WebGUI
mysql -e 'create user webgui'
grep dsn etc/*.conf | cut -d: -f4- \
  | xargs -i mysql -e "grant all privileges on {}. * to webgui@localhost identified by 'password';"
mysql -e "flush privileges"
```

## Upgrade steps

First, test current environment for any mis-configuration:

```
# root@zemlja
cd /data/WebGUI/sbin
perl testEnvironment.pl
```

If there are missing perl libraries, install them. Some will have to be forced, but all-in-all it goes well.

And run upgrade...

```
# root@zemlja
cd /data/WebGUI/sbin
perl upgrade.pl --doit
```

Now, extract new version of webgui on which you want to upgrade (I'm assume that you downloaded required versions in /data):

```
cd /data/  
ls  
tar xvfz webgui-6.6.4-beta.tar.gz  
cd /data/WebGUI/sbin/  
perl testEnvironment.pl  
perl upgrade.pl --doit
```

This will take a while.

After that you might want to remove backup of databases:

```
rm /tmp/backups/*
```

## Upgrade scripts

Here are list of upgrade scripts:

- [z-1-backpan.sh](#)
- [z-3-mysql-perms.sh](#)
- [z-4-webgui-6.6.5.sh](#)
- [z-5-webgui-6.8.10.sh](#)
- [z-6-webgui-7.0.9.sh](#)
  - ◆ [upgrade\\_6.8.9-6.8.10.pl](#)
  - ◆ [webgui-7.0.diff](#)
- [z-7-webgui-7.2.3.sh](#)
- [z-8-webgui-7.3.22.sh](#)
- [z-9-webgui-7.4.38.sh](#)

## Helper scripts

Backup/restore whole WebGUI instance (data+database) so that you can re-try upgrade

This was probably one of most useful ideas enabling me to re-try upgrade until it finish without errors!

- [webgui-snap.sh](#)
- [webgui-restore.sh](#)
- [webgui-recover-db.sh](#)

## Tweaks

Here are some commands to help you with various changes which break WebGUI:

```
# root@zemplja  
cd /data/WebGUI  
rm lib/HTML/Template.pm  
  
cd /data/WebGUI/etc/  
mv log.conf.original log.conf
```

```
mv spectre.conf.original spectre.conf
```

## MySQL encoding

My data has `iso-8859-2` while MySQL thinks it's in `latin-1`.

```
mysqldump --all-databases --add-drop-database --default-character-set=latin1 > all.sql
cat all.sql | sed \
-e 's/DEFAULT CHARSET=latin1/DEFAULT CHARSET=utf8/g' \
-e 's/character set latin1 collate latin1_bin/character set utf8 collate utf8_bin/' \
-e 's/DEFAULT CHARACTER SET latin1/DEFAULT CHARACTER SET utf8/' \
-e 's/SET NAMES latin1/SET NAMES utf8/' \
| iconv -f iso-8859-2 -t utf-8 \
| mysql --default-character-set=utf8 mysql
```

### **This should work, but it doesn't**

So, we are using much simpler solution which keep encoding in database in wrong latin1 encoding, but it displays correctly in WebGUI:

```
cat all.sql | iconv -f iso-8859-2 -t utf-8 | mysql mysql
```

## Fix configuration

Upgrade procedure isn't still perfect. Here is a list of tweaks to fix configuraion:

```
cd /data/WebGUI/etc/
perl -p -i -n -e 's/WebGUI::Asset::Wobject::WebGUI::Asset::Wobject::Survey/WebGUI::Asset::Wobject
```

## Code cleanup

You might want to replace `WebGUI/lib/WebGUI` which is result of upgrade with clean version from distribution. I have accumulated some cruft there, and it was spitting errors in `/var/log/apache2/error.log` and `/var/log/webgui.log`

## Fix permissions

```
cd /data/WebGUI
sudo chown -R www-data */uploads
```

# Spectre

I didn't have `spectre.conf` after upgrade. Test connectivity with:

```
cd /data/WebGUI/sbin/  
perl spectre.pl --test
```

Add following to `/etc/rc.local`

```
cd /data/WebGUI/sbin/  
perl spectre.pl --daemon
```

## Cleanup versions

Since I created few workflows which always returned commerr (because of mis-configuration) I needed to find way to manually erase them.

```
select tagId from assetVersionTag  
join WorkflowInstance on WorkflowInstance.instanceId = assetVersionTag.workflowInstanceId ;
```

## Delete workflow instances

```
delete from WorkflowInstance ;
```

This will erase **all** pending workflow instances. So, be sure to check if they are all stuck before running this command!

## **/etc/resolv.conf**

Make sure your `nameserver` entries point to existing servers. If you have one of DNS servers which doesn't respond (I had 127.0.0.1 for DNS and there wasn't any DNS server on machine running WebGUI) you will get Errors 500 from spectre without any indication where the problem is.

# Critical verision you have to upgrade to

**This part is obsoleted by scripts above which provide correct and workable upgrade path**

## 6.6.4

last 6.6.x version required for upgrade to 6.7.0

## 6.8.10

Upgrade to this version will die at one point with:

Cannot open config file '../etc/log.conf' at /usr/share/perl5/Log/Log4perl/Config.pm line 612.

```
# root@zemlja
cd /data/WebGUI/etc/
cp log.conf.original log.conf
cp spectre.conf.original spectre.conf
```

And then, re-run upgrade procedure:

```
cd /data/WebGUI/sbin/
perl upgrade --doit
```

Unit it succeeds.

## 6.99.5

this version will convert config files to JSON format

First, install **old** JSON or you will be in pain:

<http://backpan.perl.org/authors/id/M/MA/MAKAMAKA/JSON-0.991.tar.gz>

You might have problems upgrading to version 6.99.0 or later because scripts expect configuration in JSON format, and upgrade won't convert files for you (because it created config files from tar).

So, change your `upgrade.pl` beginning to something like this:

```
## Find site configs.

print "\nTesting site config versions...\n" unless ($quiet);
opendir(DIR, "../etc");
my @files = readdir(DIR);
closedir(DIR);
foreach my $file (@files) {
    next unless ($file =~ m/\.conf$/);
    # next if ($file eq "spectre.conf" || $file eq "log.conf");
    my $configFile = "../etc/".$file;
    open(FILE, "<".$configFile);
    my $line = <FILE>;
    close(FILE);
    unless ($line =~ m/JSON 1/) {
        print "\tConverting ".$file." from PlainConfig to JSON\n" unless ($quiet);
        convertPlainconfigToJson($configFile);
    }
}

exit;
```

Note commented out line and added exit.

Now you can run `upgrade.pl` once to convert configuration files to JSON, then run `testEnvironment.pl` and finally run `upgrade.pl` once more to do actual update.

## 7.3.22

requirement for upgrade to 7.4.0

To prevent `perl upgrade.pl --doit` die with:

```
Cannot open config file '../etc/log.conf' at /usr/share/perl5/Log/Log4perl/Config.pm line 612.
```

fix it:

```
cd /data/WebGUI/etc/  
mv log.conf.original log.conf
```

and **then** run upgrade!

## Version related notes

This are just notes. You can ignore them for normal upgrade described above.

### 7.3.4

Re-run upgrade procedure if it breaks

### 7.4.17

Attack of JSON again:

```
Can't locate object method "relaxed" via package "JSON"
```

Install <http://backpan.perl.org/authors/id/R/RI/RIZEN/Config-JSON-1.1.0.tar.gz>

### 7.4.33

It will install newer JSON and Config::JSON packages from CPAN

## Hint

For testing, I created `*.zemplja.example.com` DNS entry, so that all sites become `old.site.name.zemplja.example.com`.