

ApplyRedirectorExample

Redirector example: copies files by using the apply task to execute `cat` with input redirected from source file `src/*.src` and output redirected to `dest/*.out`.

```
<project default="copycat">

  <target name="copycat">
    <apply executable="cat" addsourcefile="false">
      <redirector>
        <outputmapper>
          <mapper type="glob" from="*.src"
            to="dest${file.separator}*.out" />
        </outputmapper>
        <inputmapper>
          <mapper type="glob" from="*"
            to="src${file.separator}*" />
        </inputmapper>
      </redirector>
      <fileset dir="src" includes="*.src" />
      <mapper type="glob" from="*.src"
        to="dest${file.separator}*.out" />
    </apply>
  </target>

</project>
```

This uses three mappers and a fileset. The fileset identifies the files to be processed. The fileset's `dir` attribute tells it what directory the files are in. The fileset produces a list of source files, e.g. "a.src" "b.src", that are used by the mappers.

The apply task's mapper is used to determine which files are already up to date. It maps each *.src file to dest/*.out.

Note that `${file.separator}` must be used if you need this to work cross-platform. Mapper doesn't do the slash-backslash translation for you.

The mapper used by outputmapper identifies the file to redirect output to. It maps each *.src file to dest/*.out. This is identical to the mapper the apply task uses for determining dependencies. Alternately you could define a single mapper and reference it using `refid` for both of these mappers.

The mapper used by inputmapper just maps each source file "*" to the file with the same name in the src directory, 'src/*'. This is essentially an identity mapping, but the files passed to the mapper consist of just the filename, so the directory name needs to be prepended to pick up the right file.

By default, the apply task passes the source file name as a command line argument to its executable for each execution. We need to inhibit this behavior for this example by setting the `addsourcefile` attribute false, so that `cat` will take input from the input redirect instead of reading the named file. (The result would be the same in this case, but we're trying to demonstrate input redirect.)

Note One can use `<globmapper>` instead of `<mapper type="glob">`, and with this one can use the "handledirsep" attribute.