

# IzPack Utilities

## IzPack utilities

The IzPack project includes a set of utilities that you may find useful.

These projects live outside the regular IzPack installer Subversion repository. You can access it from <http://svn.berlios.de/svnroot/repos/izpack/izpack-utils/> instead of <http://svn.berlios.de/svnroot/repos/izpack/izpack-src/> for IzPack itself.

They will be shipped in the official IzPack releases under their own pack.

## Windows executable wrapper (izpack2exe)

### Description

The 7-Zip project (see <http://www.7-zip.org/>) provides a so-called SFX for installers, i.e., an image that can be used to create self-extracting Windows executables. Once its content has been extracted, such a self-extracting executable can launch an executable or a file. In the later case, it is assumed that there exists an association between a file extension and a software component.

7-Zip SFX for installers works as follows:

1. an executable image is provided
2. a configuration file has to be written, specifying among other things the executable or the file to be launched after the extraction phase
3. a 7-Zip archive containing the files (including the executable or file to be launched)
4. these files simply need to be concatenated to form a SFX Windows executable.

The IzPack Windows executable wrapper takes an IzPack-generated installer JAR file, and wraps it inside a Windows SFX executable. This has several advantages, among these two ones:

1. some users may find it strange to have a JAR and not an executable
2. by naming it with *setup* or *install*, Windows VISTA will perform a rights elevation whereas IzPack cannot enforce it when launched as a regular JAR.

And of course, you can ship a JAR and a Windows executable from the very same IzPack installer descriptor!

We have customized the SFX image from the 7-Zip project as follows:

- we have changed the icon
- we have customized some strings to mention that this is a customized version.

While the rest of the work is licensed under the Apache License version 2, the sole SFX module is licensed under the Lesser GNU General Public License version 2 or later as required by the 7-Zip project.

### Requirements

**izpack2exe** is written in Python. It depends on 7-Zip and optionally UPX, a tool that can compress executables.

In official IzPack releases, we provide this tool *batteries-included* on Windows, i.e., we will provide:

\***izpack2exe** as a Windows executable, so that you don't need to install Python, and

- 7-Zip and UPX executables, so you don't have to download them.

## Usage

The usage is quite easy:

```
usage: izpack2exe.py [options]
```

```
options:
```

```
  -h, --help          show this help message  
and exit  
  --file=FILE         The installer JAR file /  
files (1 per bundled file, specify at least  
the installer JAR!)  
  --output=OUTPUT    The executable file  
  --with-7z=P7Z      Path to the 7-Zip  
executable  
  --with-upx=UPX     Path to the UPX  
executable  
  --no-upx           Do not use UPX to further  
compress the output  
  --launch-file      File to launch after  
extract (e.g., native launcher)
```

A typical wrapping will be done like:

```
izpack2exe --file=installer.jar
```

## Mac OS X Application bundle wrapper (izpack2app)

### Description

**izpack2app** is the Mac OS X brother of **izpack2exe**. It bundles a JAR installer inside an application bundle, so that your installer will look like a regular Mac OS X application.

To do that, we started from the Mac OS X *Jar Bundler* tool that you can find under */Developer/Java*. We bundled a Jar installer, then stripped it. Indeed, an application bundle is nothing else but a structured set of files and directories.

What **izpack2app** does is simply:

1. copy the bundle files structure,
2. put your Jar at the good place, and
3. edit the *Info.plist* to reference your Jar.

## Requirements

**izpack2app** is written in Python. It does not require any third-party module to work. Python is bundled with Mac OS X and is mainstream on Linux and other Unix variants. We ship it as a Windows executable in the official IzPack releases (and of course also as a Python application!).

## Usage

The usage is straightforward:

```
izpack2app.py installer.jar Installer.app
```

wraps *installer.jar* as *Installer.app*

## Java Web Start JNLP file generator (izpack2jnlp)

### Description

**izpack2jnlp** is able to generate Java Web Start JNLP files so that IzPack-based installers can also be shipped via Java Web Start.

### Requirements

**izpack2jnlp** is written in Python and does not need third-party modules to work.

Please note that it only generates JNLP files: **you must sign your installer JAR** before you actually upload them to a web server as **Java Web Start refuses to launch unsigned JARs**.

### Usage

You need to pass a few arguments on the command-line, run:

```
izpack2jnlp.py --help
```

to see them.