Ports - Porting #280

[openjdk6] have a working Java implementation

2012-12-28 02:27 AM - mtjm

Status:	open	**% Done:**	0%
Priority:	critical		
Assignee:			
Category:			

Description

GCJ can be compiled (even cross compilation works with some changes: libjava expects to be able to call ./gcj-dbtool, calling gcj-dbtool instead works; it also needs $(GCJ) to be a cross gcj), it needs --enable-java-home and --with-ecj-jar (see http://gcc.gnu.org/install/configure.html, the eclipse-ecj package provides an ecj.jar that cannot be used instead of this one). A trivial javac wrapper script to call ecj is needed (shouldn't it be built by the package?). (I haven't tried using the PKGBUILD from pcr for it on mips64el, used it on x86 to build the cross gcj. I think there is no need for a GCJ package.)

However, GCJ provides only a 1.5.0 JDK, while recent openjdk6 (the package using IcedTea) needs 1.6.0. Disabling the version check leads to compilation errors. (It also wanted to use programs at incorrect paths like /bin/egrep, I don't know a proper way to fix it.)

I think we should try one of these solutions:

- cross compile IcedTea6, it doesn't seem easy
- build an older release of IcedTea6 that doesn't need a 1.6.0 JDK (and that has the Zero VM) and use it to build a newer one
- implement missing GCJ features, I don't know what would be needed for it
- try another solution that I don't know about

The aim is to have both 6 and 7 JDKs built: they have different uses. (This won't be enough to have a fast JDK, probably Shark will be the easiest way, although it needs porting LLVM.)

(Assigning high priority since many packages including LibreOffice depend on this and our users asked about using a YeeLoong for Android development.)

History

#2 - 2016-05-11 11:50 PM - Anonymous

- Project changed from 3 to Ports