

# Опыт поддержки догоняющей сборки Сизифа

Иван Мельников <[iv@altlinux.org](mailto:iv@altlinux.org)>

Сентябрь 2023г.

# Обо мне

Ivan A. Melnikov <[iv@altlinux.org](mailto:iv@altlinux.org)>

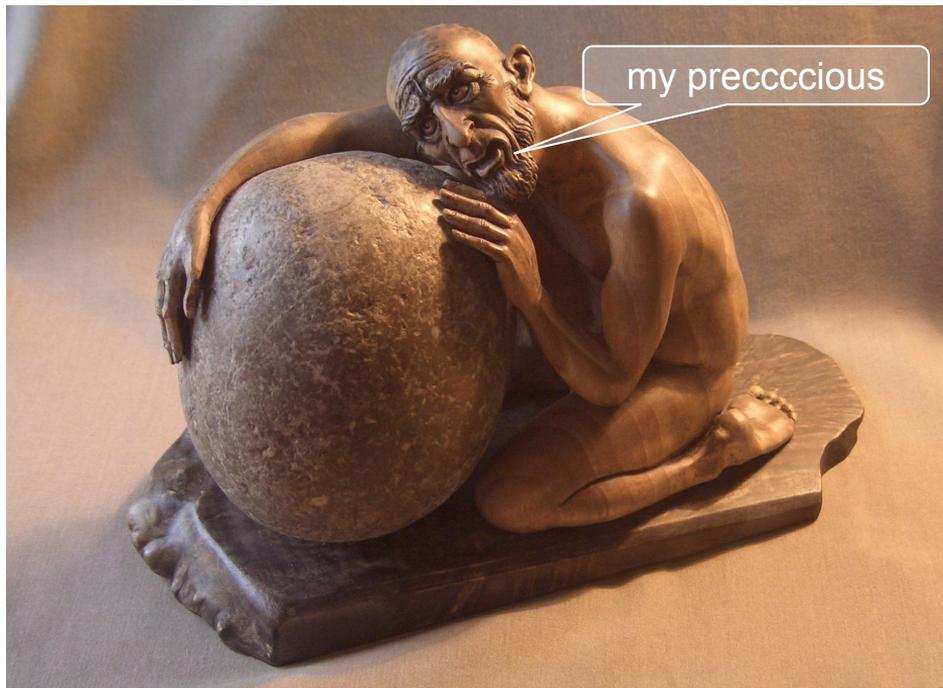
- в ALT Linux Team с 2008г.
- boost, krb5, ...

Иван Мельников <[iv@basealt.ru](mailto:iv@basealt.ru)>

- в ООО «Базальт СПО» с 2018г
- мейнтенейр портов Сизифа на mipsel и riscv64
- участвую в работе над недавно созданным портом на архитектуру loongarch64 



Это доклад в основном про sisyphus\_riscv64



Это доклад в основном про sisyphus\_riscv64



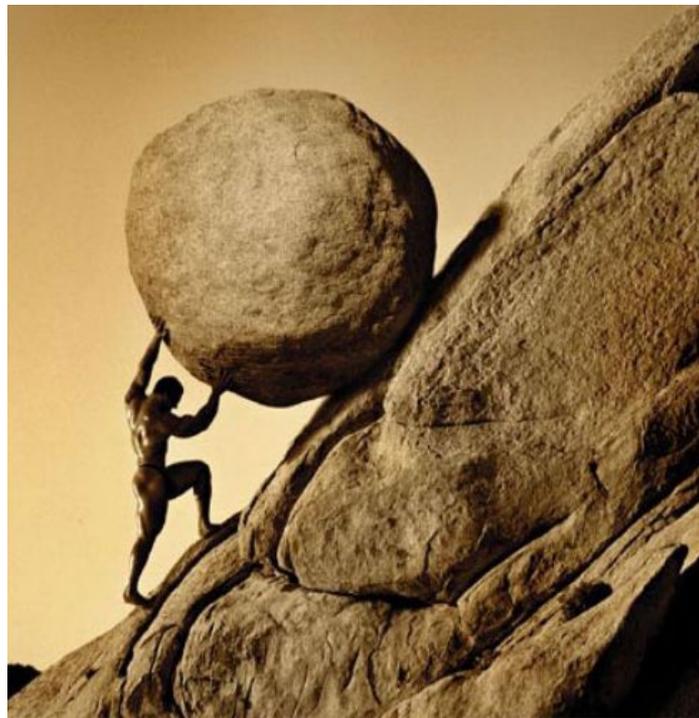
и немного про loongarch64 

# Основные архитектуры (ISA) в Сизифе

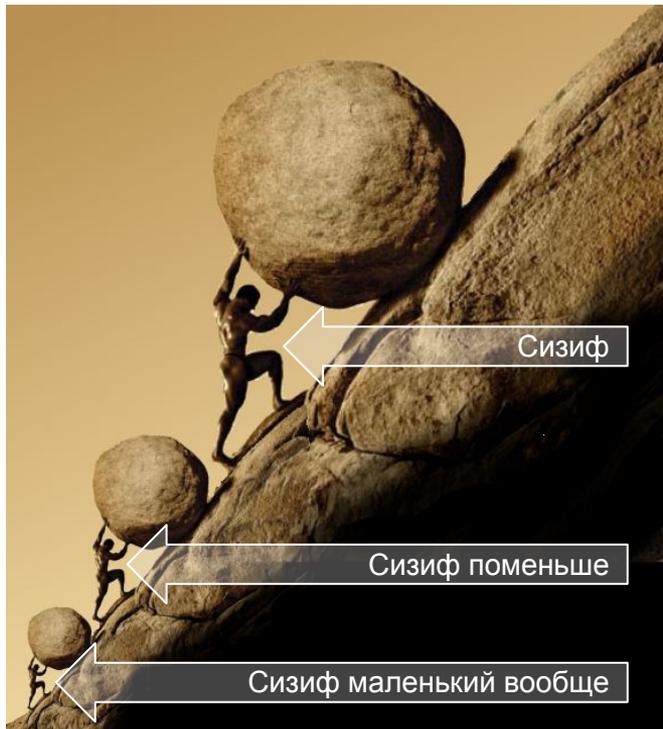
- мейнтейнеры пакетов создают задания на сборку/удаление пакетов
- задания обрабатывается синхронно на всех основных архитектурах
- если что-то пошло не так на одной архитектуре, надо перезапускать задачу для всех архитектур

Основные архитектуры это:

- x86\_64, i586
- aarch64, armh
- ppc64le



# «Догоняющие» архитектуры



Задания собираются:

- отдельно, после прохождения задания в основной Сизиф
- не блокируя задания для основных архитектур
- для каждой догоняющей архитектуры по-своему

Это:

- эльбрусы
- mipsel
- riscv64
- loongarch64 

# Догоняющая сборка подробнее

- ✓ Задание проходит в основной Сизиф
- ✓ Специальный робот копирует его в догоняющий girar (свой для каждого семейства архитектур)
- ✓ Специальный человек принимает какие-то из созданных роботом заданий или делает свои.
- ✓ Задания собираются
  - если не собираются, мейнтейнер порта с этим что-то делает
- ✓ Результаты сборки публикуются (тоже отдельно) (<http://ftp.altlinux.org/pub/distributions/ALTLinux/ports/...>)

# Результаты сборки

## Отдельный репозиторий

- независимый
- замкнутый (no unmets)
- *собственный poarch*
- публикуется ежедневно



# Подтверждаем задания

- не всегда сразу, часто в тот же день
- в основном – обновление уже собранных пакетов
  - для этого есть специальный инструмент, показывающий задачи, требующие внимания
- ЧТО-ТО НОВОЕ:
  - если нужно для обновления чего-то уже собранного
  - если есть цель
  - если попросят
  - если вдруг захочется
  - если в Сизифе появился новый интересный пакет

```
38106 [2023-09-16 15:29:39] NEW try=0 recycler test-only
      100 srpm firefox-unmet-holder-117.0.1-alt2.src.rpm RED
      200 srpm firefox-117.0.1-alt2.src.rpm YELLOW

38184 [2023-09-19 15:01:47] FAILED try=2 recycler
      100 srpm flite-2.2-alt2.src.rpm YELLOW

38218 [2023-09-19 21:29:03] NEW try=0 recycler test-only
      100 srpm gzip-1.13-alt2.src.rpm YELLOW

38219 [2023-09-19 22:29:06] NEW try=0 recycler test-only
      100 srpm openssl3-3.1.3-alt1.src.rpm ORANGE
      200 srpm ntp-4.2.8p15-alt4.src.rpm GREEN

38236 [2023-09-21 12:36:24] BUILDING try=4 recycler
      100 srpm gsettings-desktop-schemas-45.0-alt1.src.rpm ORANGE
      140 srpm icon-theme-awdaita-45.0-alt1.src.rpm ORANGE
      200 ("userid": "recycler")
      300 srpm libiso9660-1.8.0-alt1.src.rpm ORANGE
```

# Почему догоняющая сборка?

- Производительность и доступность оборудования
- Готовность репозитория



# Почему догоняющая сборка?

- Производительность и доступность оборудования
- Готовность репозитория
- Это Эльбрус



# Доступность оборудования

Не всегда нужный сервер можно купить за деньги.

- **loongarch64**: у нас есть достойные сервера (32 ядра, 128 Gb RAM); этого хватает для догоняющей сборки, но для основной нужно БОЛЬШЕ СЕРВЕРОВ
- **riscv64**: производительное оборудование появилось не так давно и пока не особо продаётся, так что собираем на HiFive Unmatched (4 in-order ядра, 16 Gb RAM)

```
top - 17:32:53 up 30 days, 11:22, 2 users, load average: 2.08, 2.02, 1.89
Tasks: 370 total, 3 running, 367 sleeping, 0 stopped, 0 zombie
%Cpu0 :  0,0/0,0  0t
%Cpu1 :  0,0/0,0  0t
%Cpu2 : 29,6/0,3 30t ██████████
%Cpu3 :  0,3/0,3  1t
%Cpu4 :  8,0/1,3  9t █████
%Cpu5 :  0,3/0,3  1t
%Cpu6 :  0,0/0,0  0t
%Cpu7 : 57,8/1,3 59t ██████████
%Cpu8 :  0,3/0,0  0t
%Cpu9 :  0,0/0,0  0t
%CPU10 : 0,0/0,0 0t
%CPU11 : 0,0/0,0 0t
%CPU12 : 0,0/0,0 0t
%CPU13 : 0,0/0,0 0t
%CPU14 : 0,0/0,0 0t
%CPU15 : 0,0/0,3 0t
%CPU16 : 0,0/0,0 0t
%CPU17 : 0,0/0,0 0t
%CPU18 : 0,0/0,0 0t
%CPU19 : 0,0/0,0 0t
%CPU20 : 0,0/0,0 0t
%CPU21 : 0,0/0,0 0t
%CPU22 : 0,0/0,0 0t
%CPU23 : 0,0/0,0 0t
%CPU24 : 0,0/0,0 0t
%CPU25 : 100,0/0,0 100t ██████████
%CPU26 : 0,0/0,0 0t
%CPU27 : 0,0/0,0 0t
%CPU28 : 0,0/0,0 0t
%CPU29 : 0,0/0,0 0t
%CPU30 : 0,0/0,0 0t
%CPU31 : 0,0/0,0 0t
MIB Mem : 130377,7 total, 38716,1 free, 5039,9 used, 86621,7 buff/cache
MIB Swap:  0,0 total,  0,0 free,  0,0 used, 109673,7 avail Mem

  PID USER      PR  NI  VIRT  RES  SHR  S  %CPU  %MEM    TIME+  COMMAND
1615368 loongar+  36  16 1824,4g 256000 120832 R 100,0  0,2  0:09.88 /usr/lib64/gh
1615465 loongar+  36  16 1824,4g 233472 123904 R 29,8  0,2  0:03.72 /usr/lib64/gh
  15 root       20  0      0      0      0  I  0,3  0,0 26:32.33 [rcu_preempt]
1615361 root       20  0      0      0      0  R  0,3  0,0  0:00.26 top
```

# Производительность оборудования

Догоняющая сборка позволяет существовать и развиваться портам даже если с мощных сборочных серверов под эту архитектуру нет.

|                      | <b>x86_64</b> | <b>armh</b> | <b>loongarch64</b> | <b>riscv64</b> |
|----------------------|---------------|-------------|--------------------|----------------|
| <b>php8.1</b>        | 14:04         | 40:09       | 20:48              | 4:02:16        |
| <b>qt5-base</b>      | 21:10         | 58:05       | 57:06              | 6:02:56        |
| <b>python3</b>       | 22:33         | 32:47       | 55:31              | 16:38:36       |
| <b>libwebkitgtk4</b> | 21:46         | 55:36       | 44:48              | 19:56:52       |
| <b>gcc13</b>         | 2:31:04       | 5:08:41     | 3:14:53            | 4:03:25:19*    |

\* Обычно gcc под riscv64 собирается без profiledbootstrap, и поэтому значительно быстрее. Это был эксперимент.

# Готовность репозитория: Bootstrap

Bootstrap Сизифа на новую архитектуру:

- базовая среда
- rpm-build
- hasher (изолированное окружение)
- догоняющая сборка
- ???\*
- архитектура становится основной



\* Так далеко я ещё не заходил

# Готовность репозитория

Не только собрать вообще всё, но и:

- Забутсрапить всё, что нужно забутсрапить
- Оптимизировать всё, что должно быть оптимизировано
  - jdk, ghc, ocaml, luajit...
- Передать все исправления в основной Сизиф

Тем временем, порт уже полезен, например

- Можно делать образы
- Можно работать над поддержкой оборудования

# Развитие порта

Двухминутка обожания

- ExclusiveArch vs ExcludeArch
- циклы зависимостей
- FTBFS

# Утрируем

## Когда взялся за новый порт Сизифа

### Ожидание

- отлаживаешь кодогенераторы
- портируешь код с ассемблерными вставками
- правишь binutils



### Реальность

- добавляешь в сотню пакетов ручку `with_bootstrap`
- исправляешь тесты какому-то питоновскому коду
  - апстримными патчами
  - на `x86_64`



# Обновления, которые всё ломают

Можно придерживаться!

- особенно на раннем этапе развития порта
- особенно если сложно полноценно протестировать

Однако, важно помнить, что

- порт – это не форк;
- решать проблемы основного Сизифа важнее и интереснее.



докладчик  
не придумал  
шутку для  
последнего  
слайда

- <https://www.altlinux.org/Ports/riscv64>
- <https://www.altlinux.org/Ports/loongarch64>