

Опыт разработки нового Альтератора

Михаил Чернигин,
Младший разработчик, ООО «Базальт СПО»



Немного о себе

- Работаю в Базальте 1,5 года как младший разработчик
- Работаю над Альтератором примерно 1 год
- Занимаюсь разработкой приложений для нового Альтератора

**Что и зачем мы
разрабатываем?**

01

Альтератор — это инсталлятор и конфигуратор операционных систем «Альт».

Первоначальная его реализация предполагает возможность написания модулей на языке Scheme для фронтендов и на языке Bash для бэкендов.

Что не так со старым Альтератором?

- Scheme с Qt усложняет написание фронтов
- Запуск в графике доступен только `root`
- Отсутствие API модулей для приложений

Цели разработки нового Альтератора

- Спецификация интерфейсов бэкендов для создания фронтендов
- Обеспечение свободы в выборе технологий для реализации фронтендов и бекэндов
- Предоставление разработчикам фреймворка для создания собственных приложений
- Обобщение механизмов локального и удалённого конфигурирования системы

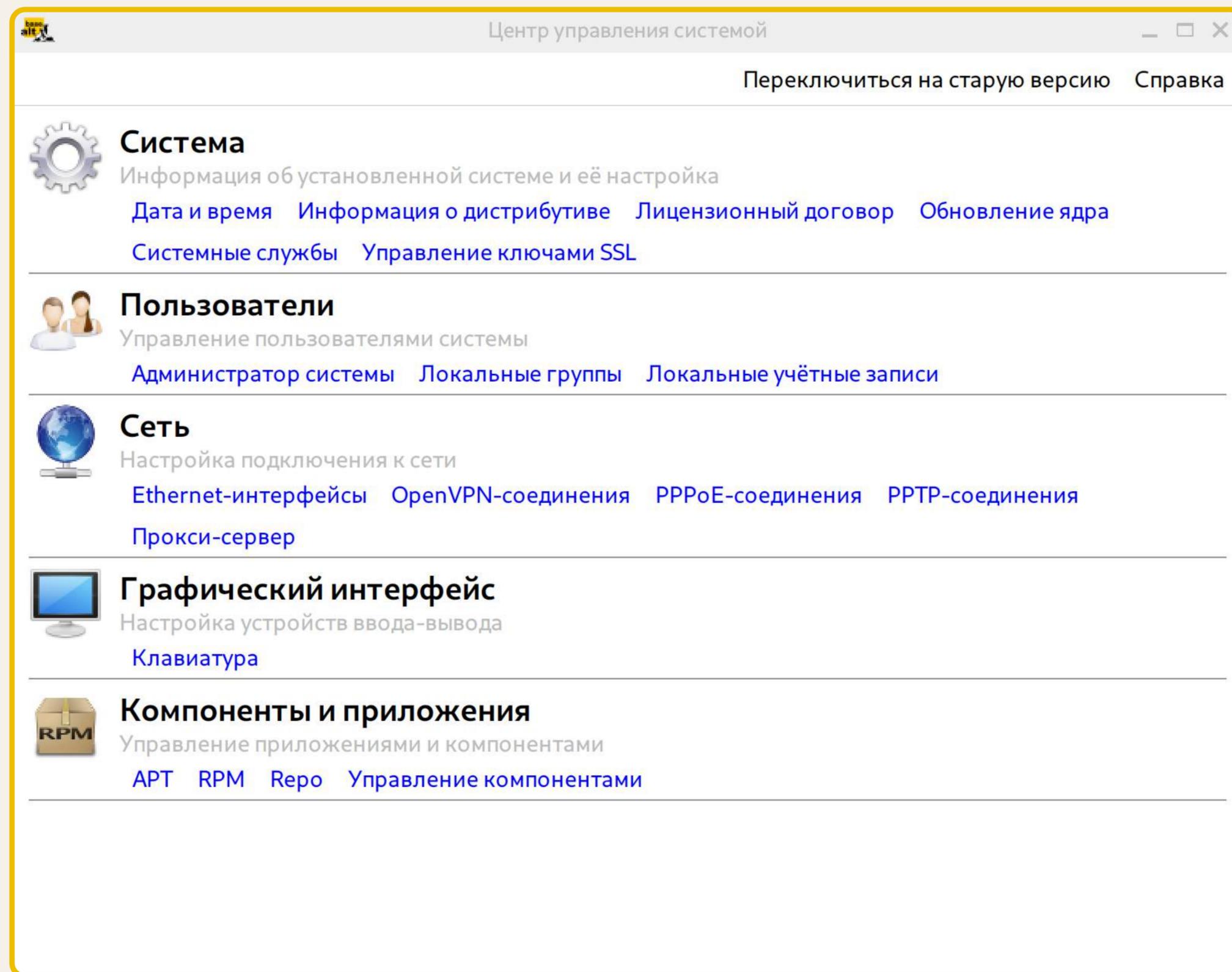
**Что мы уже
сделали?**

02

Альтератор Браузер

Центр управления системой.
Служит для отображения
модулей старого Альтератора и
приложений нового.

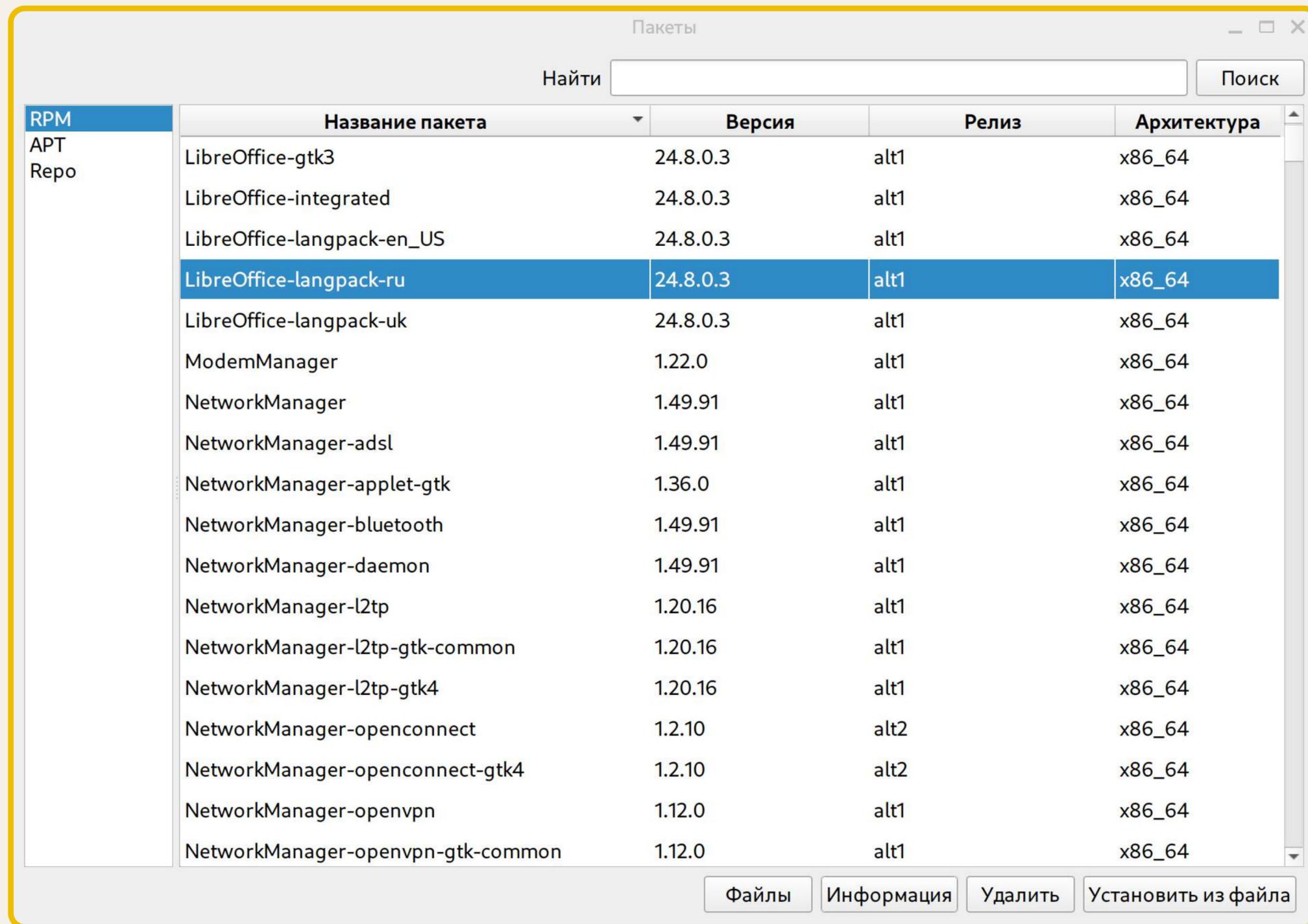
Доступен в Сизифе как пакет
`alterator-browser`



Управление пакетами

Служит для просмотра и получения информации об установленных и доступных пакетах, а также их установки и удаления.

Доступен в Сизифе как пакет
`alterator-application-packages`

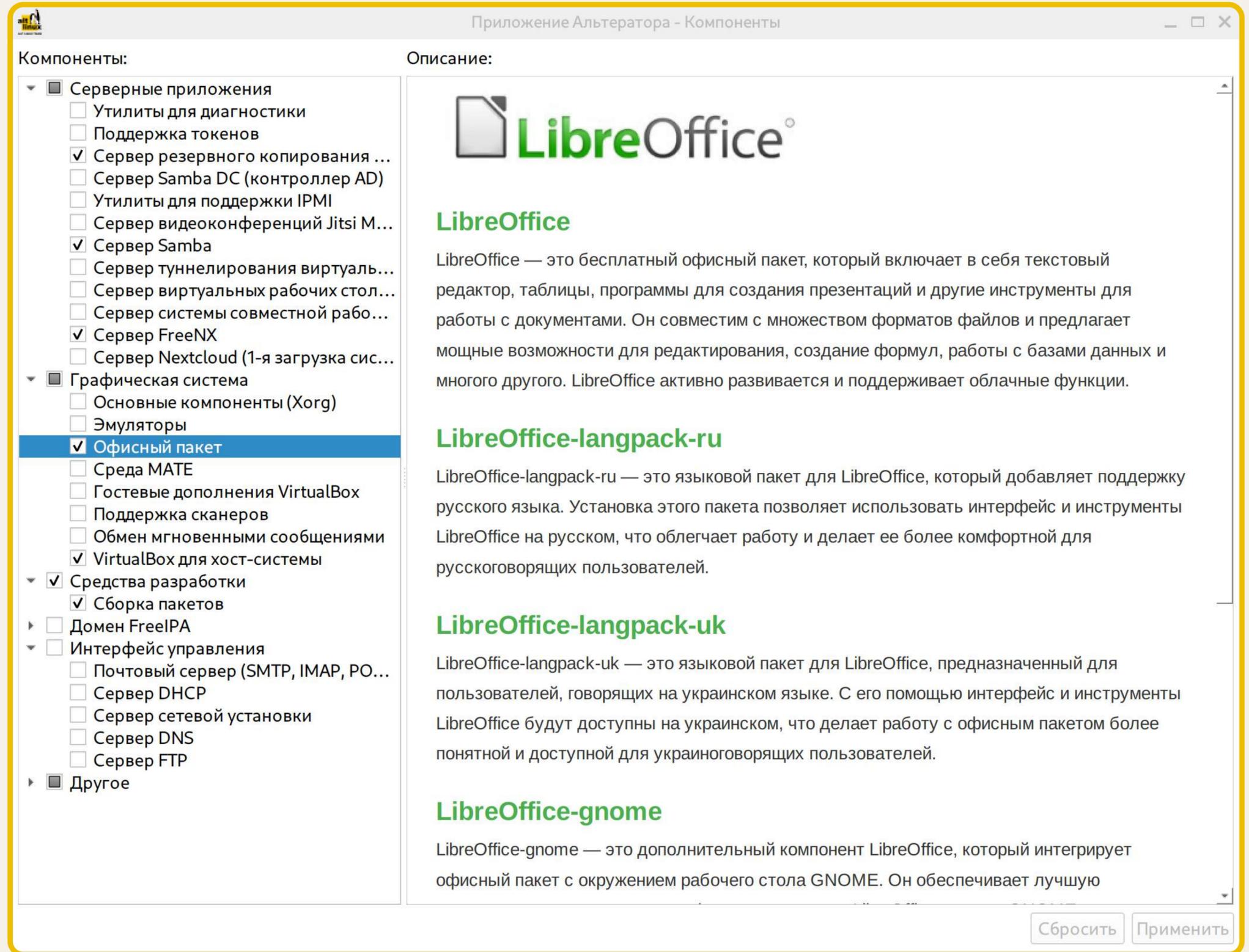


RPM	Название пакета	Версия	Релиз	Архитектура
APT	LibreOffice-gtk3	24.8.0.3	alt1	x86_64
Repo	LibreOffice-integrated	24.8.0.3	alt1	x86_64
	LibreOffice-langpack-en_US	24.8.0.3	alt1	x86_64
	LibreOffice-langpack-ru	24.8.0.3	alt1	x86_64
	LibreOffice-langpack-uk	24.8.0.3	alt1	x86_64
	ModemManager	1.22.0	alt1	x86_64
	NetworkManager	1.49.91	alt1	x86_64
	NetworkManager-adsl	1.49.91	alt1	x86_64
	NetworkManager-applet-gtk	1.36.0	alt1	x86_64
	NetworkManager-bluetooth	1.49.91	alt1	x86_64
	NetworkManager-daemon	1.49.91	alt1	x86_64
	NetworkManager-l2tp	1.20.16	alt1	x86_64
	NetworkManager-l2tp-gtk-common	1.20.16	alt1	x86_64
	NetworkManager-l2tp-gtk4	1.20.16	alt1	x86_64
	NetworkManager-openconnect	1.2.10	alt2	x86_64
	NetworkManager-openconnect-gtk4	1.2.10	alt2	x86_64
	NetworkManager-openvpn	1.12.0	alt1	x86_64
	NetworkManager-openvpn-gtk-common	1.12.0	alt1	x86_64

Управление компонентами

Служит для просмотра доступных компонент, а также для их установки и удаления.

Доступен в Сизифе как пакет `alterator-application-components`



Приложение Альтератора - Компоненты

Компоненты:

- Серверные приложения
 - Утилиты для диагностики
 - Поддержка токенов
 - Сервер резервного копирования ...
 - Сервер Samba DC (контроллер AD)
 - Утилиты для поддержки IPMI
 - Сервер видеоконференций Jitsi M...
 - Сервер Samba
 - Сервер туннелирования виртуаль...
 - Сервер виртуальных рабочих стол...
 - Сервер системы совместной рабо...
 - Сервер FreeNX
 - Сервер Nextcloud (1-я загрузка сис...
- Графическая система
 - Основные компоненты (Xorg)
 - Эмуляторы
 - Офисный пакет**
 - Среда MATE
 - Гостевые дополнения VirtualBox
 - Поддержка сканеров
 - Обмен мгновенными сообщениями
 - VirtualBox для хост-системы
- Средства разработки
 - Сборка пакетов
- Домен FreeIPA
- Интерфейс управления
 - Почтовый сервер (SMTP, IMAP, PO...
 - Сервер DHCP
 - Сервер сетевой установки
 - Сервер DNS
 - Сервер FTP
- Другое

Описание:



LibreOffice

LibreOffice — это бесплатный офисный пакет, который включает в себя текстовый редактор, таблицы, программы для создания презентаций и другие инструменты для работы с документами. Он совместим с множеством форматов файлов и предлагает мощные возможности для редактирования, создание формул, работы с базами данных и многого другого. LibreOffice активно развивается и поддерживает облачные функции.

LibreOffice-langpack-ru

LibreOffice-langpack-ru — это языковой пакет для LibreOffice, который добавляет поддержку русского языка. Установка этого пакета позволяет использовать интерфейс и инструменты LibreOffice на русском, что облегчает работу и делает ее более комфортной для русскоговорящих пользователей.

LibreOffice-langpack-uk

LibreOffice-langpack-uk — это языковой пакет для LibreOffice, предназначенный для пользователей, говорящих на украинском языке. С его помощью интерфейс и инструменты LibreOffice будут доступны на украинском, что делает работу с офисным пакетом более понятной и доступной для украиноговорящих пользователей.

LibreOffice-gnome

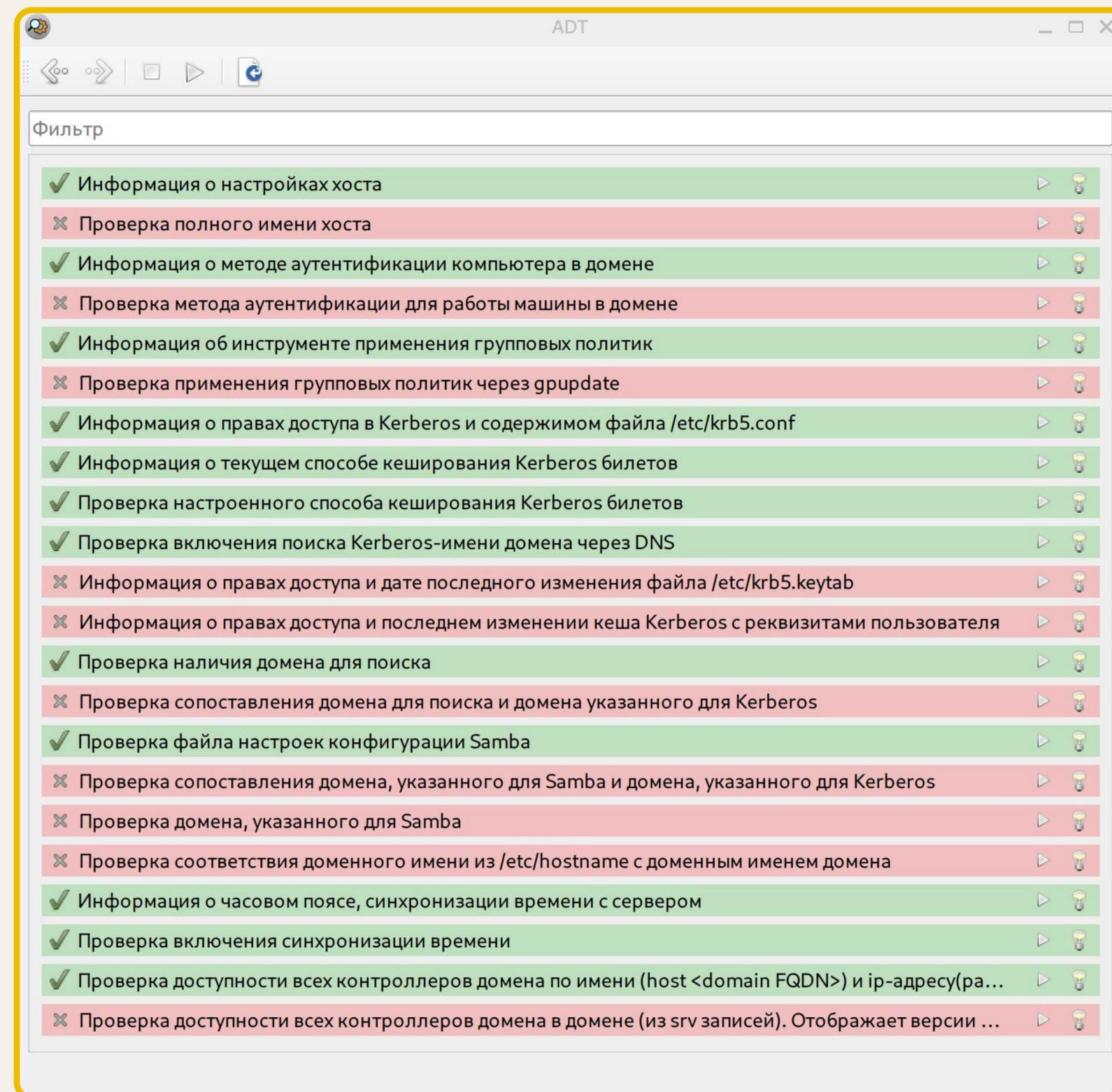
LibreOffice-gnome — это дополнительный компонент LibreOffice, который интегрирует офисный пакет с окружением рабочего стола GNOME. Он обеспечивает лучшую

Сбросить Применить

Инструменты диагностики

Служит для запуска тестов на системе и просмотра результатов их выполнения.

Доступен в Сизифе как пакет
adt



alteratorctl

Служит для управления приложениями Альтератора в формате cli-интерфэйса.

Доступен в Сизифе как пакет
alteratorctl

```
$ alteratorctl --help
```

Справка:

```
alteratorctl (-h | --help | -m | --modules | [-v | --verbose]  
<имя_модуля> <подкоманда> [<аргумент>])
```

Опции:

-h, --help	Вывести справку
-v, --verbose	Режим подробного вывода информации
-m, --modules	Отобразить зарегистрированные модули

Поддерживаемые модули:

```
manager  
packages  
components  
editions  
diag
```

Для получения дополнительной информации по конкретному модулю и его подкомандам, используйте:

```
alteratorctl <имя_модуля> help
```

alteratorctl

Служит для управления приложениями Альтератора в формате cli-интерфэйса.

Доступен в Сизифе как пакет
alteratorctl

```
$ alteratorctl components help
```

Справка:

```
alteratorctl components подкоманда [<имя_компоненты>]
```

Подкоманды:

```
list
```

Вывести список компонентов

```
info [<имя_компоненты>]
```

Вывести информацию о компонентах

```
install [<имя_компоненты>]
```

Установить компоненты из списка

```
remove [<имя_компоненты>]
```

Удалить компоненты из списка

```
description [<имя_компоненты>]
```

Вывести описание компонентов

```
status [<имя_компоненты>]
```

Статус установки компонентов

```
help
```

Вызов справки модуля components

alteratorctl

Служит для управления приложениями Альтератора в формате cli-интерфэйса.

Доступен в Сизифе как пакет
alteratorctl

```
$ alteratorctl diag help
```

Справка:

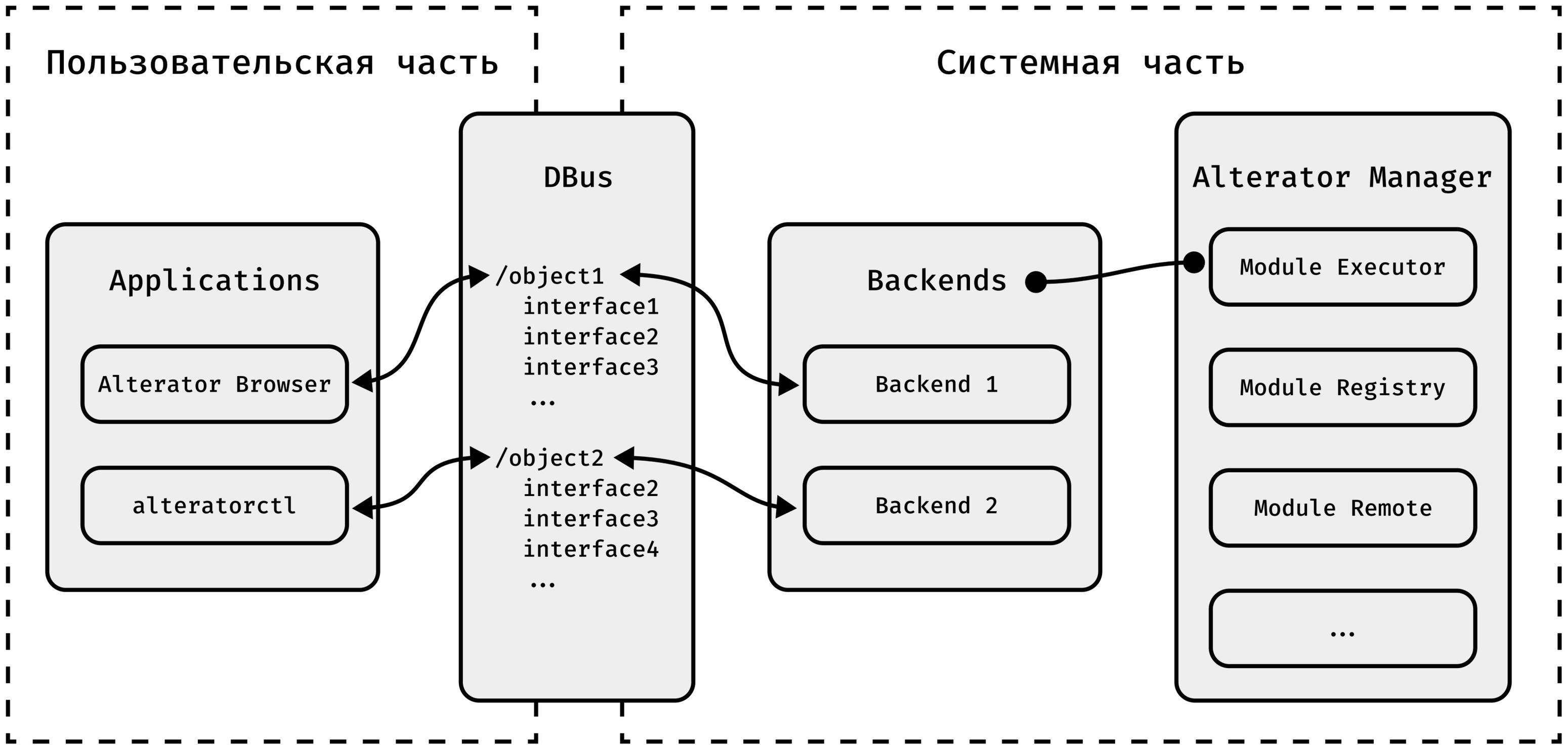
```
alteratorctl diag подкоманда [<путь_к_инструменту> [<тест> | <файл>]]
```

Подкоманды:

info [путь]	Вывести информацию о выбранном инструменте
list-tests <путь>	Вывести список тестов инструмента
list-all	Вывести список диагностических инструментов
report [путь] [файл]	Сохранить отчёт в файл с суффиксом
run-tool-test [путь] [тест]	Запустить выбранный тест
run-all [путь]	Запустить все тесты в выбранном инструменте
help	Вывод справки модуля diag

**Как это
работает?**

03



Как создать своё приложение Альтератора?

1. Реализовать интерфейс на DBus в объекте ``/ru/basealt/alterator/...``
2. Реализовать приложения, которое использует этот интерфейс

1. Реализация бэкенда

Определение объекта apt с методом Update с помощью backend файла и Alterator Module Executor

```
[Alterator Entry]
Type = Backend
Module = executor
Name = apt
Interface = apt1
```

```
[Info]
execute = cat /usr/share/alterator/objects/apt.object
stdout_bytes = enabled
```

```
[Update]
execute = apt-get update
stderr_strings = enabled
```

Файл /usr/share/alterator/backends/apt.backend

1. Реализация бэкенда

```
<node name="/">
  <interface name="ru.basealt.alterator.ap1">
    <method name="Info">
      <arg type="ay" name="stdout_bytes" direction="out"></arg>
      <arg type="i" name="response" direction="out"></arg>
    </method>
    <method name="Update">
      <arg type="as" name="stderr_strings" direction="out"></arg>
      <arg type="i" name="response" direction="out"></arg>
    </method>
  </interface>
</node>
```

Описание интерфейса ap1
для валидации

Файл /usr/share/dbus-1/interfaces/ru.basealt.alterator.ap1.xml

2. Реализация фронтенда

Фронтенд можно реализовать в виде графического приложения на любом языке и фреймворке, которое будет обращаться к предоставляемому бекэндом интерфейсу на DBus.

Чтобы реализованное приложение появилось в Alterator Browser, ему так же необходимо создать объект на DBus с интерфейсом `application1`.

2. Реализация фронтенда

Определение объекта с
интрефейсом `application1`
для отображения в
Альтератор Браузере

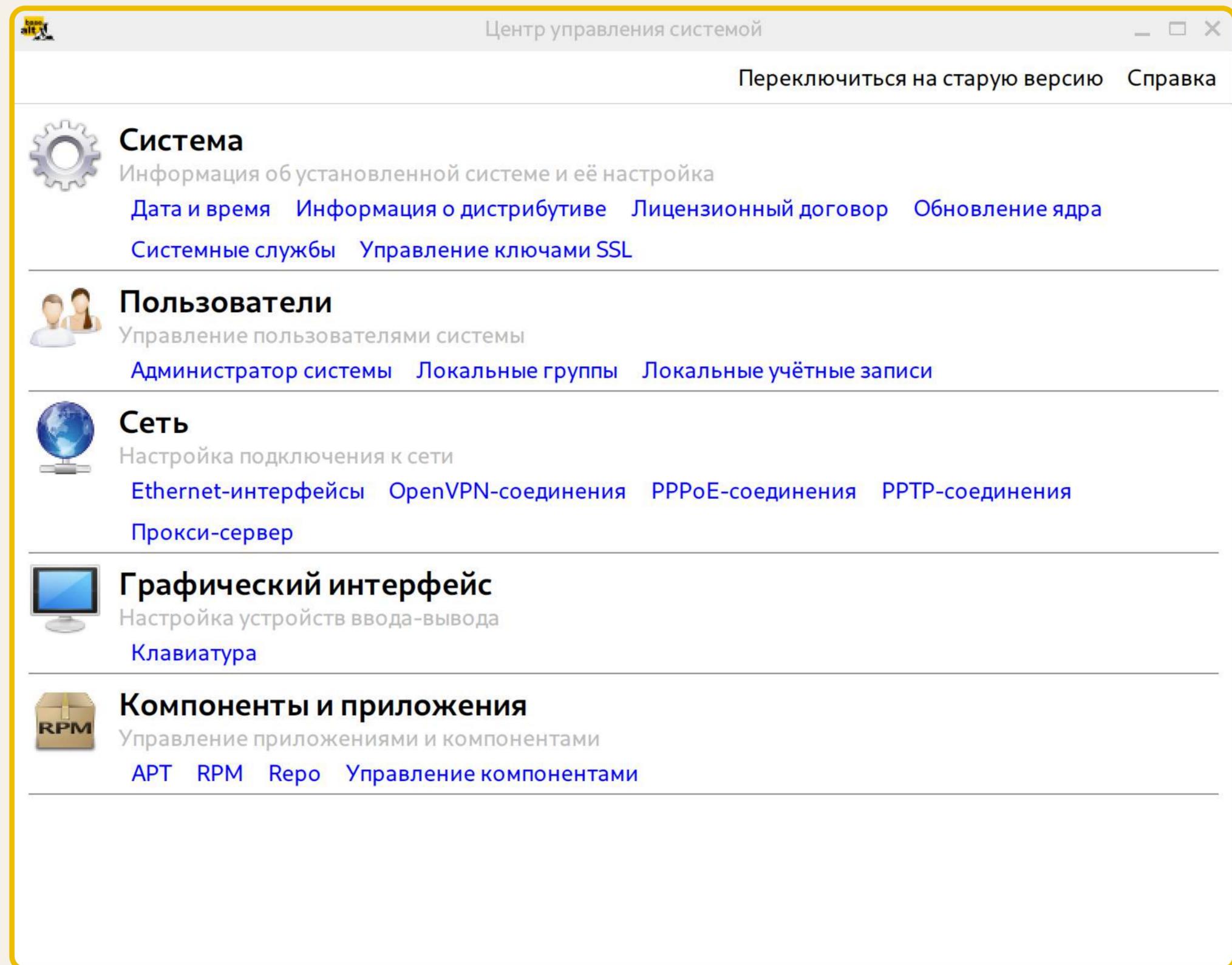
```
[Alterator Entry]
Type = Backend
Module = executor
Name = packages
Interface = application1
```

```
[Info]
execute = cat /usr/share/alterator/applications/packages.application
stdout_bytes = enabled
```

Файл `/usr/share/alterator/backends/application-packages.backend`

Готово!

Теперь ваше приложение
доступно из
Альтератор Браузера.



Наши наблюдения и планы

04

Всё хорошо!

Всё хорошо! Но...

Много кода для работы с Dbus в приложениях

Все фронтенды являются независимыми приложениями



.Они требуют повторной реализации обращений и обработки сообщений через Dbus и построения объектов в рамках ООП.

Что с ЭТИМ делать?

На данный момент решением данной проблемы видится создание библиотеки, условно называемой `libalterator`, предоставляющей инструменты для работы с `alterator-manager` на шине Dbus, отправки сообщений и парсинга ответов.

Какие-то ещё ближайшие планы

- Приложение для управления редакциями
- Приложение для получения информации о системе
- Реализация модуля Remote, чтобы вся эта красота могла работать и с удалённого хоста

Спасибо за внимание!

Теперь время для вопросов :)