



Особенности построения виртуальных лабораторий на основе платформы Docker



Газуль Станислав Михайлович

Заведующий сектором информационного обеспечения приёма студентов УОПСР
Санкт-Петербургский государственный экономический университет

Кияев Владимир Ильич

профессор кафедры информатики
Санкт-Петербургский государственный экономический университет

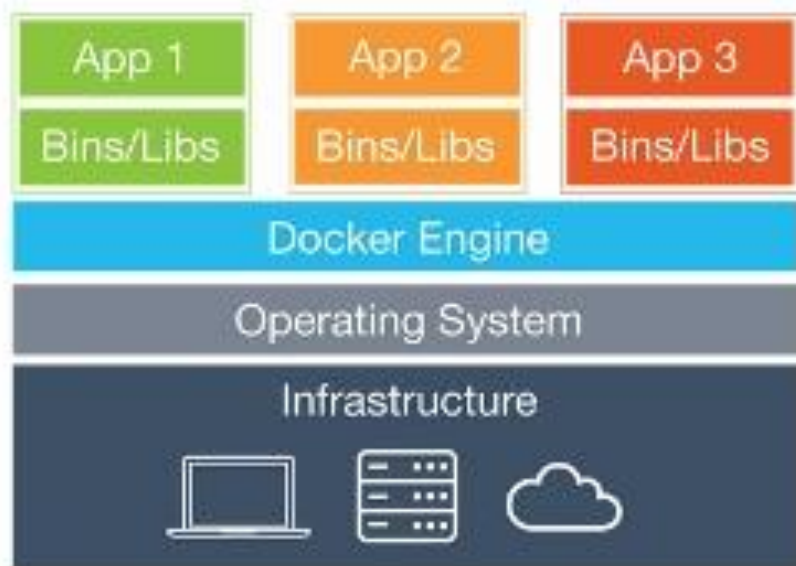


Описание проблем

- Необходимость проведения лекционных и лабораторных занятий с большим количеством групп студентов разных специальностей
- Проблемы с сохранением прогресса по выполненным заданиям и контролем качества усвоенных знаний
- Регулярная потребность в типовых информационных ресурсах, которые можно использовать после выполнения конкретных комплексов лабораторных работ
- Нехватка квалифицированных специалистов в университетах для поддержки компьютерных классов
- Ограниченные серверные мощности, которых недостаточно для размещения требуемых образовательных ресурсов

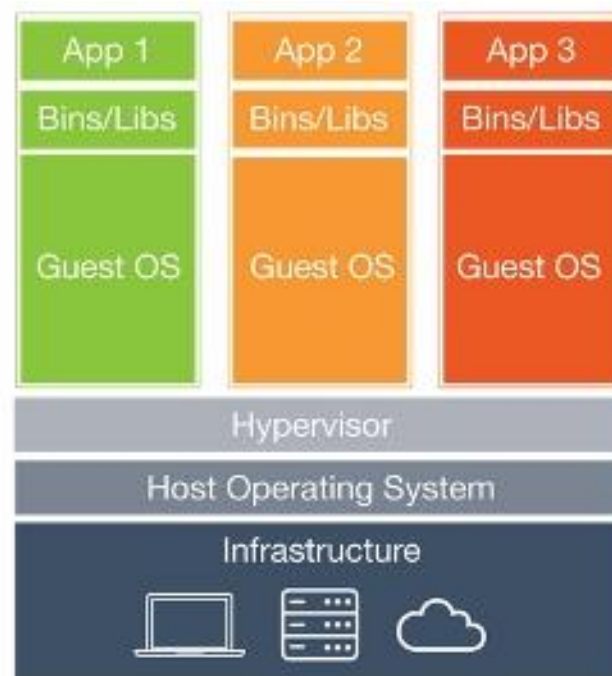
Способы решения указанных проблем

- Контейнеры



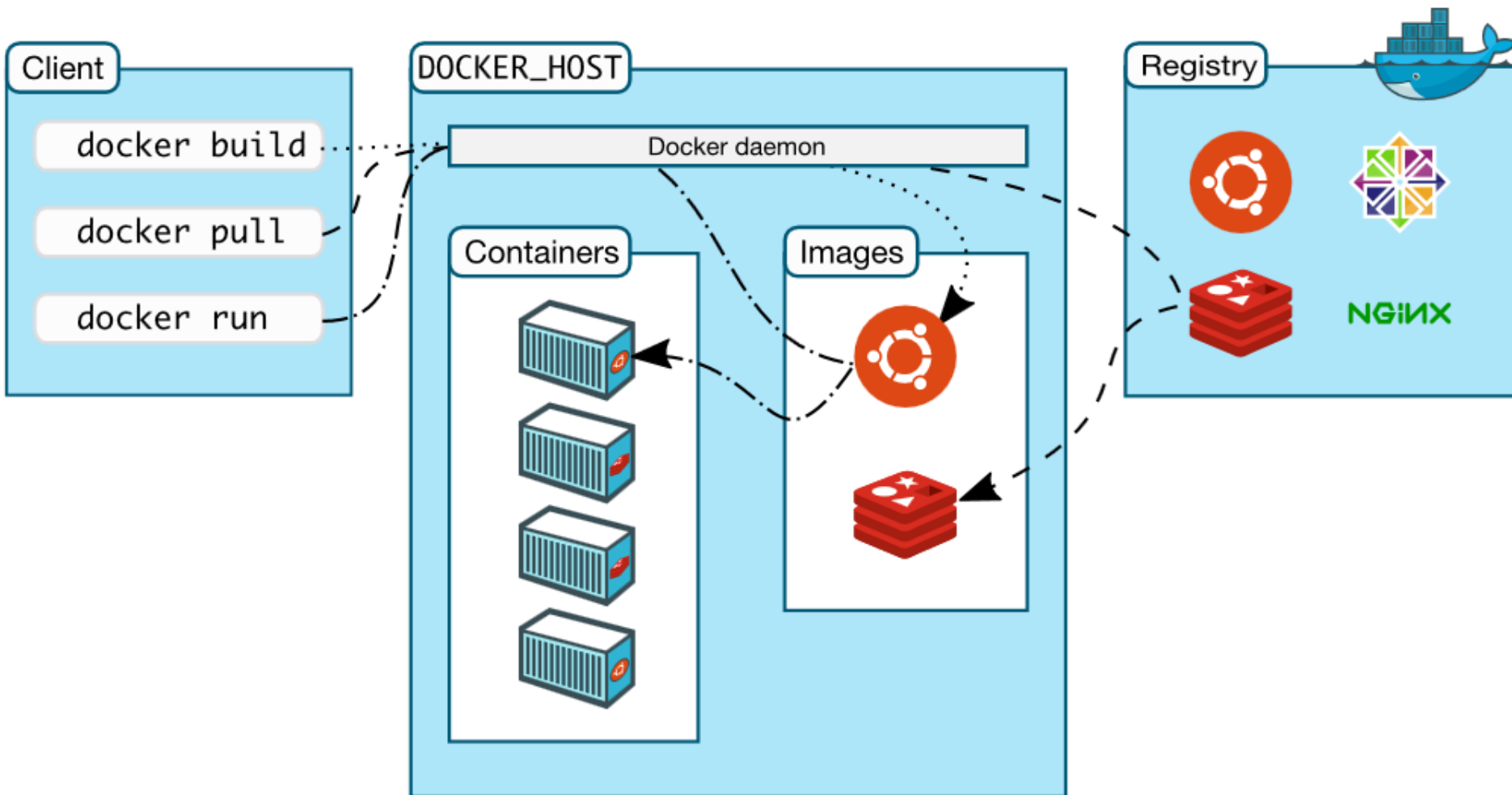
Containers

- Виртуальные машины



Virtual Machines

Архитектура Docker





Эксперимент

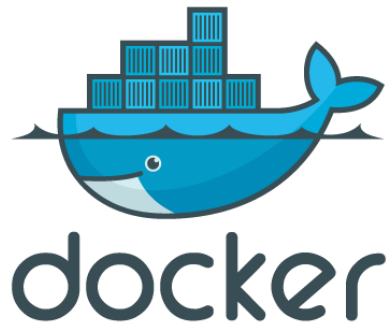
Сравнение времени подготовки к работе виртуальных машин kvm и контейнеров Docker для аналогичных приложений

Демонстрационный стенд

Компонент	Спецификация
Системная плата	Intel S2400GP2 Granite Pass
Процессор	Intel(R) Xeon(R) CPU E5-2407 0 @ 2.20 ГГц, 4 ядра
Оперативная память	8 Gb., PC3-12800, 1600 МГц, DDR3, ECC Reg.
Жесткий диск	Объем: 60 Гбайт; Скорость вращения шпинделя: 5400 RPM; Cache buffer: 8 Мбайт.
Сетевой интерфейс	Ethernet 100 Мбит/с.



Демонстрационный стенд



- Платформа Ubuntu Server 16.10
- Docker 1.12.6
- KVM

Шаблоны:

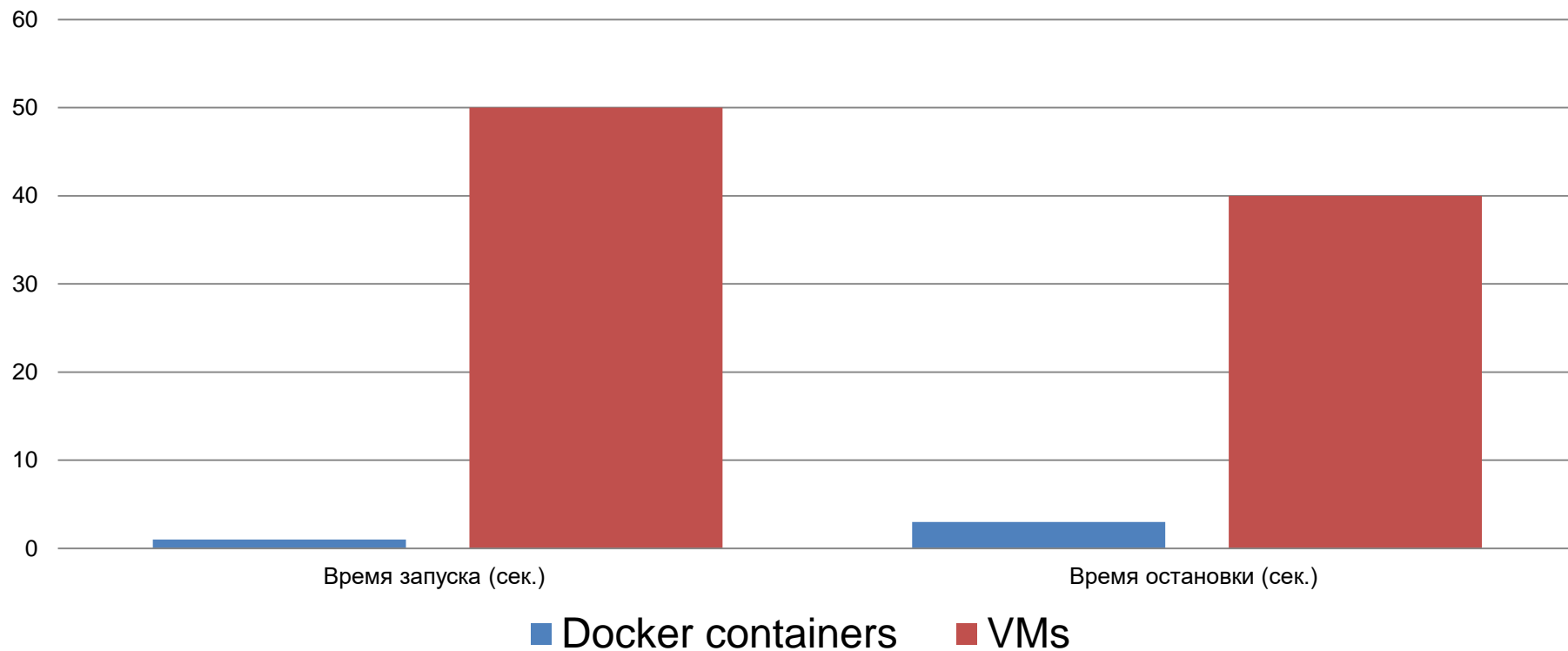
- MySQL Server
- phpMyAdmin
- Drupal





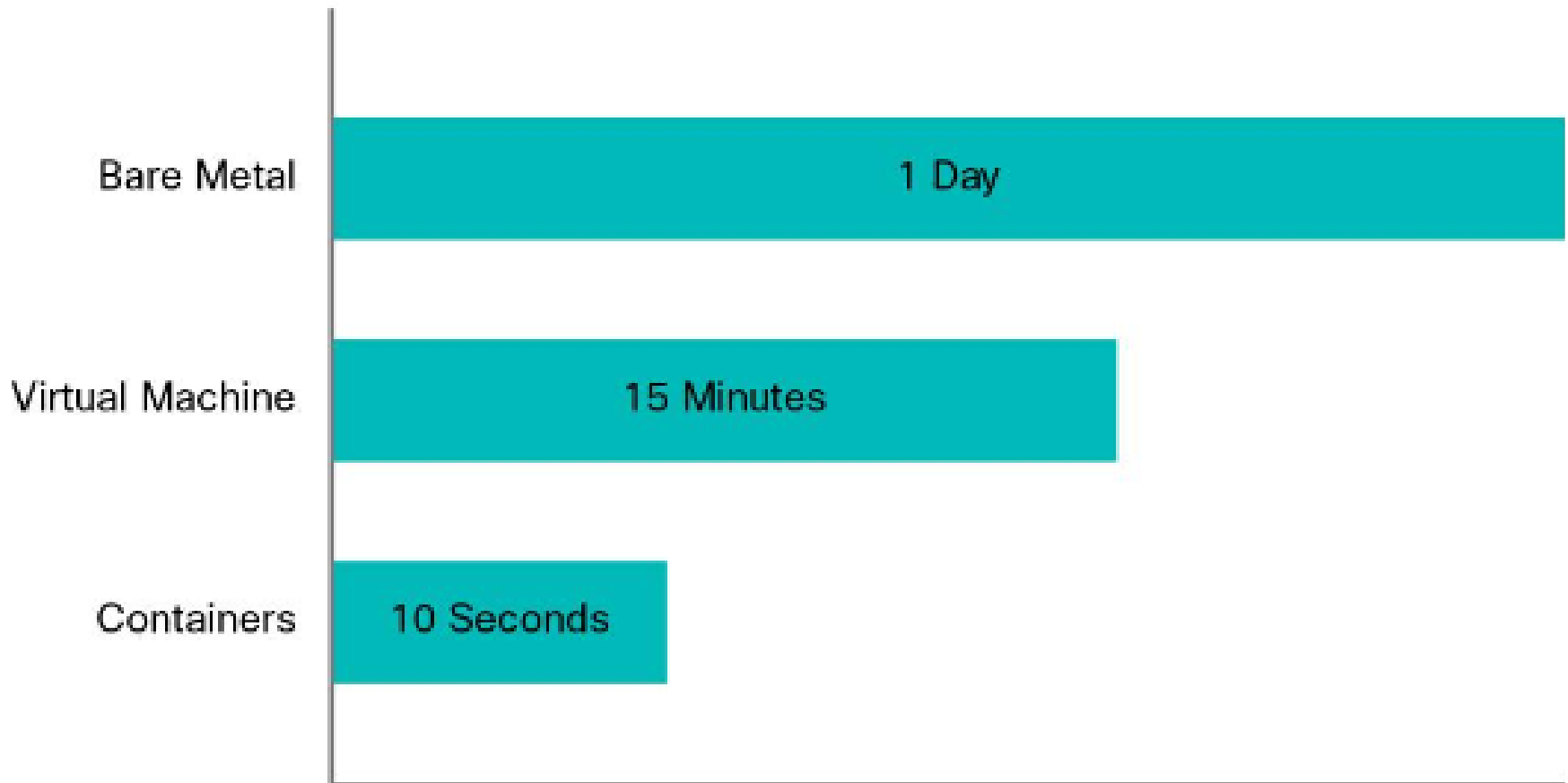
Сравнение производительности (Docker и kvm)

	Время запуска (сек.)	Время остановки (сек.)
Docker containers	1	3
VMs	50 секунд	40 секунд





Время подготовки к работе ИТ-ресурса



Данные, полученные в результате эксперимента
(данные компании Cisco)




Сравнение производительности виртуальной машины kvm (гипервизор) и контейнера Docker для вычислительных операций

	Параметры	Примечания
Процессор	Два 2,4—3.0 Гц. Intel Sandy Bridge-EP Xeon E5-2665, суммарно 16 ядер	Два процессора/сокета соединены через QPI, что обеспечивает неравномерный доступ к памяти (NUMA).
Оперативная память	256 Гбайт	

Workload	Native	Docker	KVM-untuned	KVM-tuned
PXZ (MB/s)	76.2 [±0.93]	73.5 (-4%) [±0.64]	59.2 (-22%) [±1.88]	62.2 (-18%) [±1.33]
Linpack (GFLOPS)	290.8 [±1.13]	290.9 (-0%) [±0.98]	241.3 (-17%) [±1.18]	284.2 (-2%) [±1.45]

Данные, полученные в результате эксперимента
(данные корпорация IBM)





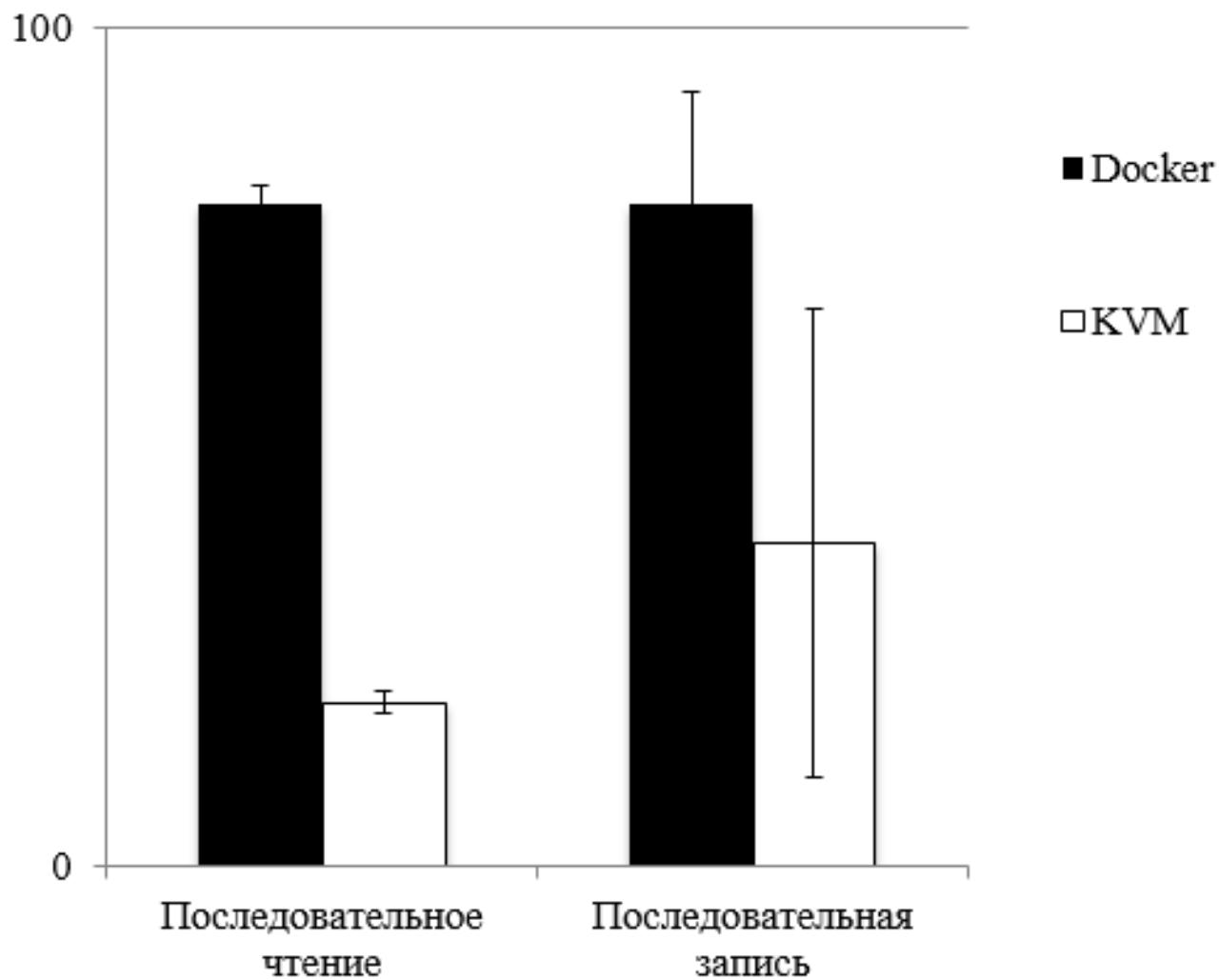
Эксперимент

Сравнение производительности виртуальной машины kvm и контейнера Docker при работе с хранилищем данных и сетевым интерфейсом



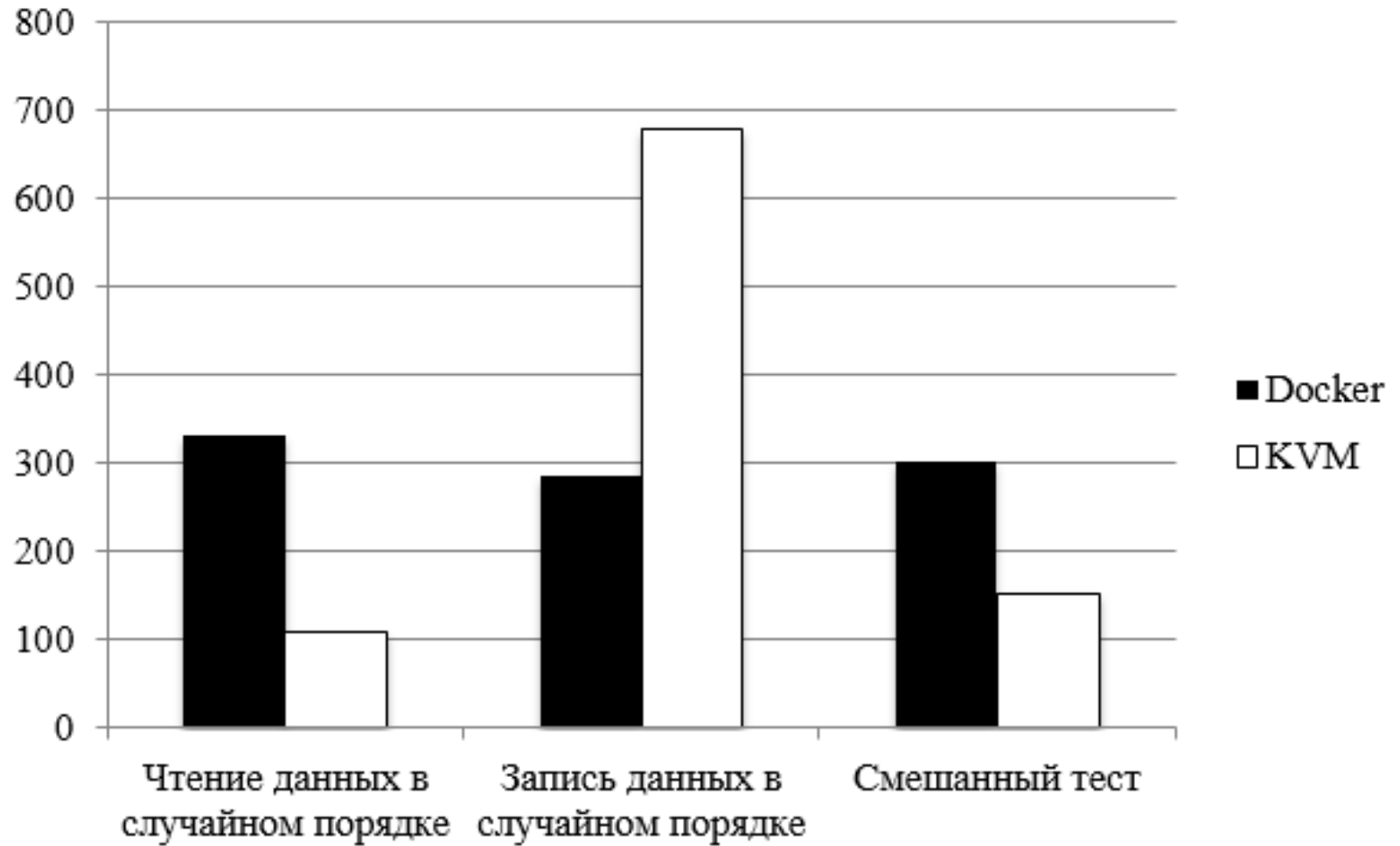


Результаты теста fio: последовательное чтение и запись (Мбайт/с)



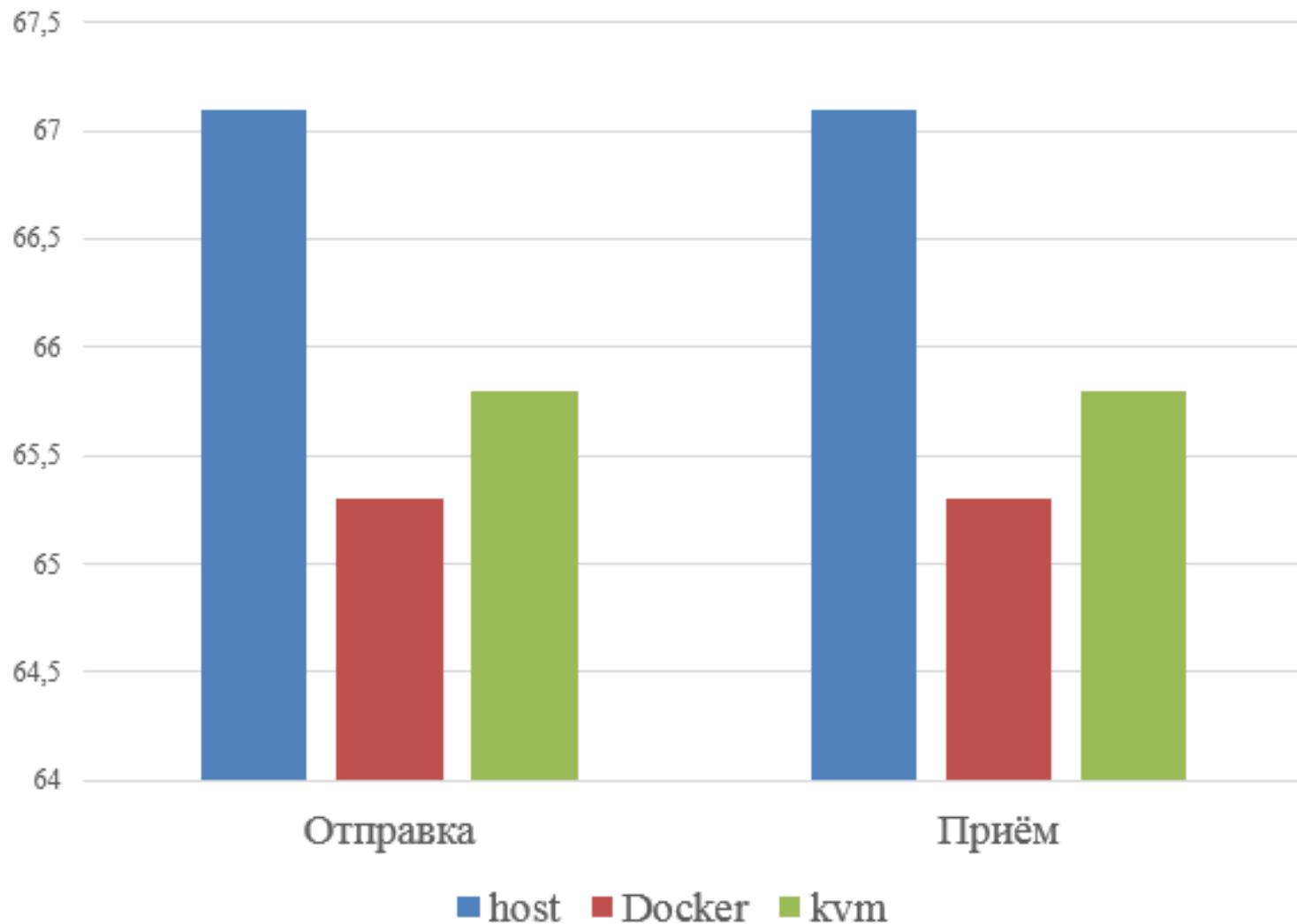


Результаты теста fio: чтение и запись данных в случайном порядке и смешанный тест (iops)



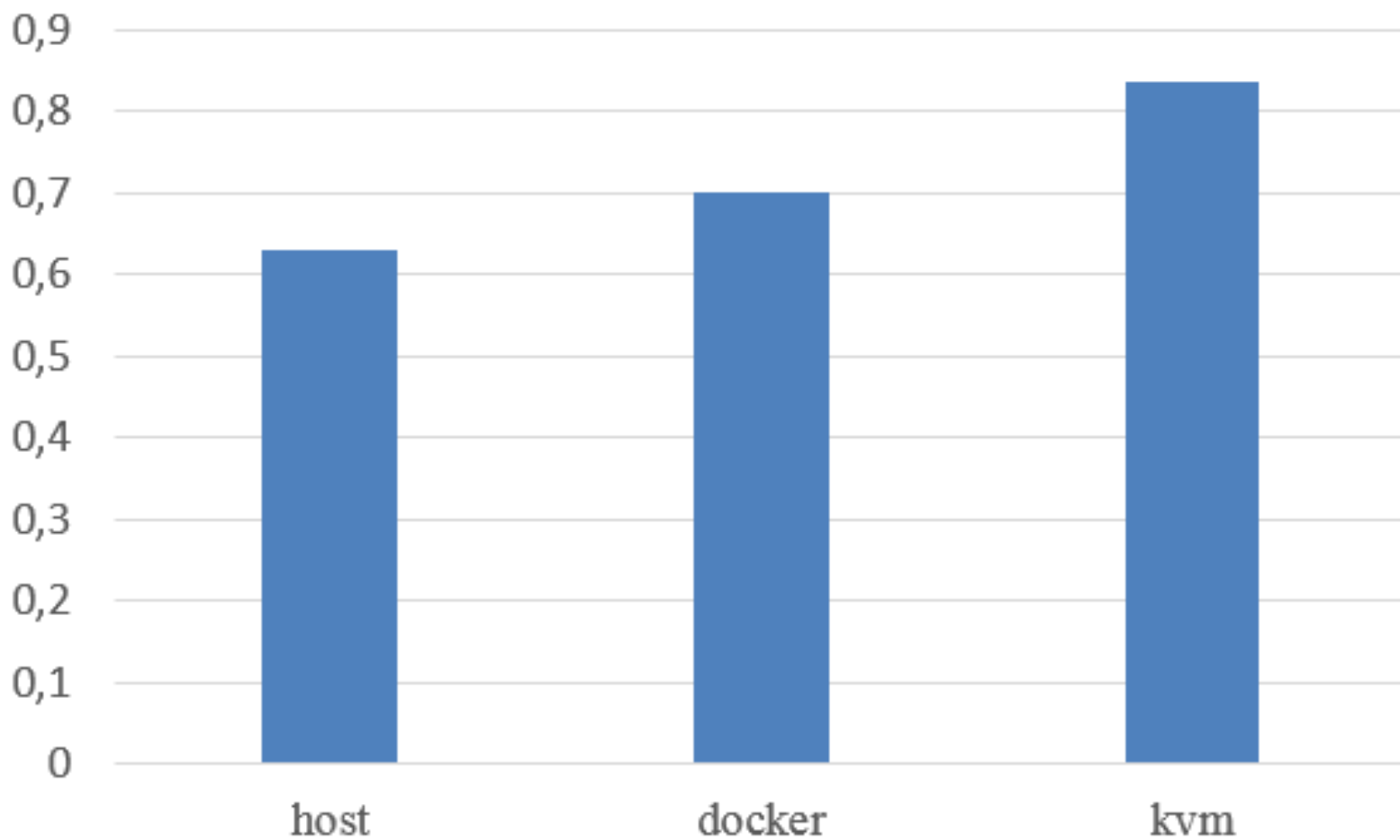


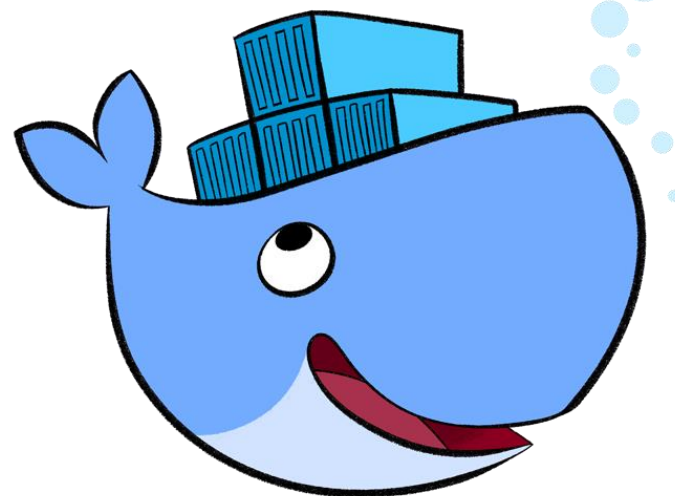
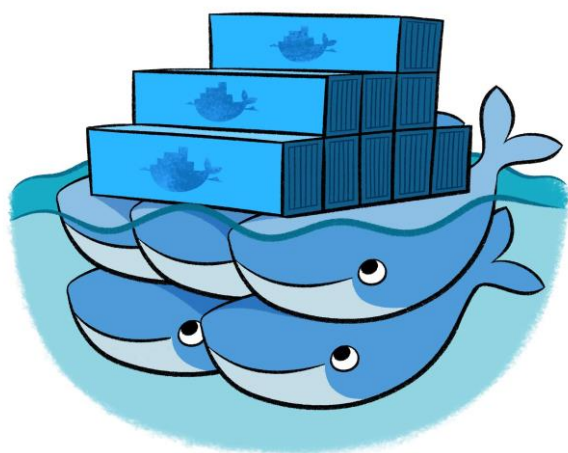
Результаты теста iperf (усреднённые оценки, Мбит/с)





Результаты применения утилиты ring (усреднённые оценки, мс)






Результаты экспериментов и тестов убедительно показали, что при размещении большого количества однотипных виртуальных ресурсов использование контейнеров под управлением Docker позволяет добиться *лучшей производительности* — как при работе с хранилищем данных, так и при работе с сетевыми интерфейсами



Образовательные сервисы построенные на основе контейнеров

Пример построения образовательных сервисов
(лабораторий) на базе контейнерной технологии Docker



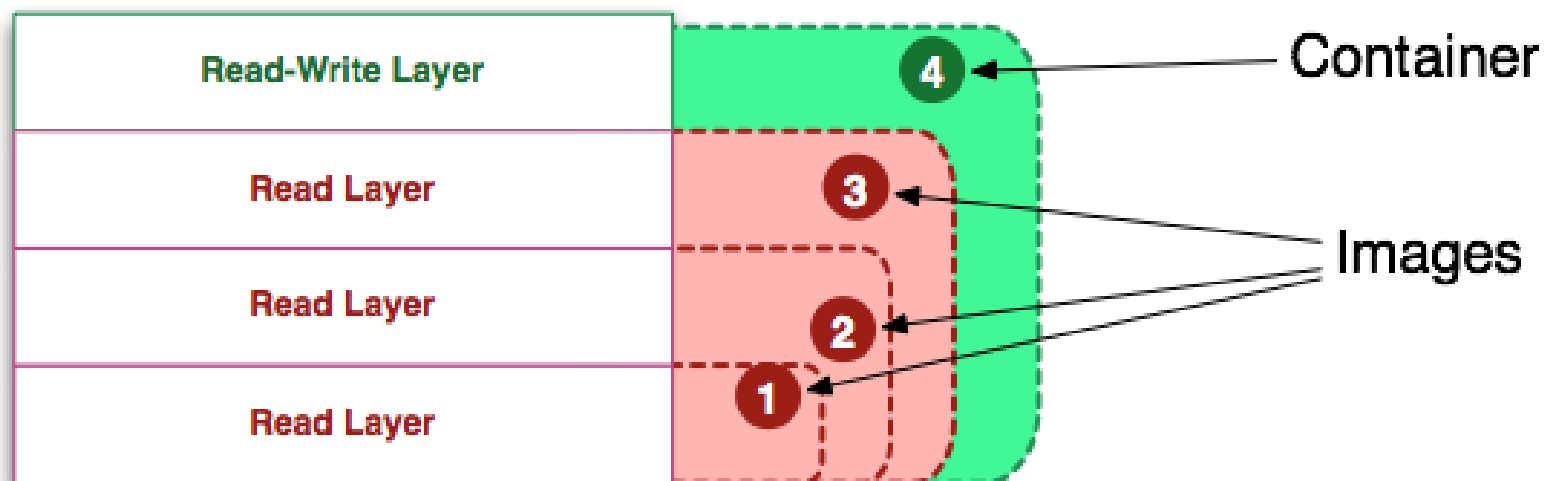


Организация и проведение лабораторных работ в рамках учебного курса с применением Docker-технологии

Сценарий:

- на первом занятии студентам необходимо получить навыки развёртывания web-сервера
- на втором занятии студентам необходимо приступить к изучению CMS

Структура ИТ-лаборатории с применением Docker-технологии



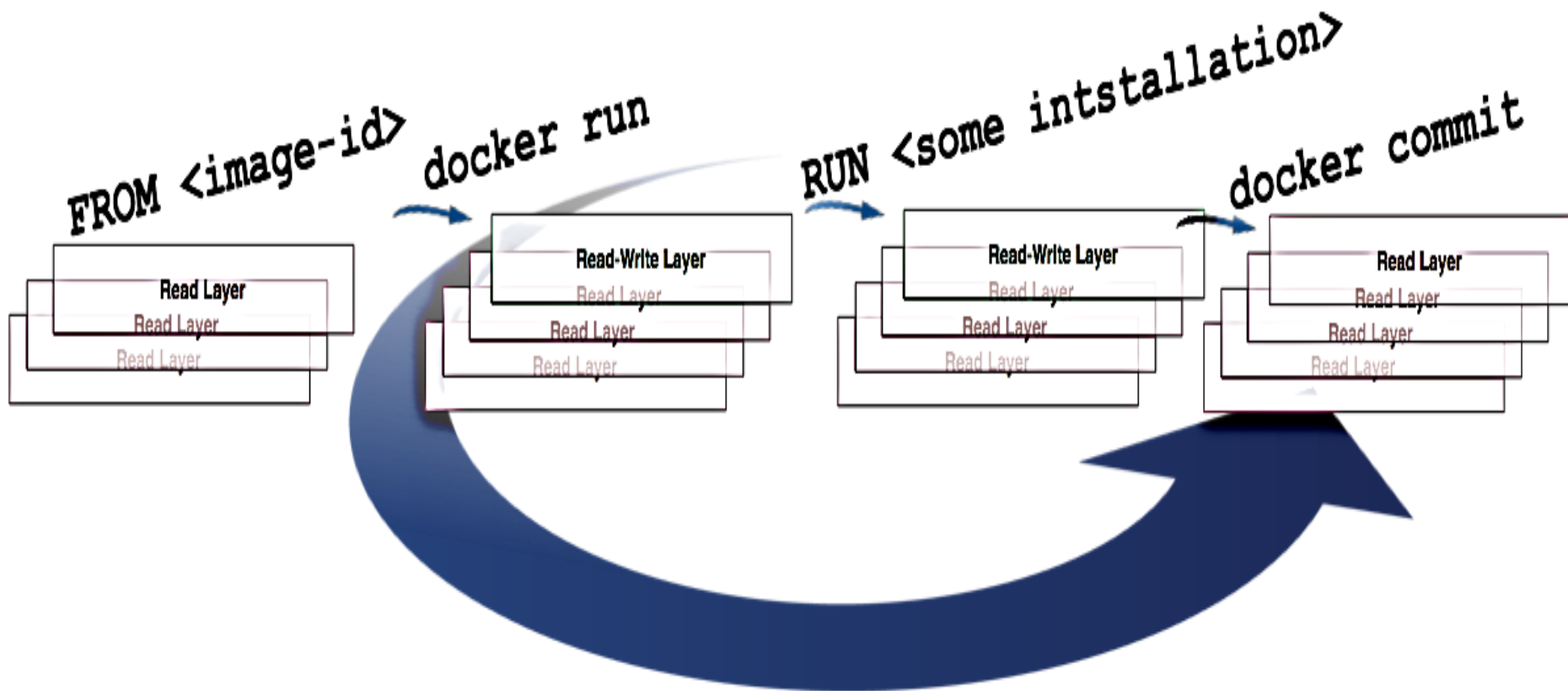
Мы предлагаем сценарий организации такой ИТ-лаборатории при помощи скриптов (например, bash-скрипт).

При этом *автоматически* создаётся необходимое количество контейнеров (по числу обучающихся и преподавателей).

Каждый из контейнеров будет обладать необходимым количеством указанных выше «слоёв».



Алгоритм подготовки шаблона ИТ-лаборатории с применением платформы Docker





Пример автоматизации процесса создания контейнеров с помощью инструмента Docker compose

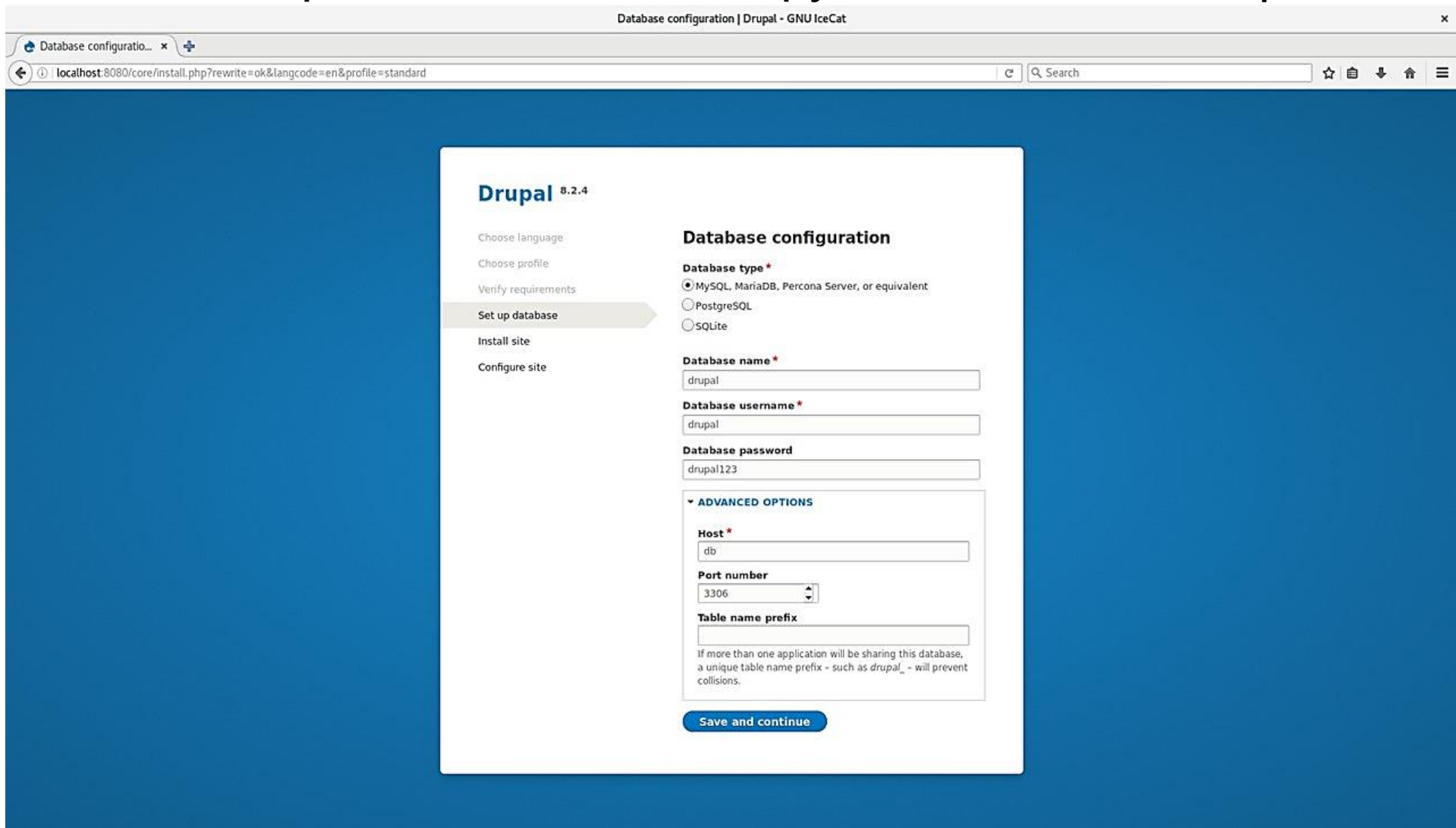
```
version: '2'

networks:
  drupal:

services:
  db:
    image: "mariadb:10.1"
    volumes:
      - $PWD/db:/var/lib/mysql:z
    restart: always
    networks:
      - drupal
    environment:
      MYSQL_ROOT_PASSWORD:
"drupal_admin_pass"
      MYSQL_DATABASE: "drupal"
      MYSQL_USER: "drupal"
      MYSQL_PASSWORD: "drupal123"
```

```
app:
  depends_on:
    - db
  image: "drupal:8-apache"
  # volumes:
  # - $PWD/app:/var/www/html:z
  ports:
    - "8080:80"
  restart: always
  networks:
    - drupal
```

Пример автоматизации процесса создания контейнеров с помощью инструмента Docker compose



The screenshot shows a web browser window with the URL `localhost:8080/core/install.php?rewrite=ok&langcode=en&profile=standard`. The page title is "Database configuration | Drupal - GNU IceCat". The main content area is titled "Drupal 8.2.4" and contains a navigation menu on the left with the following items: "Choose language", "Choose profile", "Verify requirements", "Set up database" (highlighted), "Install site", and "Configure site".

The "Database configuration" section is active and contains the following fields:


- Database type ***: MySQL, MariaDB, Percona Server, or equivalent; PostgreSQL; SQLite
- Database name ***:
- Database username ***:
- Database password**:

Below these fields is a section titled "ADVANCED OPTIONS" with a dropdown arrow:

- Host ***:
- Port number**:
- Table name prefix**:

A note below the advanced options states: "If more than one application will be sharing this database, a unique table name prefix - such as `drupal_` - will prevent collisions."

At the bottom of the configuration area is a blue button labeled "Save and continue".



Демонстрация: виртуальные лаборатории с применением Microsoft HoloLens

Образовательные сервисы по
запросу — новые возможности

Виртуальные лаборатории и дополненная реальность

Проект **Sm4rtLab**, реализованный компанией Process Genius Oy на базе Университета Восточной Финляндии

Сервис позволил преподавателям и студентам:

- подключаться к реальной лаборатории (с помощью очков дополненной реальности Microsoft HoloLens или через веб-сайт)
- управлять лабораторным оборудованием, наблюдая за результатом эксперимента удалённо

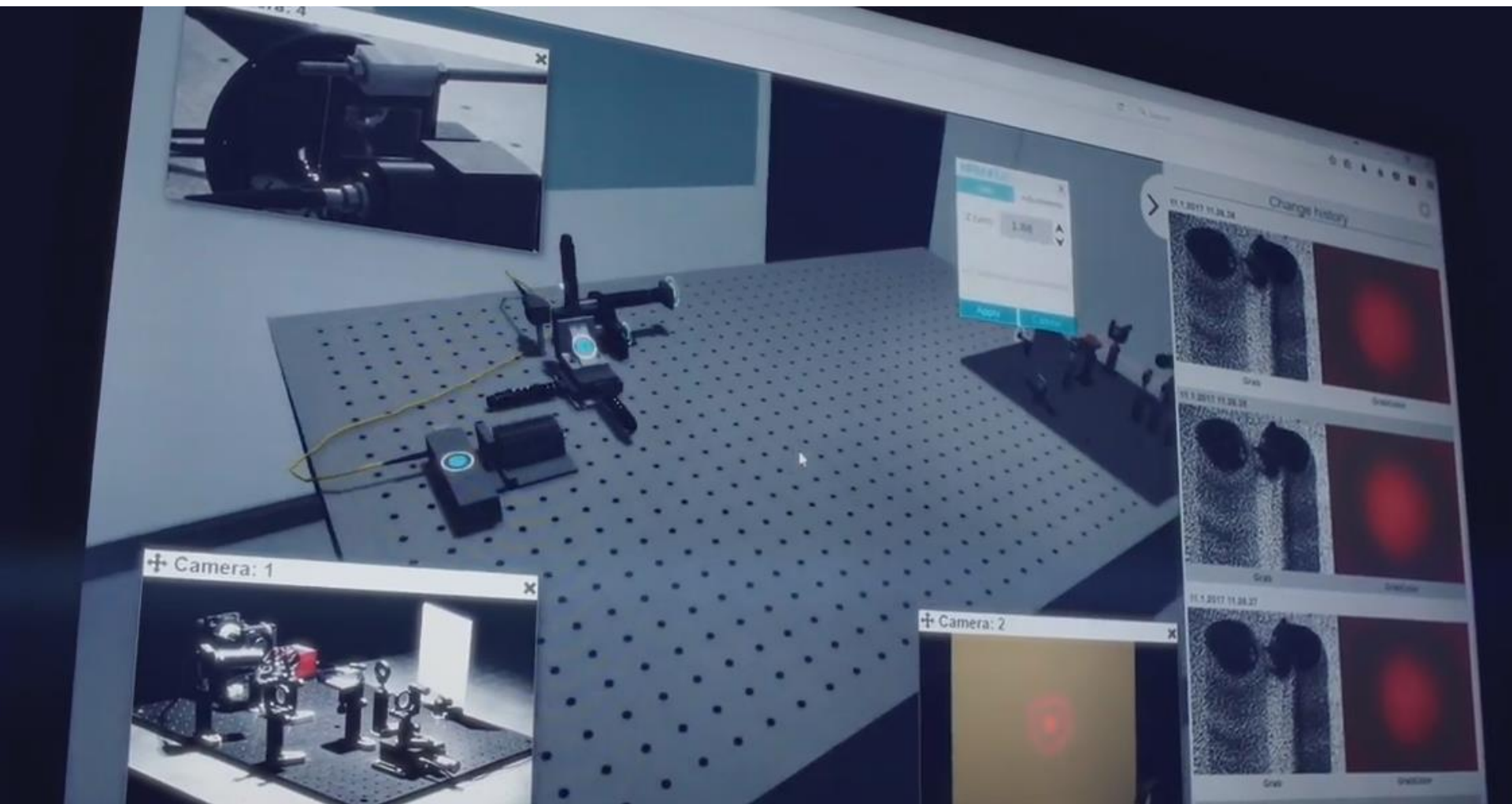


<https://www.processgenius.fi/fi/>

Виртуальные лаборатории и дополненная реальность



Виртуальные лаборатории и дополненная реальность



Виртуальные лаборатории и дополненная реальность





Выводы

- Контейнерная виртуализация на основе Linux-контейнеров Docker чрезвычайно эффективна для формирования тестовой (производственной) и учебной сред
- Такой подход обеспечивает отсутствие «зависимости от одного производителя» (Vendor lock) и позволяет значительно увеличить ресурсное поле при построении информационных инфраструктур
- Контейнерная виртуализация позволяет существенно экономить время и значительно уменьшить затраты на создание требуемых информационных ресурсов
- Рассмотренный подход позволяет разворачивать требуемые образовательные сервисы по запросу в рамках учебных курсов



Наши рекомендации

Перспективную платформу Docker можно с успехом применять для создания виртуальных лабораторий в широком спектре учебных курсов 😊:

- использовать `images` с разными наборами слоёв для лабораторных работ по настройке и конфигурированию необходимых ИТ-ресурсов
- подготовить шаблоны для многократного использования и существенной экономии времени и вычислительных ресурсов при подготовке к сложным комплексным лабораторным работам

Благодарим за внимание!



Заведующий сектором информационного обеспечения приёма студентов УОПСР СПбГЭУ

Газуль Станислав Михайлович

stanislav@gazul.ru



Профессор кафедры информатики СПбГЭУ

Кияев Владимир Ильич

kiyaev@mail.ru