ATLauncher

XML and API Documentation

Version 1.0.22
Last Updated: 18/10/2013

Table of Contents

XML Introduction

This specification document outlines the correct way that Pack Specifications should be laid out and formatted for use in the ATLauncher.

All Pack Specifications must follow valid XML 1.0 specifications (http://www.w3.org/TR/xml/) and must also follow this specification before it's accepted as a valid and available Pack Specification for use in the ATLauncher.

Please note that this specification can change at anytime and you should keep track of updates to this spec by following our Twitter account (@ATLauncher).

If you have any questions about this specification such as errors you may find or if you need help formatting your XML Pack Specifications then please feel free to post a thread in the forums (http://www.atlauncher.com/forums/) or email us (contact@atlauncher.com).

Basic Layout

Each version of your Pack Specification should include the following 4 sections, with the libraries and colours section being optional, as outlined below. All sections should be wrapped in <version></version> tags in order to specify that it is a version.

<pack>

This section contains information about your pack including version number and minecraft version number.

<libraries>

This section contains information about the libraries that your pack uses. These libraries are additional jars required to run your pack such as for Minecraft Forge. You don’t need to include the Vanilla Minecraft libraries as they are already downloaded by default.

<colours>

This section contains information about the different colours used to identify mods and is an optional section. You can colour mods that show in the mod selection screen to make it easier to identify grouped mods for example, but the use is completely up to you.

<mods>

This section contains information about the mods included in your pack including things like name, version, url and others.

<pack>

This section of the Pack Specification contains several different attributes as listed below.

<version>

Version should contain the version number of your pack. This should be different from any previous version of your pack. It may only contain numbers, letters, dashes, underscores and periods (a-zA-Z0-9-_.)

<minecraft>

Minecraft should contain the Minecraft version number that this version of your pack uses.

<mainclass> (Optional)

This specifies the main class java calls when launching your pack.

<extraarguments> (Optional)

This specifies extra arguments to pass into Minecraft when launching.

<memory> (Optional)

This specifies the minimum amount of memory/ram to use when launching Minecraft in MB as an integer. If the user has set a larger memory/ram size it will use their set amount. If this is set to more than the user can have, then it will prompt the user that the pack has a minimum and the pack may fail to work correctly.

<permgen> (Optional)

This specifies the minimum amount of PermGen size to use when launching Minecraft in MB as an integer. If the user has set a larger PermGen size it will use their set amount.

<libraries>

This section deals with needed libraries for installs using 1.6 versions of Minecraft and later. Version 1.6 of Minecraft changed a lot of things and added in Libraries which are essentially jar files that contain code needed for Minecraft or Mods to run.

Here you can specify additional libraries which should be downloaded and installed with you pack, wrapped in <library /> tags. These aren't shown to the user as selectable mods, rather installed by default in order to make Minecraft and your mods work. Each <library /> tag can contain the following elements.

To make this process easier, you can login to the Admin area and use the 'Forge Libraries' link and input the Minecraft version number and the Forge version number to do this section automatically.

url=""

The URL of the file for the ATLauncher to download. Must be a file that is downloaded without user intervention.

file=""

The filename of the downloaded file.

server="" (Optional)

If this library should be installed in the server then the folder stub should be put here. For instance if the file must go in a folder called, net/test/test/1.2/test-1.2.jar then the server should be server="net/test/test/1.2/test-1.2.jar "

md5="" (Optional)

This is optional and should be in all lowercase, but should be included where possible in order to maintain file integrity and reduce errors in installation. This should be the MD5 sum of the file. You can use any tool you like to generate the MD5 sums (Windows, Mac, Linux) or it's provided in the admin area.

depends="" (Optional)

This specifies a mod name which if isn't installed will mean this library won't be installed.

download="" (Optional)

This specifies the download method that should be used. It can be one of 2 different values (direct, server). Direct will download the file from an external server directly and server will download a file directly from our servers that you've uploaded. If this attribute isn't provided, it will default to direct.

<colours>

This section deals with the use of colours for mods for use in the mod selection screen.

You can colour the mods so that they stand out or to easily distinguish which mods are part of a group for example. The use of this is completely up to you.

Each defined colour has a name and a code as defined below.

name=""

This is the name of your colour and is used in the colour=”” section of your <mod /> to identify and know which colour to use.

code=""

This is the RGB colour code of the colour you wish to use with a #, for instance #FF0000 is red. You can get the colour codes from this website. It should ALWAYS contain 6 characters which can only contain the following characters (0-9ABCDEF)

<mods>

This section is more complicated than the pack section and contains all the information needed by the ATLauncher to play with and install your pack, as well as all the information needed to correctly attribute the mod creators.

Each of the different mods are defined by a <mod /> tag with each of the variables below inside the tag within quotes.

name=""

The name of the mod.

version=""

The version of the mod.

url=""

The URL of the file for the ATLauncher to download or open in the users browser.

file=""

The filename of the downloaded file.

website=""

The website of the mod. This should only be for the OFFICIAL website of the mod.

donation="" (Optional)

This is optional, but should be included where possible. This is a link to a page with the donation link or button for the mod.

colour="" (Optional)

This is optional and contains the name of the colour defined in the <colours></colours> section of the XML to colour the mod in the mod selection screen.

md5="" (Optional)

This is optional and should be in all lowercase, but should be included where possible in order to maintain file integrity and reduce errors in installation. This should be the MD5 sum of the file. You can use any tool you like to generate the MD5 sums (Windows, Mac, Linux) or it's provided in the admin area.

type=""

This is the type of mod it is and how it should be installed. There is a bunch of different values that can be specified here and is constantly changing. For a full list of mod types please see the 'Mod Types' section.

client="" (Optional)

This specifies if the mod should be installed on the client. Defaults to "yes" but you can exclude this mod from being installed on the client by specifying client="no".

server="" (Optional)

This specifies if the mod should be installed when installing the server of the pack. If this isn't specified it will assume that the mod should be installed on the server. If you don't want to install this mod on the server then specify it with the keyword 'no' (without quotes). If this mod requires a separate download for the server side, then you can specify that with the keyword 'seperate' (without quotes) and specifying the below attributes.

serverurl="" (Only if server="seperate")

This specifies the url of the file for the ATLauncher to download or open in the users browser for the server portion of the mod.

serverfile="" (Only if server="seperate")

This specifies the filename of the downloaded server file.

servertype="" (Only if server="seperate")

This specifies the type of file of the server mod. Again this can have a bunch of different values and for a full list see the 'Mod Types' section.

optional="" (Optional)

This allows the user to not use this mod if they don't want to. If this isn't specified then it will assume that it's not optional. To make a mod optional, use the keyword 'yes' (without quotes) and if you want to make it mandatory then you can use the keyword 'no' (without quotes).

serveroptional="" (Optional)

This works the same as it does for the original optional attribute, but only applies to when creating a server. It defaults to whatever the optional attribute is set to but can be set to 'yes' to make it optional when installing a server or 'no' to make it not optional.

download=""

This specifies the download method that should be used. It can be one of 3 different values (browser, direct, server). Browser will open the file in the users browser, direct will download the file from an external server directly and server will download a file directly from our servers that you've uploaded.

group="" (Optional)

This tells the Launcher if this mod is part of a group, meaning that mods with the same group can only have one of the group selected. So if you have Mod A and Mod B in the group Mods then only one of them can be selected. Upon selection of one of the group, the rest will be auto unselected.

hidden="" (Optional)

This tells the Launcher if this mod should be hidden from the mod select screen or not. Use the keyword 'yes' to hide it from the mod select screen.

library="" (Optional)

This tells the Launcher if this mod is a library. Should be used with the depends attribute. These mods are hidden by default.

linked="" (Optional)

This allows you to set a mod which this mod is a child off. The mod listed here must be selected before this one can. Simply insert the exact name of the mod (as defined in the name=”” attribute of the parent mod) in this field and it won’t allow users to install the child mod unless the parent mod is installed. Due to limitations in the Launcher, all child mods must be listed in your XML file directly under the parent mod. It’s important to know that the linked and depends attributes are very different. Depends auto selects parent mods whereas linked doesn’t.

depends="" (Optional)

This allows you to set a mod which this mod depends on being installed. Simply insert the exact name of the mod (as defined in the name=”” attribute of the parent mod) in this field and it will auto select the mods listed in this field. You can specify multiple dependencies by listing the name of the mods separate by commas with no spaces. Mods which are listed here will most likely have the library attribute so that they are hidden from the users view. It’s important to know that the linked and depends attributes are very different. Depends auto selects parent mods whereas linked doesn’t.

recommended="" (Optional)

This specifies if the optional mod should be selected when the Select All button is pressed. It defaults to yes so if you want to make a mod not get selected upon pressing the Select All button then set recommended=”no”.

description=""

This allows you to set a description for the mod and is required. The ATLauncher allows the user to see a short description of the mods in your pack. This should be filled with a short and accurate description of the mod so that users who aren’t familiar with the mod can see briefly what it’s about.

Mod Types

The type="" attribute for each of the mods in your pack tells the ATLauncher how to deal with the mod and how to install it. Some mod types contain extra attributes which must be added to your <mod /> declaration in order for it to work.

The ATLauncher website doesn’t check to make sure these work, so you must make sure and test yourself to make sure they work and install correctly.

type="jar"

Specifies that this mod file should be extracted into the minecraft.jar file.

type="forge"

Specifies that this mod file is Minecraft Forge and should be installed in the Jar or placed in the Root directory for Minecraft >1.6 server installs.

type="mods"

Specifies that this mod file should be placed into the ‘mods’ folder.

type="coremods"

Specifies that this mod file should be placed into the ‘coremods’ folder.

type="mcpc"

Specifies that this mod file is MCPC and should be used as the main server jar.

type="plugins"

Specifies that this mod file should be placed into the ‘plugins’ folder for MCPC.

type="extract"

Specifies that this mod file should be extracted and it's contents stored somewhere.

extractto=""

Specifies where the extracted contents should go. It can be one either:

extractto="root" – Which puts them in the .minecraft/ folder.

extractto="mods" – Puts the files in the mods/ folder.

extractto="coremods" – Puts the files in the coremods/ folder.

extractfolder=""

Specifies which folder within the extracted file that should be acted upon. Any folder slashes should be replaced with %s%, so for instance if there is a folder in the zip file called SomeMod/Put In Mods Folder/ which contains all the files which need to go in the mods folder then you would have extractfolder="SomeMod%s%Put In Mods Folder%s%". A trailing %s% should always be added.

type="decomp"

Specifies that the file should be extracted and it's contents acted upon. This is a more advanced version of the 'extract' type and allows you to specify which file from the extracted contents to work with and what to do with it.

decompfile=""

Specifies the file from the extracted contents to work with.

decomptype=""

Specifies the action to perform on the file specified above. This can be either jar, mods or coremods.

type="millenaire"

Special install method for Millenaire only. This deals with the special characters inside the zip file and installs it as needed.

type="texturepack"

This installs the mod in the texturepacks folder for Minecraft <1.6

type="resourcepack"

This installs the mod in the resourcepacks folder for Minecraft >=1.6

type="texturepackextract"

This extracts the file into a temporary folder and at the end of the install process will zip up all the file and place it in the texturepacks folder for Minecraft <1.6

type="resourcepackextract"

This extracts the file into a temporary folder and at the end of the install process will zip up all the file and place it in the texturepacks folder for Minecraft <1.6

Other Things To Note

Other things you should note and keep in mind is that this must follow XML standards which means that values cannot have certain characters in it otherwise it will break the file and not parse. If you have any of the characters below in any of your values anywhere then you must replace them with the text next to it.

- < must be replaced with &lt;
- > must be replaced with &gt;
- ‘ must be replaced with &apos;
- “ must be replaced with &quot;
- & must be replaced with &amp; EXCEPT when used in any of the above

You also don’t have to indent your code, the Launcher will read it just fine if it’s indented or not. You should make a habit of indenting your code though so that it’s easier to read. When indenting your XML file, you should use spaces rather than tabs. If you use the Advanced Editor in the admin area then that is done for you automatically.

Rules For Packs

When creating and updating your pack/s, you should keep in mind the following set of rules. If you fail to follow these rules, your pack/s may be removed temporarily or permanently from the ATLauncher and the website. These rules were made in order to protect other packs in the ATLauncher and to also protect relevant parties copyrights.

- You may not upload any of Mojang's property (including the minecraft.jar) onto our servers for use in the ATLauncher.
- If you upload mods to the ATLauncher server in order to use in your pack, you must have the relevant permissions to do so. Screenshots of permissions must be maintained by you at all times and must be provided if asked by any of The Astocky Team (astocky, dwinget2008, haighyorkie or RyanTheAllmighty). If you fail to have permissions, the offending versions of your pack and the files will be removed. The only exception for this is mods which are on the "Allowed Mods" list in the ATLauncher Admin Help area and if your pack is a private server pack.
- You may not make a pack which closely resembles any of the other packs within the ATLauncher. If any pack creator lodges a complaint against your pack being too similar then The Astocky Team will decide on what to do if you don't want to change it. This is done in order to protect people's packs and to make sure there isn't multiple packs with the same mods in them.
- If someone approaches us complaining about use of their copyrighted work (such as mod makers) which is infringing in your pack, then you must remove it within 24 hours of being sent the notice.

Please keep an eye on our Twitter for changes to this document to reflect new rules as they come into effect.

Example XML Pack Specification

```
<version>
   <pack>
      <version>1.0.0</version>
      <minecraft>1.0.0</minecraft>
      <mainclass>net.minecraft.main.Main</mainclass>
      <extraarguments>--forge-runtime=forge.lib.jar --test=test</extraarguments>
   </pack>
   <libraries>
      <library file="some-library.jar" url="http://www.somelibrary.com/library/some-library.jar" server="library/some-library.jar" md5="43586987123047b21" />
   </libraries>
   <mods>
      <mod name="Some Mod" version="1.0.0" url="http://files.atlauncher.com/packs/SomePack/files/SomeMod1.0.0.zip" file="SomeMod1.0.0.zip" website="http://www.somemod.com" donation="http://www.somemod.com/donate/" type="mods" server="seperate" serverurl="http://files.atlauncher.com/packs/SomePack/files/SomeModServer1.0.0.zip" serverfile="SomeModServer1.0.0.zip" servertype="jar" optional="no" directdownload="yes" description="Some Mod bring us Something to Minecraft. Finally you can do something in Minecraft with the aid of Some Other Thing!" />
   </mods>
</version>
```

API Introduction

This section of the documentation deals with the API used to control, edit and update your packs.

Please note that this specification can change at anytime and you should keep track of updates to this spec by following our Twitter account (@ATLauncher).

If you have any questions about this specification such as errors you may find or any suggestions you may have then please feel free to post a thread in the forums (http://www.atlauncher.com/forums/) or email us (contact@atlauncher.com).

How To Use The API

Using the API is as simple as making a POST call to the url https://api.atlauncher.com/api.php with 2 variables, your API key and the action you wish to perform.

Your API key can be found in the Account section in the Admin area where you change your password and email.

The API Key must be sent as a 'key' attribute, and the action must be sent as a 'action' attribute.

Various different actions are available, some with their own arguments you need to provide. See below for a full list of each action and their required attributes.

Once a call is made, you will receive a JSON output back from the API. This contains the information including any error messages and the output of your API call.

All API calls have a "responseCode" key sent back which contains the status of your API call. There are 4 possible outputs of this key which are:

- 200 – Meaning everything went well
- 400 – Meaning not all the attributes required were sent
- 403 – Meaning your API key is incorrect
- 404 – Meaning the action specified wasn't found

Anything but a 200 responseCode indicated that an error occurred, and an appropriate message is give in the 'errorMessage' key.

API Actions

This section details all the actions you can send to the API through the action argument and what they do as well as any extra arguments that you must send along with it.

listActions

This lists all the actions you can perform on the API.

Required Arguments: None

listPacks

This lists all the packs you have on the ATLauncher and lists their details.

Required Arguments: None

listPackVersions

This lists all the versions for the specified pack id.

Required Arguments: packid (The ID of the Pack to list versions for)

getVersionXML

This gets the XML for a published version of a pack.

Required Arguments: versionid (The ID of the Version to get the XML for)

getDevVersionChangelog

This gets the changelog for a published version of a pack.

Required Arguments: packid (The ID of the Pack to get the dev changelog for)

getDevVersionXML

This gets the XML for the development version of a pack.

Required Arguments: packid (The ID of the Pack to get the dev XML for)

getVersionChangelog

This gets the changelog for a version of a pack.

Required Arguments: versionid (The version ID of the Pack to get the changelog for)

getPackTesters

This gets a list of usernames of those who have are testers for a pack.

Required Arguments: packid (The ID of the Pack to get the testers for)

setPackTesters

This sets the usernames of testers for a pack.

Required Arguments: packid (The ID of the Pack to get the testers for), testers (A comma separated list of the usernames of people who are testers for this pack)

addPackTester

This adds a user to the testers for a pack.

Required Arguments: packid (The ID of the Pack to add the tester to), username (The username of the user to add as a tester for this pack)

removePackTester

This removes a user from the testers for a pack.

Required Arguments: packid (The ID of the Pack to remove the tester from), username (The username of the user to remove as a tester for this pack)

getPackAllowedPlayers

This gets a list of usernames of those who are allowed to install a pack for private packs.

Required Arguments: packid (The ID of the Pack to get the allowed players for)

setPackAllowedPlayers

This sets the usernames of users who are allowed to install a pack for private packs

Required Arguments: packid (The ID of the Pack to get the testers for), players (A comma separated list of the usernames of people who are allowed players for this pack)

addPackAllowedPlayer

This adds a user to the allowed players for a pack.

Required Arguments: packid (The ID of the Pack to add the user to), username (The username of the user to add as an allowed player for this pack)

removePackAllowedPlayer

This removes a user from the allowed players for a pack.

Required Arguments: packid (The ID of the Pack to remove the tester from), username (The username of the user to remove as an allowed player for this pack)