

How to create a classpath

For Java, the classpath argument is simply a list of paths joined with an OS-specific path separator:

Paths

```
cp = paths.join(File::PATH_SEPARATOR)
```

This assumes `paths` points to files and/or directories, but what if you have a list of artifact specifications? You can turn those into file names in two steps. First, use `artifacts` to return a list of file tasks that point to the local repository:

artifacts

```
tasks = Buildr.artifacts(specs)
```

Next, map that list of tasks into list of file names (essentially calling `name` on each task):

Tasks

```
paths = tasks.map(&:name)
```

This works as long as the artifacts are already in your local repository, otherwise they can't be found, but you can ask Buildr to download them by calling `invoke` on each of these tasks:

invoke artifacts

```
tasks = Buildr.artifacts(specs).each(&:invoke)
```

So let's roll this all into a single line:

One Liner

```
cp = Buildr.artifacts(specs).each(&:invoke).map(&:name).join(File::PATH_SEPARATOR)
```