

# Building an IPS manifest

When I build a third-party package, I publish it into an IPS repo using the following steps:

1. Put the metadata into a file called "metadata". Most of this information can come from a template file, eg a build Makefile or something. For example:

```
set name=pkg.fmri value=my-package@$version
set name=pkg.description value="My Great Package"
set name=description value="My Great Package"
license /path/to/license/file license=BSD
```

2. Put the package dependencies into a file called "deps". This is hard to generate automatically, one (good, but incomplete) approach might be to use ldd to find all the linked libraries and then use pkg search to find which package provides them. I tend to calculate the dependencies once and commit them as part of the build. For example:

```
depend fmri=pkg:/library/zlib type=require
depend fmri=pkg:/system/library/math type=require
depend fmri=pkg:/library/expat type=require
```

3. Do a "make install" into a local install root, for example:

```
DESTDIR=insroot make install
```

4. Move to the install root and run the attached script to generate the file "payload". For example:

```
perl dir2manifest.pl usr > ../payload
```

5. Join "metadata", "deps", and "payload" together into "manifest":

```
cat metadata deps payload > manifest
```

6. Upload the package:

```
pkgsend -s $REPO publish -d insroot --fmri-in-manifest manifest
```