

# Особенности разработки дистрибутивов Linux для архитектуры Эльбрус

Андрей Савченко

ООО «Базальт СПО»

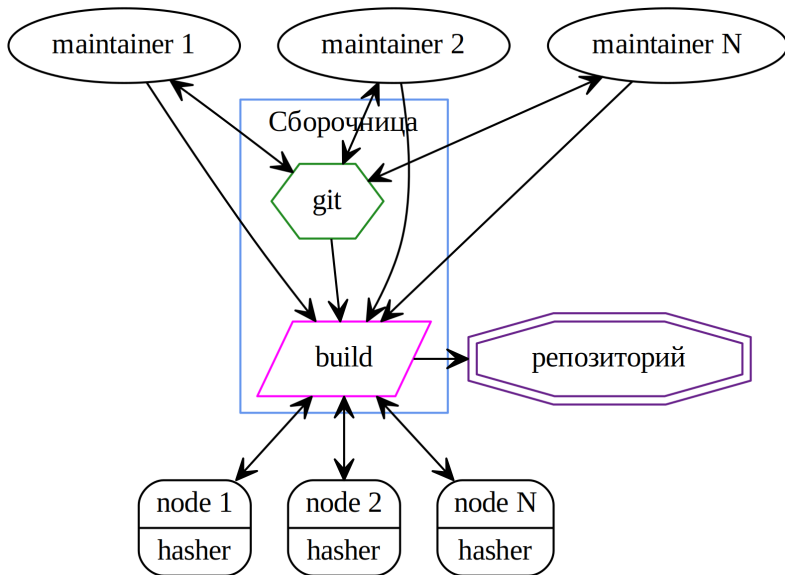
11 Июня 2019



- 1 Инфраструктура сборки
  - Сборка дистрибутивов Alt
  - Сборка Alt на Эльбрусе
- 2 Типовые проблемы портирования
  - Особенности компилятора
  - Особенности архитектуры
  - Взаимодействие с upstream



# Архитектура сборки Alt



## Управление пакетами:

- rpm
- apt
- GUI: synaptic, gnome package manager, ...

## Репозиторий пакетов: Sisyphus

- Собственный, независимый репозиторий

## Свободный доступ к исходным кодам:

- <http://git.altlinux.org>



# Особенности сборочницы e2k

- Обособленное сборочное оборудование
- Непубличный репозиторий пакетов
- Подстройка сборочницы под особенности e2k:
  - Неопределённый символ `__prof_ProfDirName` в `liberprof`
  - Догоняющий режим
- Родная сборка (native) ПО:
  - Меньше нагрузка по поддержке
  - Возможность профилирования ПО
  - Мощности e2k уже вполне достаточны для сборки на e2k



# Особенности сборочницы e2k

- Обособленное сборочное оборудование
- Непубличный репозиторий пакетов
- Подстройка сборочницы под особенности e2k:
  - Неопределённый символ `__prof_ProfDirName` в `liberprof`
  - Догоняющий режим
- Родная сборка (native) ПО:
  - Меньше нагрузка по поддержке
  - Возможность профилирования ПО
  - Мощности e2k уже вполне достаточны для сборки на e2k



Система программирования e2k: смесь проприетарных и основанных на СПО компонент.

Собираем из исходников (и своих патчей):

- glibc
- binutils
- выделяются из поставки lcc:
  - libstdc++
  - libgcc\_s
  - libatomic
  - libfcc
  - C runtime (crt)



# Текущий статус репозитория e2k

Всего пакетов: > 8000

Версии основных компонент:

icc	1.23	~ gcc-5.5
ядро Linux	4.9	
glibc	2.23	
gdb	8.1	
strace	4.20	
gtk	2.24	3.24
qt	3 / 4	5.9
boost	1.67	
perl	5.28	
python	2.7	3.7
firefox (+jit)	52.9	
libreoffice	5.4.3.2	





- lcc-1.23 ~ gcc-5.5
- Фронтенд EDG (Edison Group), бэкенд МЦСТ

Основные поддерживаемые компилятором стандарты:

- C11
- **C++11**, частично C++14
- Fortran-2008
- OpenMP-2.5



Исс нередко видит проблемы, которые пропускает gcc:

- особенно это касается предупреждений
- ⇒ приходится выключать `-Werror`

Препроцессор предполагает, что обрабатываемый код C/C++:

- табуляции меняются на пробелы
- ломается обработка Makefile и иных файлов
- решается через флаг `-xassembler` или изменением логики работы с препроцессором

Нет поддержки вложенных функций в C.



# Проблемы компилятора

Не все типы данных gcc-5.5 реализованы:

- `__(u)int128_t` — будут в lcc-1.24
- `_Decimal64`

Не все builtin gcc-5.5 есть (и с тем же поведением):

- `__builtin*_overflow` — будут в lcc-1.24
- `__builtin*_overflow_p`
- `__builtin_constant_p` — в особых случаях поведение отличается от gcc
- `__builtin_shuffle`
- ...

Отладка: `-O0 -g0, -fcontrol-spec`



# Проблемы компилятора

Не все типы данных gcc-5.5 реализованы:

- `__(u)int128_t` — будут в lcc-1.24
- `_Decimal64`

Не все builtin gcc-5.5 есть (и с тем же поведением):

- `__builtin*_overflow` — будут в lcc-1.24
- `__builtin*_overflow_p`
- `__builtin_constant_p` — в особых случаях поведение отличается от gcc
- `__builtin_shuffle`
- ...

Отладка: `-O0 -g0, -fcontrol-spec`



Не хватает более «редких» языков:

- Go
- Rust
- Haskell

Что можно сделать?

- бэкенд к LLVM
- ждём :)
- попутно решаются проблемы с «нестандартностью» компилятора



- Своя таблица системных вызовов (syscall)
- Своя таблица ioctl
- Иное устройство стека:
  - Раздельные стеки данных, функций, аргументов функций
  - Требуется особая работа со стеком (сборщики мусора и т.п.)
  - Требуется особая обработка работы с переключением контекстов: `makecontext_e2k()`, `freecontext_e2k()` и т.п.



## МЦСТ:

- ⇔ Обмен патчами
- ⇐ Новые версии
- ⇒ Сообщения об ошибках и проблемах

## Апстримы СПО:

- Можем отправлять только собственные патчи
- Кто-то принимает, кто-то нет
- Часть исправлений касается всех архитектур, но всплывает на e2k
- Зачем:
  - чтоб самим не тащить
  - чтоб СПО развивать



- Имеется полноценный репозиторий и инфраструктура сборки на e2k
- Эльбрусы самодостаточны для сборки дистрибутивов
- У архитектуры есть свои особенности
- Разработчики, пишите переносимый, стандартный код

Спасибо за внимание!

