



CubiCommandLine

describe cubi command line tools

Phase-Implementation

Updated Jun 30, 2011 by [rockys...@gmail.com](#)

Cubi Command Line Tools

Cubi provides a set of command line tools or developers to manage Cubi application more efficiently. The scripts can be found under `/cubi/bin/tools/`

Module loader

This script is to load module into Cubi. Once you have a new module ready (mod.xml and other assets are completed), you need to load the module in the application.

The load module script does

- Check the module dependency defined in `<Dependency>`. If a module that this module depending on is not loaded, a warning will be given.
- Execute the `mod.install.sql` the first time of loading the module.
- Load the module attributes defined in `<Module>`.
- Load module's resource and actions defined in `<ACL>`.
- Load module's menu items defined in `<Menu>`.
- Load metadata objects (dataobj, form, view).
- Give administrator all permissions to the module.

How to use:

```
# php load_module.php module_name [-i]
```

'-i' indicates if install module sql which is usually for fresh new installation

Metadata Generator

Cubi Metadata Generator allows developers to create a new module quickly.

How to use the script

```
# php gen_meta.php dbname table
```

For example, say you have a new table called "abc". First you create abc table in Cubi database. Then enter `cubi/bin/tools/`

```
# php gen_meta.php Default abc
```

After the script is executed, the following files are generated.

- `/modules/abc/mod.xml`
- `/modules/abc/do!AbcDO.xml`
- `/modules/abc/form/AbcListForm.xml`
- `/modules/abc/form/AbcDetailForm.xml`
- `/modules/abc/form/AbcEditForm.xml`
- `/modules/abc/form/AbcNewForm.xml`
- `/modules/abc/form/AbcCopyForm.xml`
- `/modules/abc/view/AbcListView.xml`

Cubi theme generator

This tool allows user to generate a new theme based on the default Cubi theme.

How to use the script

```
# php gen_theme.php new_theme_name
```

After the script is executed, the following files are generated. `cubi/theme/new_theme/...`

Cubi module upgrader

This tool allows user to upgrade module to higher version.

Prepare the upgrade

In order to upgrade a module, you just need to copy a newer version of module folder to cubi/upgrade/module/mod_name/.

In mod.xml, make sure the correct version is set to higher number than the current version.

Create or modify cubi/upgrade/module/mod_name/upgrade.xml. Add proper change for in <Upgradesql> for the target version. For example,

```
<?xml version="1.0" standalone="no"?>
<Upgrade>
  <Version Name="0.1">
  </Version>
  <Version Name="0.1.1">
    <Upgradesql><![CDATA[
      ALTER TABLE `help` ADD `add1` varchar(255) default NULL AFTER `content`;
      ALTER TABLE `help` ADD `add2` int(10) default NULL AFTER `add1`;
    ]]></Upgradesql>
  </Version>
  <Version Name="0.1.2">
    <Upgradesql><![CDATA[
      ALTER TABLE `help` ADD `add3` varchar(255) default NULL AFTER `add2`;
      ALTER TABLE `help` ADD `add4` int(10) default NULL AFTER `add3`;
    ]]></Upgradesql>
  </Version>
</Upgrade>
```

How to use the script

```
# php upgrade.php module_name
```

What does the script do

1. Verify the module version in upgrade folder is higher than current module version
2. Back up the current module folder to cubi/backup/modules/mod_name
3. Copy the files in the upgrade folder to current module folder
4. Execute the upgrade SQL describe in module's upgrade.xml
5. Reload the module

Example of upgrade help 0.1 to 0.1.2

```
# php upgrade.php help
Start upgrading help module ...
-----
Upgrade 'help' module from version 0.1 to 0.1.2. Please backup data first.
Press enter to continue ...
Backup source files to C:\xampp\htdocs\ob24\cubi\backup\modules/help/0.1 ...
Copy source files from C:\xampp\htdocs\ob24\cubi\upgrade\modules/help to C:\xampp\htdocs\ob24\cubi\modules/help...
Execute upgrade sql files ...
Upgrade from version 0.1 to 0.1.1 ...
Execute ALTER TABLE `help` ADD `add1` varchar(255) default NULL AFTER `content`
Execute ALTER TABLE `help` ADD `add2` int(10) default NULL AFTER `add1`
Upgrade from version 0.1 to 0.1.2 ...
Execute ALTER TABLE `help` ADD `add3` varchar(255) default NULL AFTER `add2`
Execute ALTER TABLE `help` ADD `add4` int(10) default NULL AFTER `add3`
Reload module ...
[2011-01-11T00:15:53-08:00] Install Module.
[2011-01-11T00:15:53-08:00] Install Module ACL.
[2011-01-11T00:15:53-08:00] Install Module Menu.
[2011-01-11T00:15:53-08:00] help is loaded.
Give admin to access all actions of module 'help'
-----
End loading help module
```

► [Sign in](#) to add a comment

[Terms](#) - [Privacy](#) - [Project Hosting Help](#)

Powered by [Google Project Hosting](#)