

Утилита диагностики системы ADT

Алексей Сапрунов

E-mail: sa@basealt.ru





Что такое Alt Diagnostic Tool (ADT):

ADT — утилита для запуска тестов, реализующих интерфейс `ru.basealt.alterator.diag1` в графическом интерфейсе и с помощью командной строки.

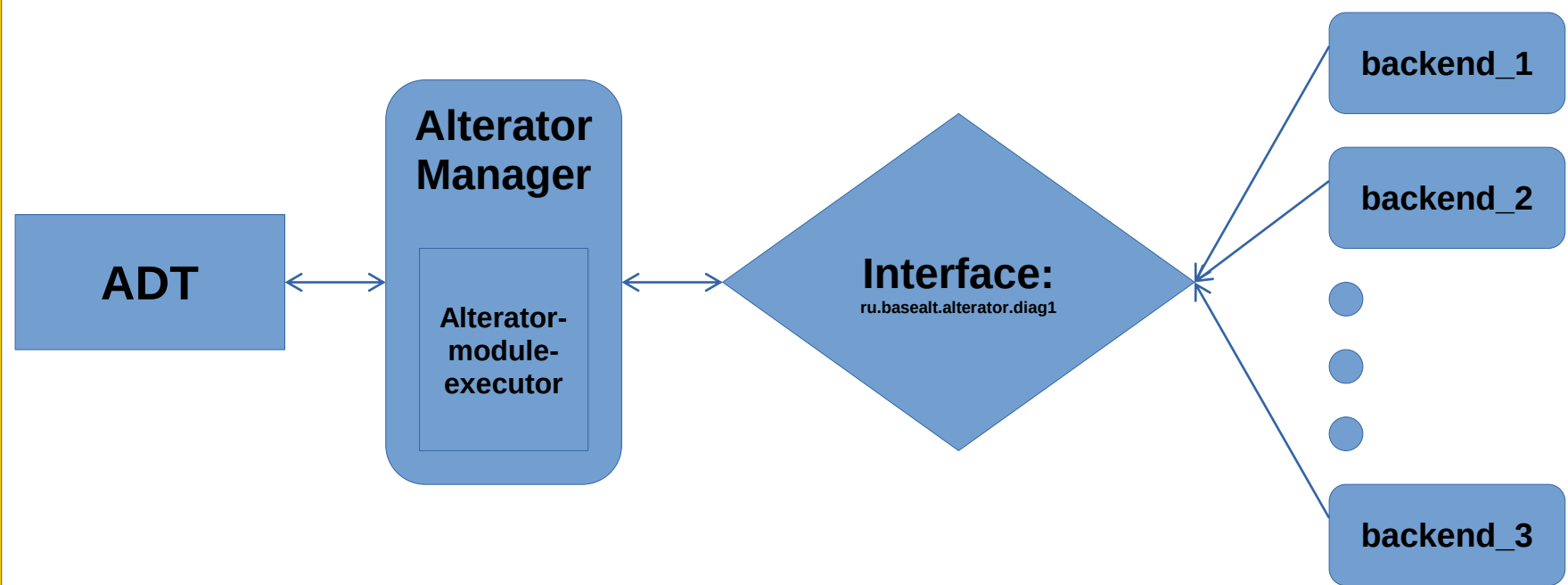


Цели создания ADT:

- создание простого инструмента для диагностики проблем и сбоев ПО
- автоматизация сборов журналов диагностики
- диагностика в закрытых контурах
- возможность проведения диагностики пользователем с последующей отправкой результатов тестирования специалисту
- возможность самостоятельного написания тестов для решения типичных проблем



Как работает ADT:





Как создать инструмент для ADT

- 1) Создать тест (например, bash-скрипт)
- 2) Создать .backend файл для alterator-module-executor, реализующий интерфейс ru.basealt.alterator.diag1
- 3) Создать .desktop файл с текстом и переводами



Требования к инструменту тестирования:

- 1. возвращение названий тестов в виде строк**
- 2. запуск отдельного теста при передаче в качестве параметра названия теста(полученного из списка в пункте 1)**
- 3. в случае успешности теста, код возврата должен быть 0. В случае выявления ошибок, код возврата должен быть не равным нулю.**
- 4. Результаты теста должны быть выведены в поток стандартного вывода.**

Полное описание есть в репозитории:

<https://github.com/AlexSP0/alt-diagnostic-tool/>



Требования к `.backend` файлу:

`.backend` файл описывает, как будет отображаться объект инструмента на системной шине D-Bus. Он используется модулем `alterator-manager`, который называется `alterator-module-executor`.

Объект, который создается на шине D-Bus, должен реализовывать интерфейс `ru.basealt.alterator.diag1`.

Полное описание есть в репозитории: <https://github.com/AlexSP0/alt-diagnostic-tool/>



Пример .backend файла из инструмента domain-diag:

[Manager]

```
module_name = executor  
node_name = domain_diag  
interface_name = diag1
```

[info]

```
execute = cat /usr/share/adtttools/diag2.desktop  
stdout_bytes = enabled  
stdout_byte_limit = 200000
```

[run]

```
execute = domain-diag {test-name}  
stdout_signal_name = diag1_stdout_signal  
stderr_signal_name = diag1_stderr_signal  
thread_limit = 3
```

[list]

```
execute = domain-diag -l  
stdout_strings = enabled  
stdout_strings_limit = 200000
```




Зачем нужен .desktop файл:

В .desktop файле содержится текстовое название, описание и переводы для инструмента и входящих в него тестов. Он необходим для правильного отображения названия и описания на нужном языке.



Пример .desktop файла:

[Desktop Entry]

name=domain_diag

name[ru_RU]=диагностика домена

name[en_US]=domain diagnostic tool

description=diag_1 description

description[ru_RU]=описание диагностического теста1

description[en_US]=description diag test1

icon=system-run

[check_hostnamectl]

name = check_hostnamectl

name[ru_RU]=тест check_hostnamectl

name[en_US]=test check_hostnamectl

description= desc check_hostnamectl

description[ru_RU]=описание check_hostnamectl

description[en_US]=description check_hostnamectl



Какие задачи можно решать с помощью ADT:

- 1. быстрое выявление и решение «типовых» проблем**
- 2. диагностика системы в закрытых контурах**
- 3. возможность проведения диагностики обычным пользователями в графическом интерфейсе с отправкой отчета специалисту**

Например, специалисты «Базальт СПО» планируют использовать ADT для диагностики проблем функционирования доменов, проблем применения групповых политик, а также для решения проблем в закрытых контурах. В настоящий момент разрабатывается инструмент domain-diag.



Вопросы