

SPD Root in docker

Дуров Андрей

За дополнительной информацией можно писать на:
russianrakkan@gmail.com

Данная инструкция рассчитана на установку и запуск SPDRoot в docker'е на нашей ферме МИФИ, доступы к которой можно получить на кафедре.

Но также можно всё установить и запустить локально – для этого нужно установить себе docker (<https://docs.docker.com/engine/install/>) и дальше следовать либо этой инструкции за исключением шагов по работе с фермой, либо более укороченной инструкции тут (подходит для MacOS, Linux; для Windows тоже можно запустить, но нужно будет разобраться с тем, как выводить графику из docker'а на экран):

https://git.jinr.ru/nica/spdroot/-/blob/feature/renewed_docker/docker/README.md?ref_type=heads

Setup (1)

1) Зайти на ферму МИФИ (вместо user использовать свой, выданный вам, логин ; после выполнения команды также будет предложено ввести пароль, который вам должны были выдать вместе с логином):

```
ssh -Y user@ui03.lxfarm.mephi.ru
```

2) Клонировем в домашний (или другой по вашему выбору) каталог репозиторий <https://git.jinr.ru/nica/spdroot.git> :

```
git clone https://git.jinr.ru/nica/spdroot.git
```

3) Заходим в клонированный репозиторий и переключаемся на ветку feature/renewed_docker:

```
cd spdroot
```

```
git switch feature/renewed_docker
```

Setup (2)

4) Создаем директорию для сохранения выходных файлов после рабочих сессий в исполнительном режиме (см. Execution, п.2):

```
mkdir ./output
```

5) Заходим в директорию `spdroot/docker` (при условии, что клонировали мы репозиторий в домашний каталог; в ином случае пишем полный путь до нашего скопированного репозитория `spdroot`):

```
cd ~/spdroot/docker
```

Setup (3)

6) Там, где мы заменили наши скрипты, настраиваем доступы на исполнение для файлов do.run и env.run:

```
chmod +x do.run env.run
```

7) Проверяем, что у вашего пользователя есть права на пользование докером - запускаем команду ниже и смотрим на вывод:

```
docker ps
```

8) Если возникла ошибка с доступом (<...>permission denied<...>), то нужно обратиться к Сергею Юрьевичу Смирнову для получения доступа к пользованию докером. Если ошибки нет, то будет виден список запущенных контейнеров (вероятно, он будет пустой)

9) Теперь всё готово для дальнейшей работы!

Execution (1)

Есть два способа работы с SPD Root в докере:

- 1) Интерактивный режим. Для этого находясь в всё в той же директории `spdroot/docker` (в рассматриваемом случае: `~/spdroot/docker/`) запускаем скрипт `“do.run”` с аргументом `“root”`:

```
./do.run root
```

Так будет создан и запущен контейнер, а в нём – SPD root. Вы сразу попадете в интерфейс ROOT’а.

ВАЖНО!!! После выхода из ROOT’а контейнер будет уничтожен (это сделано с целью экономии дискового пространства). Все новосозданные файлы за эту сессию будут безвозвратно потеряны. Поэтому этот режим работы подходит либо только для разового выполнения какого-то макроса с целью визуализации результата, либо для “прогулки” по TBrowser, либо для еще какой-то цели, не подразумевающей артефакты после рабочей сессии.

Execution (2)

2) Исполнительный режим. Для этого находясь в всё в той же директории `spdroot/docker` (в рассматриваемом случае: `~/spdroot/docker/`) запускаем скрипт “do.run” с аргументами “root” и “-x <path>/<macro>”:

```
./do_run root -x <path>/<macro>
```

Так будет создан запущен контейнер, а в нём будет отрабатывать макрос в SPD root . Результат работы будет зависеть от вашего макроса.

ВАЖНО!!! После выхода из ROOT’а контейнер будет уничтожен (это сделано с целью экономии дискового пространства). Если необходимо сохранить результаты работы макроса необходимо в нём же (в макросе) указать выходной файл, в который будут сохранены результаты работы макроса. Этот файл необходимо сохранять по адресу `/var/src/output/<file>`, так как именно директория `/var/src` внутри контейнера (кстати говоря, не есть то же самое, что и `/var/src` вашей машины) останется “живой” после уничтожения контейнера и будет она у вас локально на машине в директории `~/spdroot/docker/output/`, которая создавалась во время настройки работы (это если репозиторий изначально в Setup, п. 1 клонировался в домашнюю директорию; в ином случае - по другому, указанному вами пути (см. Setup, п.4)).

Additional information

- Для того, чтобы зайти в существующий (остановленный) контейнер, нужно посмотреть его имя при помощи “docker ps -a”, запустить его и зайти в него:

```
docker start <name>
```

```
docker exec --it <name> /bin/bash
```