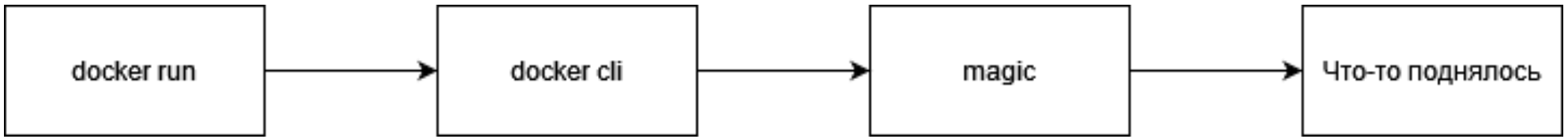
How docker works inside?

Выполнение команды

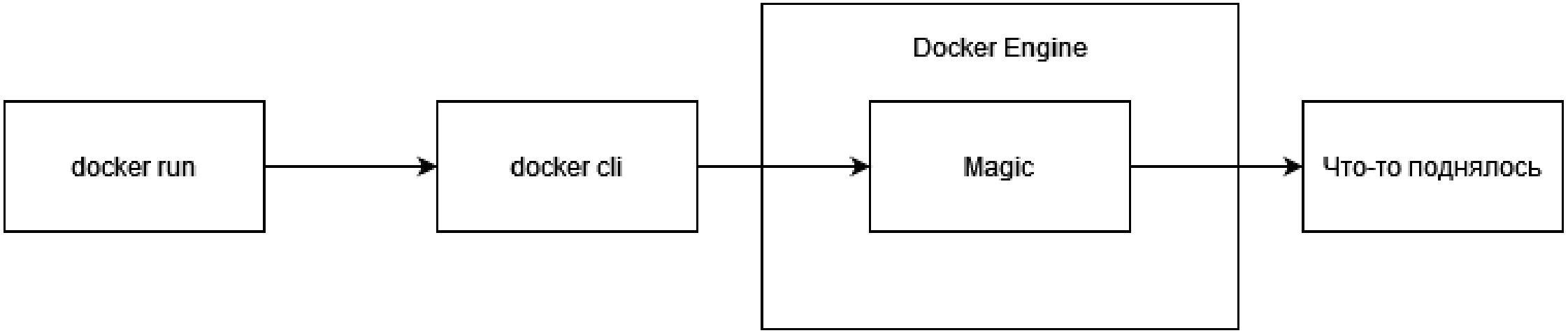
Docker cli

- Командная строка для связи с конечным пользователем
- Реинтерпретирует команды, вводимым пользователем и отправляет на docker engine



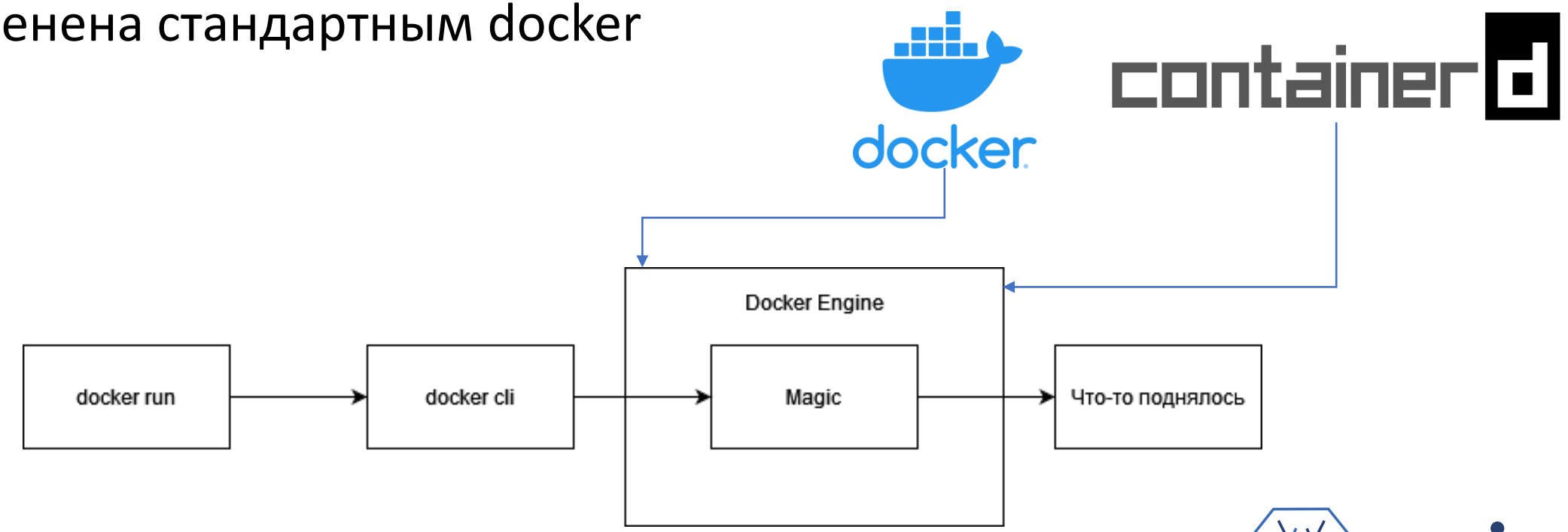
Docker Engine

- Утиллита, которая имеет спецификацию для работы с контейнерами
- Имеет спецификацию (REST API, docker engine слушает порт)



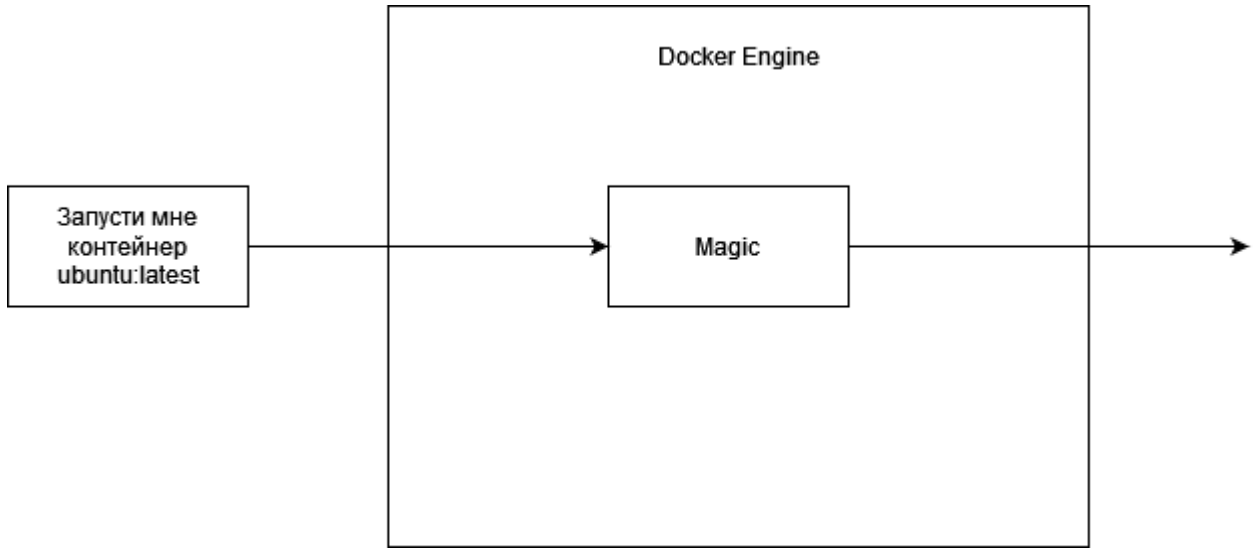
Аналоги docker engine

- Стандарт контейнеров Docker де-факто стандарт в индустрии
- Если программа выполняет спецификацию, она может быть подменена стандартным docker



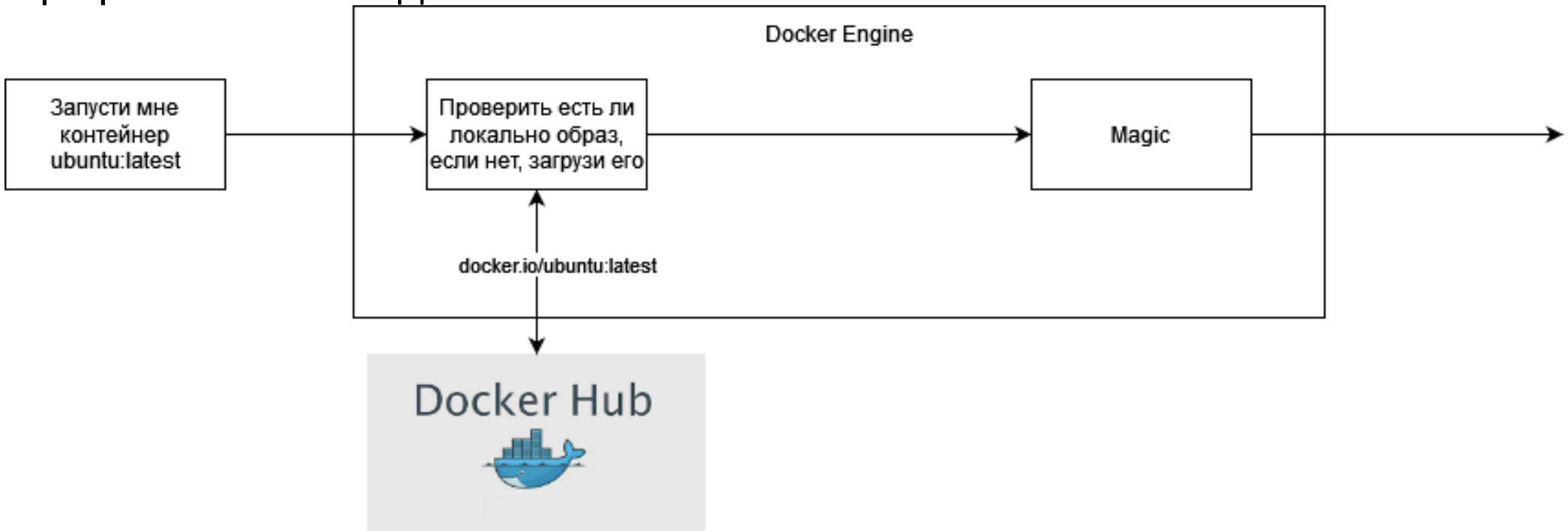
Как работает docker engine?

- Docker daemon, жизненным циклом контейнеров



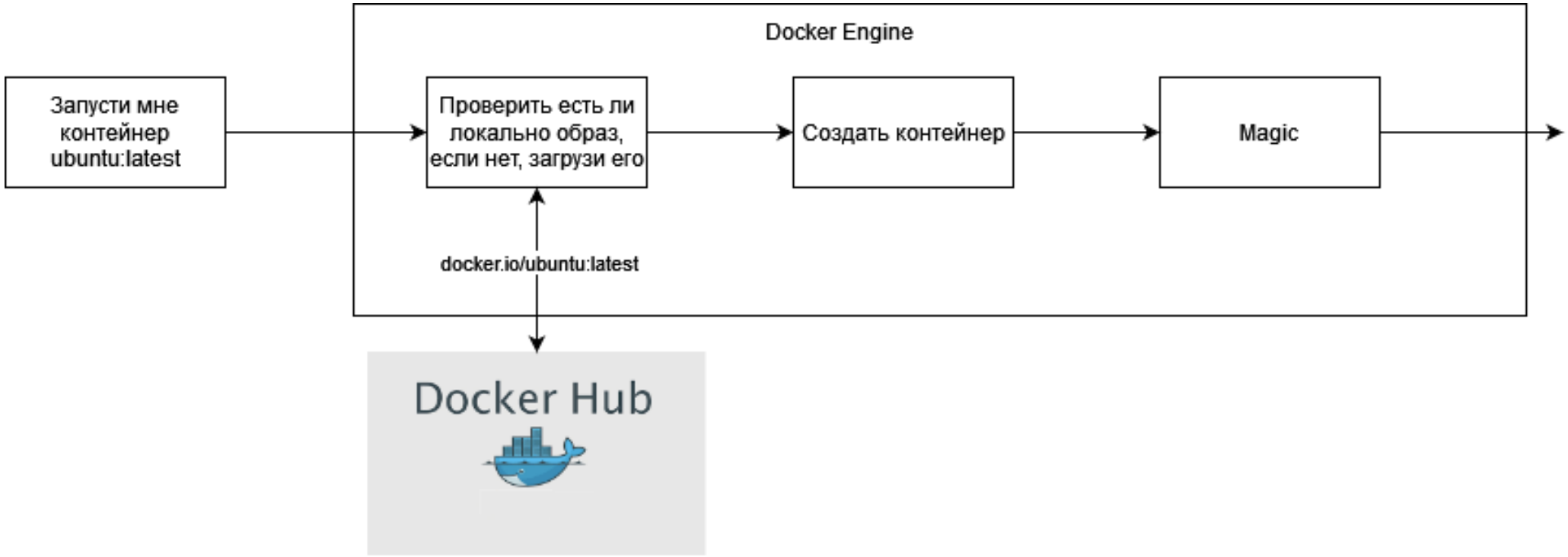
Нужен образ

- Нужно проверить есть ли образ локально, если нет, то стянуть с репозитория
- По-умолчанию, все docker образа имеют префикс docker.io/…
  - Префикс можно подменить



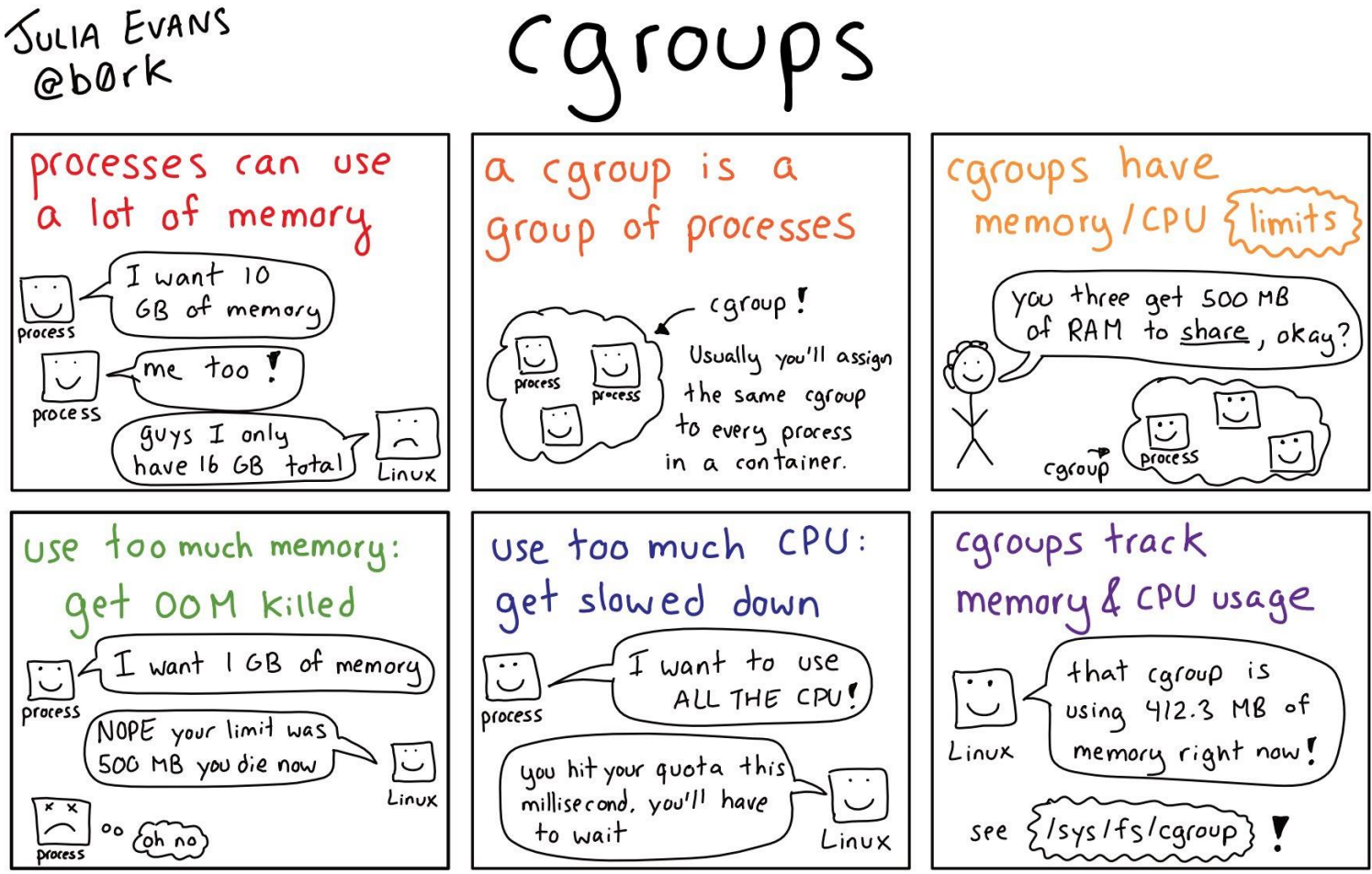
Нужно создать контейнер

- Docker Engine выполняет команду как по аналогии *docker container create*



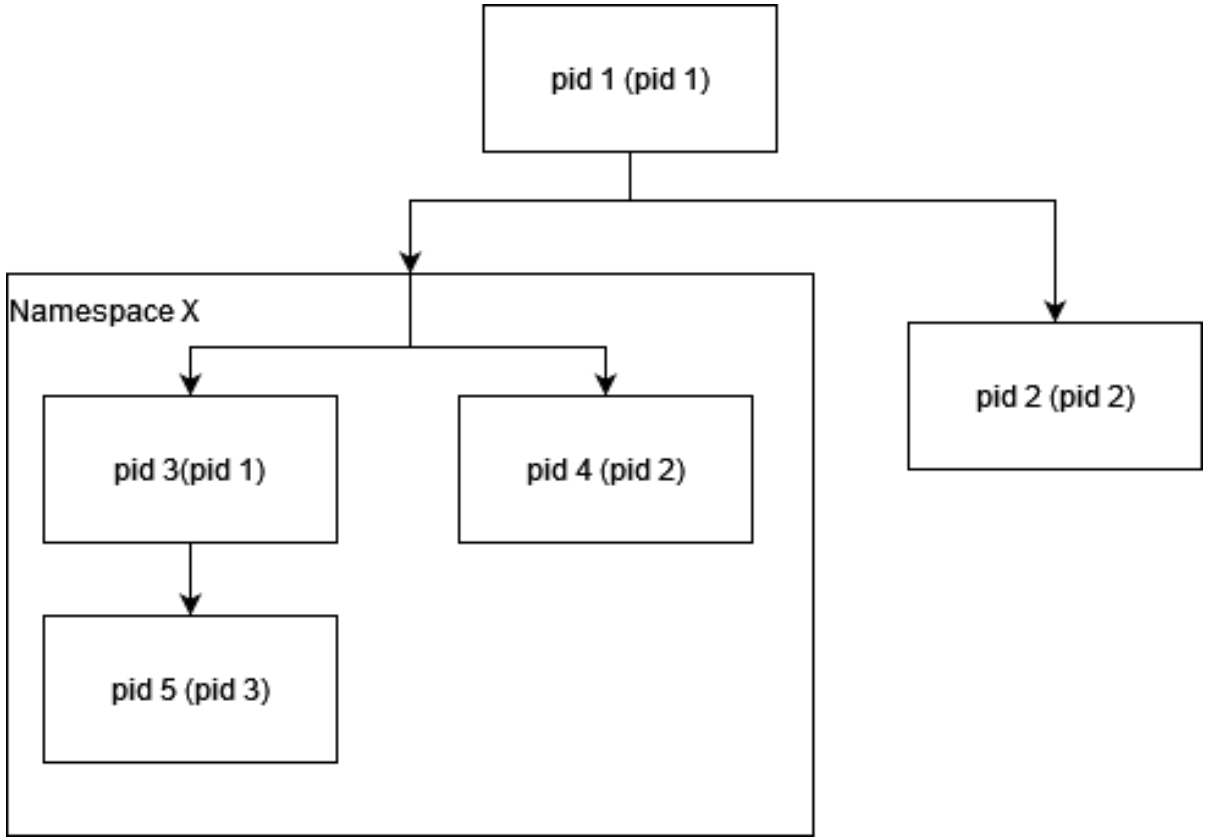
А ресурсы откуда?

- Control groups (cgroup https://www.kernel.org/doc/Documentation/cgroup-v1/cgroups.txt)
- Для контроля лимитов можно запустить: *docker stats*



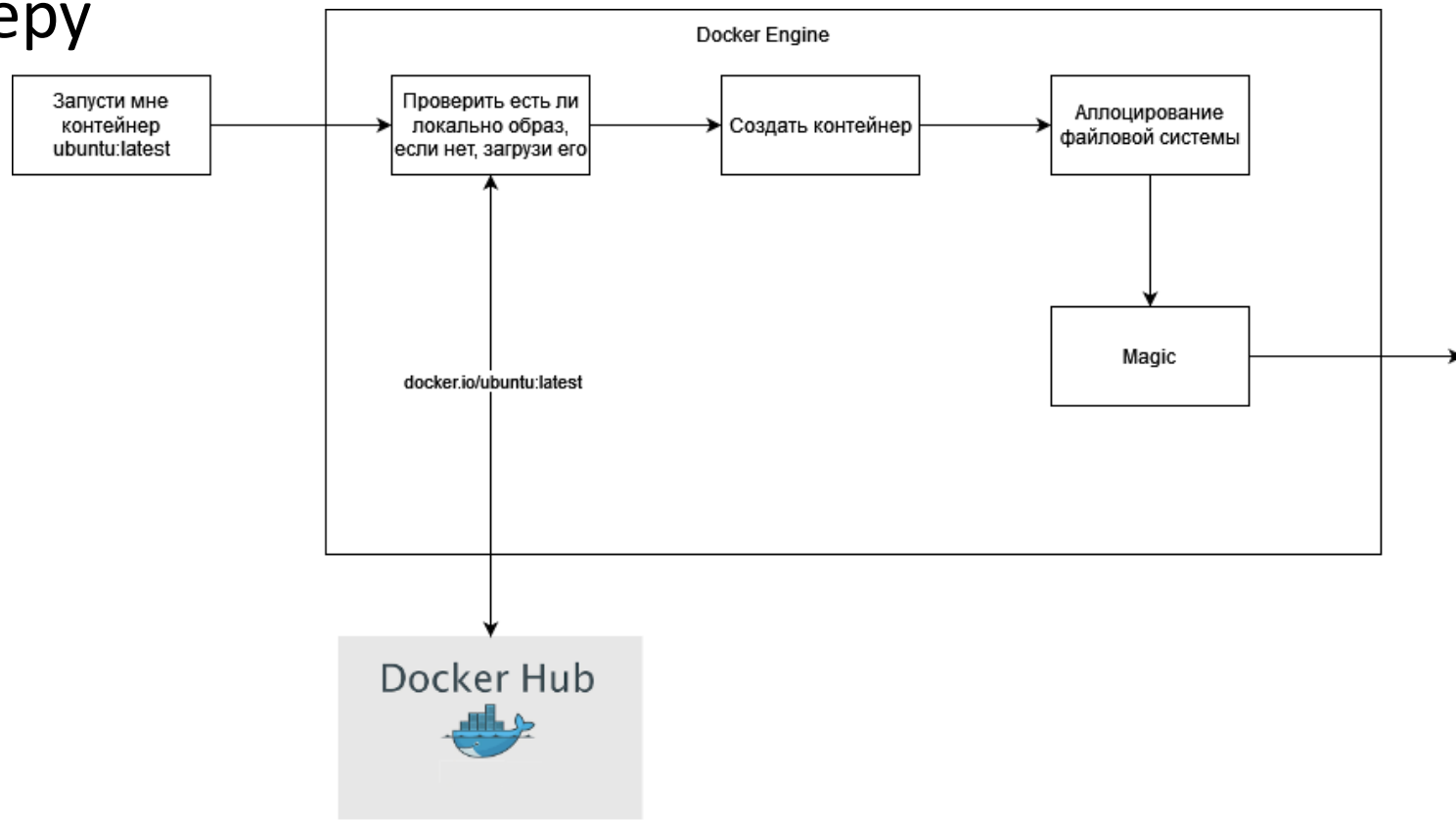
Наследование ресурсов (напоминание)

- Docker активно использует механизмы namespace для аллоцирования ресурсов



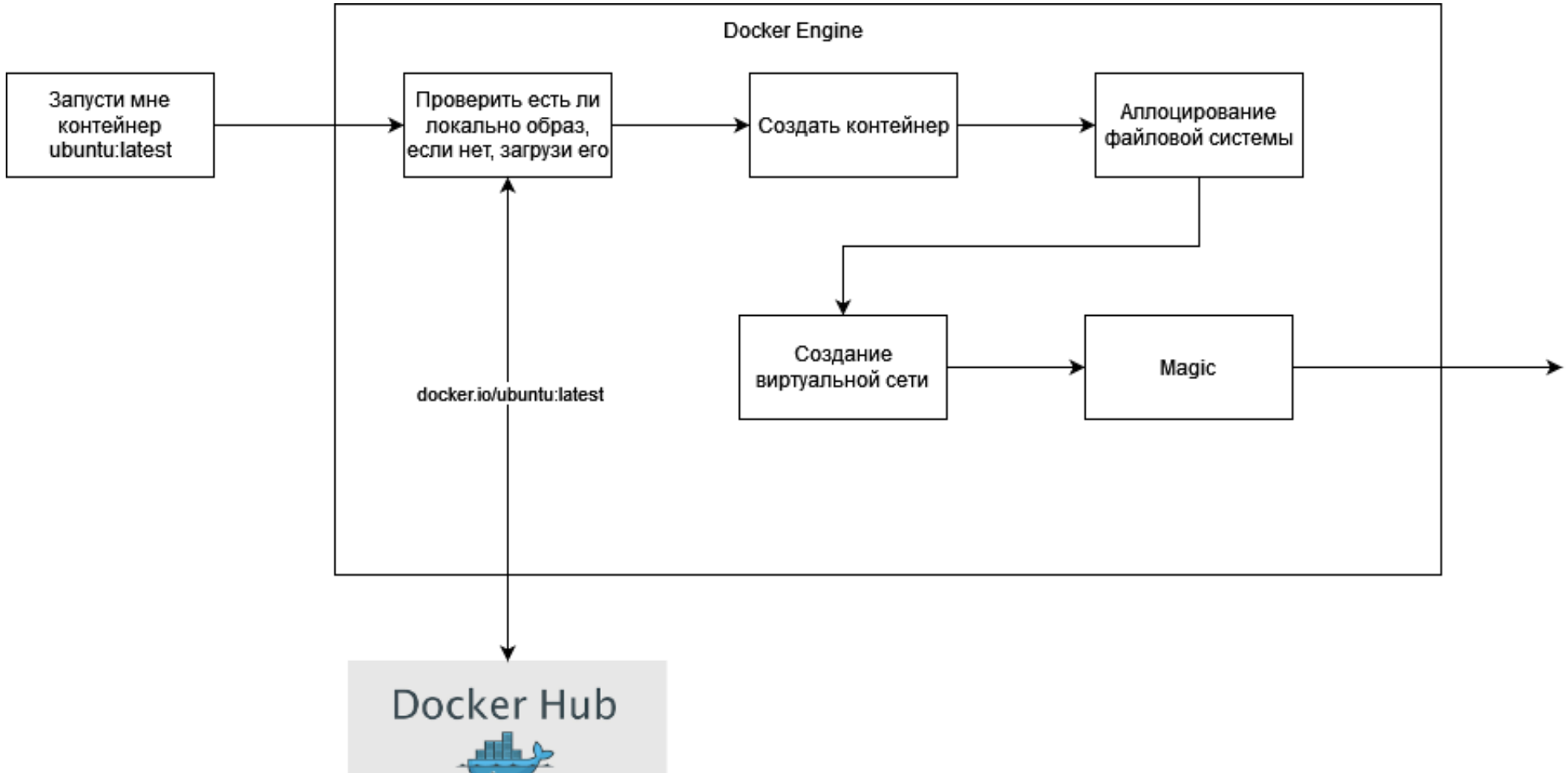
Аллоцирование файловой системы

- Docker аллоцирует файловую систему (для чтения/записи) к контейнеру
- Docker создаёт файлы, которые находятся внутри image на хостовой операционной системе



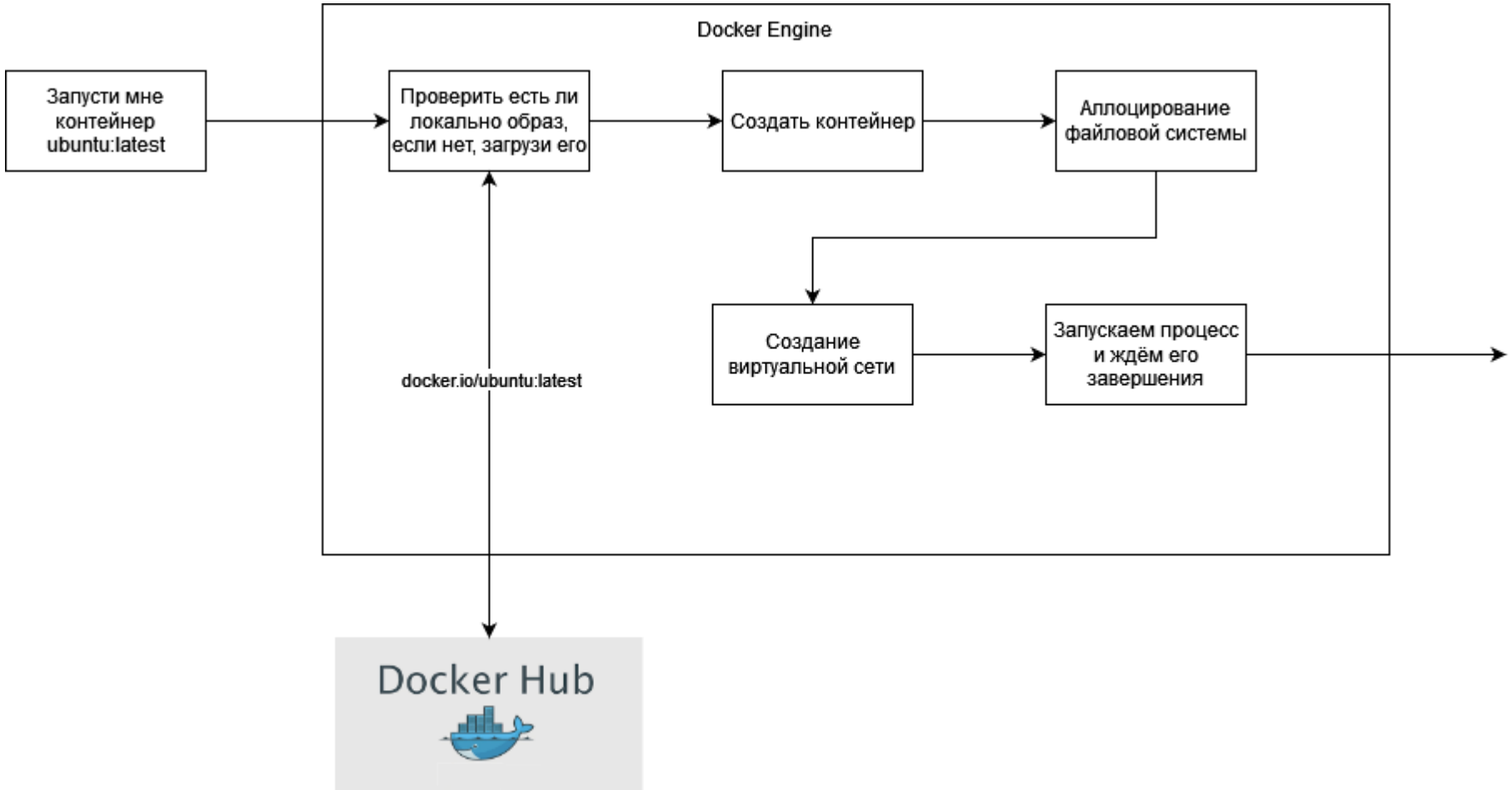
Создание виртуальной сети

- Docker создёт виртуальную сеть



Финальный шаг

- Запуск скрипта внутри контейнера и ожидание его завершения



Итоговая структура работы Docker