

Первые шаги

Перед тем как начать что-то кодировать, почитайте исходники и определите область, которая вас заинтересует для разработки. У нас есть очень широкий круг задач и, очень возможно, одна из них вас заинтересует. Только помните, что наша текущая главная задача – реализовать более или менее полный набор OS/2 API поверх микроядра L4.

Инструменты разработки

| Язык программирования | Компилятор |
|-----------------------|-------------|
| C | Open Watcom |
| C++ | Open Watcom |
| FORTRAN | Open Watcom |
| Pascal | FreePascal |
| REXX | ReginaREXX |

Другие языки тоже могут быть использованы для разработки частей ОС. Но они тоже должны быть Open Source и использовать OS/2 API. Использование платных и shareware (то есть, которые могут быть скачаны с многих мест, но стоят денег и лицензия которых имеет много ограничений) не рекомендуется. Желательно использовать инструменты из поставки OpenWatcom для сборки ваших исходников.

Когда существуют аналогичные инструменты (например NMAKE и WMAKE), всегда используйте инструменты из поставки OpenWatcom.

Получение и сборка исходных кодов

Сначала, вы должны загрузить нижеперечисленные инструменты для вашей платформы с соответствующих сайтов:

- <http://www.openwatcom.org/> для OpenWatcom
- <http://www.freepascal.org/> для FreePascal
- <http://regina-rexx.sf.net/> для ReginaREXX

Вы можете [загрузить](#) нерегулярно обновляемые снапшоты исходников с этого сайта, или самые последние версии с SVN. Исходники osFree находятся на [Sourceforge SVN](#). Sourceforge также позволяет скачать любую ревизию в виде .tar.gz архива.

Перед сборкой проверьте файлы setvars-<somename>.cmd и <somename>.conf, И поменяйте настройки (в основном, пути к инструментам разработки). После этого откройте сеанс командной строки и запустите setvars-<somename>.cmd и введите

```
wmake
```

начнется процесс сборки. Для очистки дерева исходников от созданных при сборке файлов наберите

```
wmake clean
```

Для более подробной информации о системе сборки см. документ [Система сборки](#).

Дерево каталогов

Посмотрите на код в SVN, для понимания принципа размещения файлов. Обратите внимание, что дерево файлов в SVN osFree состоит из исходных кодов операционной системы и инструментов тулкита. Пожалуйста, НЕ помещайте сюда приложения и инструменты, не соответствующие назначению тулкита. Туллит это вспомогательные утилиты, собираемые под ту систему, под которой ведется сборка и необходимые для сборки файлов ОС.

Глобальные/Общие/Приватные файлы заголовков

Каждый уровень дерева SVN содержит два стандартных каталога:

FIX THIS! Сейчас это немного не так!

| | |
|----------------|---|
| <i>shared</i> | Содержит код, общий для данного и более глубокого уровней вложенности каталогов |
| <i>include</i> | Содержит заголовки для всего выше перечисленного |

Каждая часть/уровень ОС должен иметь отдельный префикс (как каталог верхнего уровня), позволяющий разработчику легко найти часть ОС, к которой заголовочный/библиотечный файл принадлежит. Например, код, общий для всего дерева исходников, должен включать:

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

и код общий для утилит командной строки должен включать:

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

Старайтесь создавать чем меньше общих заголовочных файлов, тем лучше. Каждый "shared" каталог должен содержать один (1) библиотечный (.lib) файл (xxx_shared.lib) со всем общим кодом и каждая "include" директория должна содержать один главный заголовочный файл, включающий все другие (xxx_shared.h).

Пример использования общих файлов:

```
// 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
#include <malloc.h>
#include <string.h>

// Include all shared code and shared code for command line tools
#include <all_shared.h>
```

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

Документация

- Приватный код (не разделяемый ни с каким другим кодом) должен быть документирован только внутри исходника.
- Разделяемый код (разделяемый на одном уровне, или же разделяемый со всеми уровнями) должен быть документирован в исходнике и в документе “[Разработка и сборка исходников](#)”.
- ОС API и документация к утилитам тулкита HE должна находиться в дереве исходников, а должна быть в тулките и в комплекте каждого релиза.
- Исходный код должен быть документирован в нем самом (но не во включаемых файлах).
- Каждая функция должна предваряться специальным комментарием с описанием того, что она делает, какие параметры она использует (входные и выходные) и прочие внешние ссылки, которые она использует.
- Размещайте комментарии в исходном коде таким образом, чтобы тот, кто его читает, легко мог разобраться в логике, но не переусердствуйте в деталях.

В процессе разработки

- Используйте статическую линковку, не используйте динамических библиотек (LIBC style) или динамических рантаймов.
- Пользуйтесь мейкфайлами из дерева исходников, не изобретайте свои собственные “велосипеды”.
- На текущий момент разработка osFree происходит в среде OS/2 (как минимум, Warp 4), но в будущем разработка будет производиться в самой osFree (“self-hosting”).
- Мы используем SVN для совместной разработки.
- Мы пользуемся Doxygen и Wiki для документирования своей работы.

Submitting a Patch (FIX THIS!!!)

- 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:

<https://osfree.org/doku/> - **osFree wiki**

Permanent link:

<https://osfree.org/doku/doku.php?id=ru:develop:guidelines&rev=1403384711>

Last update: **2014/06/21 21:05**

