

itools-R6.3 README

Paul DuBois
paul@kitebird.com

Revision date: 2001-11-19

Introduction

The *itools* distribution contains *imake*, some related configuration programs, and a set of X11 configuration files. Instructions for building and installing the distribution may be found in Appendix B of the "boa book":

```
Software Portability with imake (2nd edition)
Paul DuBois
O'Reilly & Associates, 1996
ISBN 1-56592-055-4
```

Appendix B is also available online at:

```
http://www.dubois.ws/software/imake-book/
```

or at the same location from which you obtained the *itools* distribution.

Most of this distribution was derived directly from the distribution of the X Window System, Version 11, Release 6.3, patchlevel 1.

IMPORTANT: To build this distribution, you must have an ANSI C compiler. If you don't have one, or you encounter other problems building the distribution, try building the *itools-R6.1* distribution instead. (If you have problems other than the lack of an ANSI C compiler, please send me a description of what goes wrong. You may have found a bug in the distribution that can be fixed.)

Before You Build

Before you build the distribution, there are a few things you should check.

Look in the *config/misc* directory to see if there are any notes for the type of system you're using. If there are, read them.

Take a look at *config/cf/site.def* and read the comments to see if there's anything you'd like to modify.

Take a look at *config/extras/extras.tmpl* and check the following:

All systems:

- If you're going to install *imboot* and you don't want to use */usr/local/lib/config* as the *imboot* configuration root directory, change the value of `CONFIGROOTDIR`.

UNIX systems:

- Make sure the value of `PERLPATH` is correct for your system. This is needed for building Perl scripts. It can be the full pathname of either Perl 4 or Perl 5.

- If you want to use the *sh* version of *imboot*, the "#define UsePerlImboot" line should be commented out:

```
/*#define UsePerlImboot YES*/
```

If you want to use the Perl version of *imboot*, the line should be uncommented:

```
#define UsePerlImboot YES
```

- If you want to use the *sh* version of *xmkmf*, the "#define UsePerlXmkmf" line should be commented out:

```
/*#define UsePerlXmkmf YES*/
```

If you want to use the Perl version of *xmkmf*, the line should be uncommented:

```
#define UsePerlXmkmf YES
```

Windows NT systems:

- If you have the Windows NT version of Perl, you can use the Perl script *indent* and the Perl versions of *imboot* and *xmkmf*. Uncomment the "#define WinNTHasPerl" line:

```
#define WinNTHasPerl YES
```

If you don't have Perl, the line should be commented out:

```
/*#define WinNTHasPerl YES*/
```

In the latter case, you won't be able to use *xmkmf*, *imboot*, or *indent*.

Building the Distribution

First of all, do ***NOT*** run a bootstrapper like *xmkmf* or *imboot* to try to generate a *Makefile* from the *Imakefile* in the distribution root directory. The *Makefile* in the root is written by hand; running a bootstrapper destroys it and you will not be able to build the distribution. If you do run a bootstrapper inadvertently, you can restore the original *Makefile* like this:

```
% cp Makefile.sav Makefile
```

For UNIX, the command for building the distribution is:

```
% make World (or)  
% make World BOOTSTRAPCFLAGS="flags"
```

where "flags" is something like "-Dhpux", "-Dsun", etc. (See Appendix B if you need a flags value.)

For Windows NT, the command for building the distribution is shown below, but before you run it, make sure you read *config/misc/Win32.notes*:

```
nmake World.Win32 > world.log
```

Installing the Distribution

For UNIX, run this command (you may need to be *root*) to install everything built by the build step:

```
% make install
```

For Windows NT, run this command:

```
nmake install
```

To install everything that is built in a particular directory, change into that directory and run "make install" (or "nmake install" for Windows NT) there.

You can also install individual targets. For example, if you want to install a program "prog", you can change into the directory in which the program is built, then run "make install.prog" under UNIX. Under NT, use "nmake install.prog.exe" or "nmake install.prog.bat", depending on whether prog is an executable binary or batch script.

Miscellaneous

Some of the differences between the itools distribution and the original X stuff from which it was derived:

- The *config/misc* directory contains notes for various systems that might be useful or of interest.
- The *config/extras* directory (containing *imboot*, *msub*, and *indent*) is not part of X.
- The Perl version of *xmkmf* (in *config/util*) is not part of X.
- *which* (in *config/util*) is not part of X.
- Most of the original files were mode 444 (read only), the presumption being that you would just use them to build and install X, but not modify them. I'm presuming that you may very well want to mess around with anything that's here, consequently all the files are writable.
- Many of the Imakefiles and a few other files were changed from the originals in the X11R6.3 distribution. (You can usually tell that a file *xxx* was modified if a file *xxx.orig* is present.)
- The set of configuration files in *config/cf* includes *linux-pmac.cf* for Linux/PPC systems, and the Linux vendor block in *Imake.cf* was modified to recognize when to use it. (Linux/PPC support is not provided in X11R6.3 at present.)
- *imake* source has been patched to strip leading zeroes from version numbers written to *Imakefile.c*, so that *cpp* won't interpret them as octal constants.
- *makedepend* source has been patched not to have a conflict with the `getline()` function declared in *stdio.h* of glibc. (I changed `getline()` in the source to `mdgetline()`.)
- *mkdirhier.c* source has been modified to test the return value from `_mkdir()` differently (to eliminate spurious errors).