mpkg Documentation

Release 0.0.3

theunknownxy

November 30, 2014

Contents

Contents:

CHAPTER 1

Tutorial

This tutorial gives a introduction to mpkg. Please note that this tutorial describes the text console version of mpkg(At the moment there isn't a GUI version).

1.1 Setup

The first step is to download mpkg from Bitbucket. Extract it somewhere and launch *mpkg.exe init* (respectively *mono mpkg.exe init* on Linux/Mac) within a terminal. This creates some files filled with the default settings. After that open *mpkg.conf* and verify the settings. Note down the value of *mpkg-dir*. This is where all mpkg files resides.

1.2 Repositories

After you installed mpkg, you probably want to add some repositories(They contain package files, that you can install). The official repositories are:

- theunknownxy: https://bitbucket.org/theunknownxy/theunknown-mpkg/get/tip.zip

To add one, navigate to the *mpkg-dir* and open *repos.conf*. The file has the following format:

```
name=url
...
```

When you are done, save the file and go again to the terminal. Now you have to download or update the repositories which are in the *repos.conf* file. So run mpkg, but this time replace *init* with *sync* (e.g. *mpkg.exe sync*). After it's done you can continue with searching and installing packages.

1.3 Packages

To search a package just run *mpkg.exe search <QUERY>*, where you replace *<QUERY>* with your search string. If mpkg has found something, a list of packages should appear:

```
$ mpkg.exe search ModLoader
- repo1/ModLoader
  A mod, kind ...
- ...
```

The *theunknown/* part says in which repository the package was found. Then the package name follows. And below that, there is a short description of the package. So let's install ModLoader(mostly you don't need to do it manually. If a package requires it, mpkg automatically installs it and all other dependencies.) Add it with *mpkg.exe add ModLoader*. But it's not yet installed. It's just added to a list(You can view this list using *mpkg.exe list*). So now you need to run *mpkg.exe build*, which downloads and puts the mods in the right place. Then you can run Minecraft with the mods installed. If you want to get rid of a package you don't want, remove it using *mpkg.exe remove PACKAGE* and rebuild Minecraft.

1.4 Profiles

TODO

CHAPTER 2

Package description files

The package description files are written in XML. Thus you can edit them with any text editor. But an editor with syntax highlighting is recommended.

The root element of every description is *mod*:

```
<mod>
    ...
</mod>
```

In the root element can contain the following elements:

- name – Name of the package(required):

  ```
  <name>MyMod</name>
  ```

- author – Name of the author(required):

  ```
  <author>Nobody</author>
  ```

- version – Version of the package(required):

  ```
  <version>1.2.0</version>
  ```

- revision – Revision of the description file(increment after edit by one):

  ```
  <revision>3</revision>
  ```

- mcversion – Version of Minecraft, for which the package is made for(required):

  ```
  <mcversion>1.1.0</mcversion>
  ```

- shortdesc – Short description of the package(optional):

  ```
  <shortdesc>This is doing something cool</shortdesc>
  ```

- description – Long description of the package(optional):

  ```
  <description>This is doing this and that. It can also do ...</description>
  ```

- **dependencies – List of dependencies. Can contain zero or more *dependency* elements(required)**
  - dependency – Name of a dependency(optional):

    ```
    <dependencies>
        <dependency>...</dependency>
    </dependencies>
    ```

- **downloads – List of downloads(required)**

  - download – URL of the download. Specify target name with attribute *name* and in case the user has to download the file, use the *user* attribute:

    ```
    <downloads>
        <download name="mod.zip" user="true">http://adf.ly/...</download>
    </downloads>
    ```

- **install – List of instructions to install the package(required)**

  - descr – Description of one step. Set the type using the *type* attribute(The types are listed at *Install Descriptions*):

    ```
    <install>
        <descr type="...">
            ...
        </descr>
    </install>
    ```

2.1 Install Descriptions

Currently there are the following types:

- **source2jar – Copies files from a source to the minecraft.jar. Fields:**

  - source – The source
  - from – File/Folder to copy. Leave empty to copy everything in the source.
  - to – Where to place them. Leave it empty to copy it to the root directory.

  ```
  <descr type="source2jar">
      <source>somesource</source>
      <from></from>
      <to></to>
  </descr>
  ```

- **zip2source – Creates a source from a zip files contained in another source**

  - source – The source
  - from – The zip file
  - target – The name of the source to create

  ```
  <descr type="zip2source">
      <source>somes</source>
      <from>folder/Core.zip</from>
      <target>core</target>
  </descr>
  ```

- **download2source – Creates an source from an downloaded file**

  - download – The filename of the download
  - target – The name of the source to create

```
<descr type="download2source">
    <download>mymod.zip</download>
    <target>MyMod</target>
</descr>
```

CHAPTER 3

Indices and tables

- *genindex*
- *modindex*
- *search*