Entwicklungswerkzeuge

Sprache	Compiler
C	Open Watcom
C++	Open Watcom
FORTRAN	Open Watcom
Pascal	FreePascal

Andere Sprachen dürfen verwendet werden, um Teile des Betriebssystems zu entwickeln. Allerdings müssen sie Open-Source sein und die OS/2-API verwenden. Die Verwendung dieser freien Werkzeuge ist nicht verboten aber nicht empfohlen. Benutzen Sie die Open Watcom-Tools umd die osFree-Quellen zu übersetzen.

Wenn konkurrierende Werkzeuge existieren (z.B. NMAKE und WMAKE) benutzen wir immer die Open Watcom Tool.

Herunterladen und Übersetzen

In unregelmäßigen Abständen können Schnappschüsse der Quellen von der Seite heruntergeladen werden oder die jeweils aktuellste version per CVS. Die osFree-Quellen sind auf dem Netlabs CVS Server zu finden.

Vor dem Übersetzen der Quellen überprüfen Sie bitte compile.cmd und ändern Sie libpath, dass es auf die OpenWatcom-Bibliotheken zeigt. Führen Sie nach diesen Änderungen einfach compile.cmd aus und der Übersetzungsprozess wird gestartet.

Der Verzeichnisbaum

Bitte schauen Sie sich den CVS Code-Baum an um zu verstehen, wo Dateien und Ordner gespeichert werden müssen. Der osFree CVS Verzeichnisbaum enthält den Quellcode für das Betriebssystem und einige Werzeuge. Bitte speichern Sie KEINE anderen Werkzeuge oder Anwendungen in diesem Verzeichnisbaum.

Global/Shared/Private Headers

Each level of the CVS tree contains two standard directories:

shared	Contains code shared among all source at this level and deeper levels
include	Contains header files for the above

Each levels/part of the os should have a specific prefix that allows a developer to easily find what part of the os a header/library file belongs to. For example code shared by the whole tree should be included with:

```
#include <all_shared.h>
```

and code shared by all commandline tools should include:

```
#include <cmd_shared.h>
```

Try to create as few shared code headers as possible. Each “shared” directory should contain one (1) library (.lib) file (xxx_shared.lib) with all shared code and each “include” directory should contain one main header file including all other (xxx_shared.h).

Example of use of common files:

Use the normal OS/2 INCL_ since our toolkit is the OS/2 toolkit #define INCL_DOSERROR Do NOT include os2.h, use osfree.h instead

```
  #include <osfree.h>
```

Include any needed normal C library <code> #include <malloc.h> #include <string.h> </code> Include all shared code and shared code for command line tools

```
#include <all_shared.h>
#include <cmd_shared.h>
```

Dokumentation

Quellcode, der nicht wiederverwendet wird, also nur an dieser Stell gebraucht wird, soll direkt in den Qullen dokumentiert werden.

Verteilter Code (egal ob nur auf dieser Ebene oder global auf allen Ebenen) sollte direkt in den Quellen und in einem “erstellen und entwickeln” - Dokument dokumentiert werden.

Die API des Betriebssystems und die Dokumentation der Werkzeuge sollte NICHT im Quellen-Baum hinterlegt sein, sondern im Werkzeuge- und Veröffentlichung-Baum.

Quellcode sollte direkt in der Quellcode-Datei (nicht in den header-Dateien) dokumentiert werden.

Jeder Funktion soll ein Kommentar vorangehen, in der erklärt wird, was die Funktion tut, welche Parameter sie verwendet (Ain- und Ausgabe) und in dem alle externen Quellen stehen, die verwendet werden.

Bitte fügen Sie Kommentare in den Code ein, die es dem Leser ermöglichen, die Logik zu verstehen, aber übertreiben Sie nicht.

Die Dokumentation sollte unter Berücksichtigung der Internationalität dieses Projekts in erster Linie auf Englisch erfolgen, innerhalb der Quellen in jedem Fall.

When Developing

Use static linking, do not use dynamic libraries (LIBC style) or dynamic runtime.

Use the makefiles provided with the source tree, don't “do your own”.

Currently osFree development is done on OS/2 (minimum Warp 4) but in the future development will

be hosted on osFree.

We use CVS to share code among developers.

We use Doxygen and HTML to document our work.

Submitting a Patch

- Make sure your changes follow the coding guidelines above.
- Make sure you are using the current versions of the sources so that the resulting diffs are comparing your changes with the head of the source tree.
- Create your patch either by using cvs diff -u (if you are using CVS) or diff -u original-file changed-file (if you are using a source archive - you can also create differences for the whole directory contents using diff -r) In the latter case include the old code first, the new code last – in the patch anything you added will be prefixed with a +.
- Remove all/any lines that reference files without changes.
- Send the patch file as an attachment in your email. Do not paste the patch directly into the email body.
- Maintainers will often reply in response to your patch, pointing out things to fix up, etc. before a patch can be checked in. Please always follow the maintainer suggestions closely and respond by sending a new corrected patch. Please do not expect the maintainers to rework your changes, you want to be able to claim all the credit for your great patches!

From:
http://osfree.su/doku/ - **osFree wiki**

Permanent link:
http://osfree.su/doku/doku.php?id=de:develop:guidelines

Last update: **2018/06/07 20:30**