

# ME

commons-me - Provides JaveME helpers, and implementations of JavaSE functionality that JavaME lacks but programmers want

## Getting It

The commons-me project is in the commons Sanbox:

<https://svn.apache.org/repos/asf/commons/sandbox/me/trunk>

## Features Covered

## Using It

Simply place the Jar in the lib directory of your JavaME project. When building your JavaME Jars, the WTK / Antenna should unpack the commons-me Jar, pre-verify the classes, and include it in the Jar of your application. As the Jar will have been built for Java 1.1, against the JavaME classpath, it should work just the same as any code you yourself write.

## Testing It

You will need to have installed a copy of the [Sun WTK](#) to build or test the library

To unit test the library, we load up a standard JavaSE runtime, but with the JavaME classpath from the WTK at the start of the classpath. This allows us to use reflection etc in our tests, as required by junit, but running the core code against JaveME

All code under `/src/` should be for JavaME. All code under `/test-src/` can be JavaME or JavaSE

## Proposal

See <https://svn.apache.org/repos/asf/commons/sandbox/me/trunk/PROPOSAL.html>

## Porting other libraries to JavaME

Another part of the project is to take existing open source libraries under the ASL, and tweak them to work on JavaME.

- [Jakarta ORO](#) Porting suggested, no work as yet

## FAQ

*Add questions and answers here*