

Red Hat

OpenShift Container Platform

Обзор облачной платформы по работе с контейнерами

Дмитрий Севостьянов
Системный архитектор
Red Hat

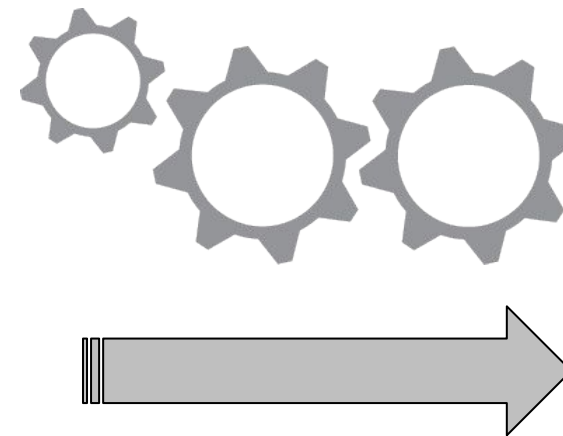
Содержание

- Что такое OpenShift
 - Современные вызовы и позиционирование продукта
 - Что дает OpenShift бизнесу, разработчикам и администраторам
 - Упрощение DevOps
 - Что входит в подписку
- Как работает OpenShift
 - Компоненты и их взаимодействие
 - Простейший CI/CD процесс

Что такое OpenShift

Трудности при решении задач IT

- IT не успевают за ростом бизнеса
- Постоянная потребность в новых сервисах (приложениях)
- Ускорение выхода систем (ввод в промышленную эксплуатацию)
- Сложность проектов (административная и технологическая)



Цифровая трансформация



- Цифровая трансформация
 - Ускорение развития бизнеса - это движущая сила IT
 - Агрегирует известные задачи - иметь доступ ко всем данным, через все каналы, с любой детализацией, и возможностью принимать решения на основе этих данных
- Разработка приложений должна учитывать требования
 - Архитектурные, административные,..
 - Иметь единый репозиторий исходного кода
 - Конфигурации сборок и зависимостей
 - Бизнес-процессы,..
 - Это "Методология" - 12 Factor App development
- Практики и архитектуры построения приложений, соответствующие духу методологии
 - DevOps, Agile, WaterFall
 - Bimodal IT
 - Microservices
- Технологии, наилучшим образом соответствующие решению задач, но в то же время являющимися универсальными
 - Контейнеризация, виртуализация, облачные вычисления
 - Web 2.0, REST API, Integration patterns
 - Организация проектной деятельности

Жизненный цикл разработки приложений

Квинтэссенция задач IT

- Постановка задачи
 - Бизнес-аналитика и архитектура
- Реализация
 - Разработка сервисов
 - Создание программного кода
 - Интеграция
- Тестирование
 - Стабилизация кода
- Внедрение
 - Развертывание и поддержка



OpenShift используется на трех (из четырех) этапах при решении пяти-шести (из восьми) шагов жизненного цикла

Что такое OpenShift

- Это система (платформа) для разработки и выполнения приложений (микросервисов), упакованных в контейнеры
 - С использованием различных языков и технологий (PHP, Ruby, Python, JEE, Spring, .NET, PostgreSQL, MySql, MongoDB,..)
 - В изолированных окружениях (dev/test/QA/prod; front-/back-;..)
 - С автоматической балансировкой нагрузки и отказоустойчивостью
- Технологические основы
 - Методы изоляции ядра Linux
 - Образы контейнеров в формате Docker и рантайм OCI
 - Распределенное управление Kubernetes
 - Сетевое взаимодействие SDN OpenVSwitch
- Обеспечивает DevOps
 - Самообслуживание
 - Blue/Green deployment, A/B routing
 - Интеграция с Jenkins

Позиционирование OpenShift

Облачная платформа для приложений



Есть сомнения в необходимости использования OpenShift?

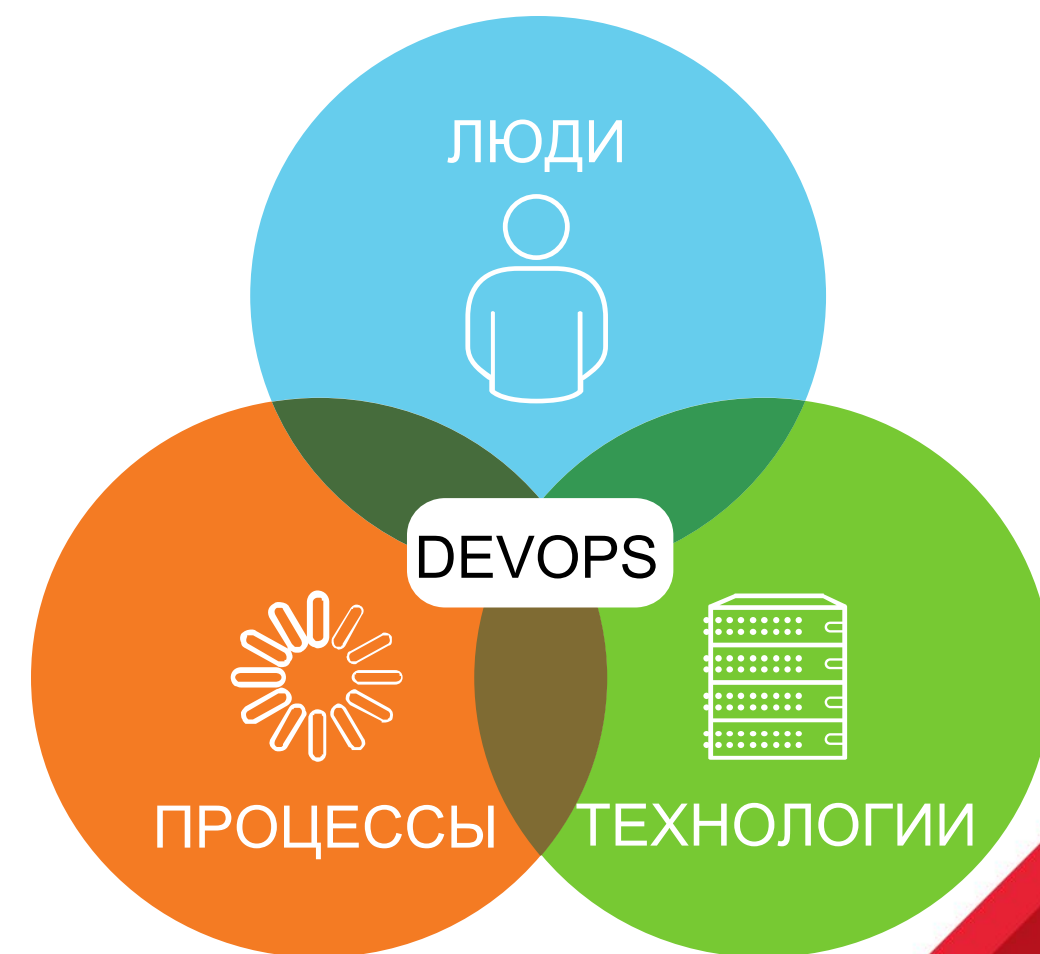
Попытайтесь ответить на вопросы

- Есть разработка (в том числе на разных технологиях)?
- Есть потребность создавать однотипные среды и контуры?
- Есть требования к отказоустойчивости, балансировке нагрузки, переключению на новые версии (автоматически или по событию)?
- Есть необходимость оптимизировать процесс разработки-тестирования-эксплуатации?
- Есть ли аутсорсинг?

Что дает OpenShift бизнесу,
разработчикам и администраторам

Что дает OpenShift для бизнеса

- Экономия денег/ресурсов при поддержке жизненного цикла и ускорение выхода на рынок
 - Реализация, тестирование, внедрение
 - Уменьшение затрат на эксплуатацию
- Внедрение практики DevOps
 - Методология создания и поддержки программных решений более эффективно путем налаживания автоматизированного процесса взаимодействия между различными командами (Разработчики, Тестировщики, Администраторы, Поддержка)
 - Оптимизация бизнес-процессов предприятия — сокращение шагов для увеличения производительности труда (количество → качество)



Что дает OpenShift разработчикам

- Привычная разработка

```
[user@host folder]$ git clone
```

```
[user@host folder]$ git commit
```

```
[user@host folder]$ git push
```

- Влияние на среду разработки и выполнения (необходимые наборы версий ПО)
- Сокращение времени регламентов (вследствие оптимизации процесса создания среды и развертывания приложения)

Что дает OpenShift администраторам

Контроль над средой выполнения

- Ограничения по ресурсам
- Возможность размещения приложения по соответствию критерию
- Обновления системы, не затрагивая кода разработчиков
- Бесконфликтное сосуществование разных версий библиотек
- Управление жизненным циклом процессов (сервисов)
 - Не все приложения имеют встроенное управление жизненным циклом

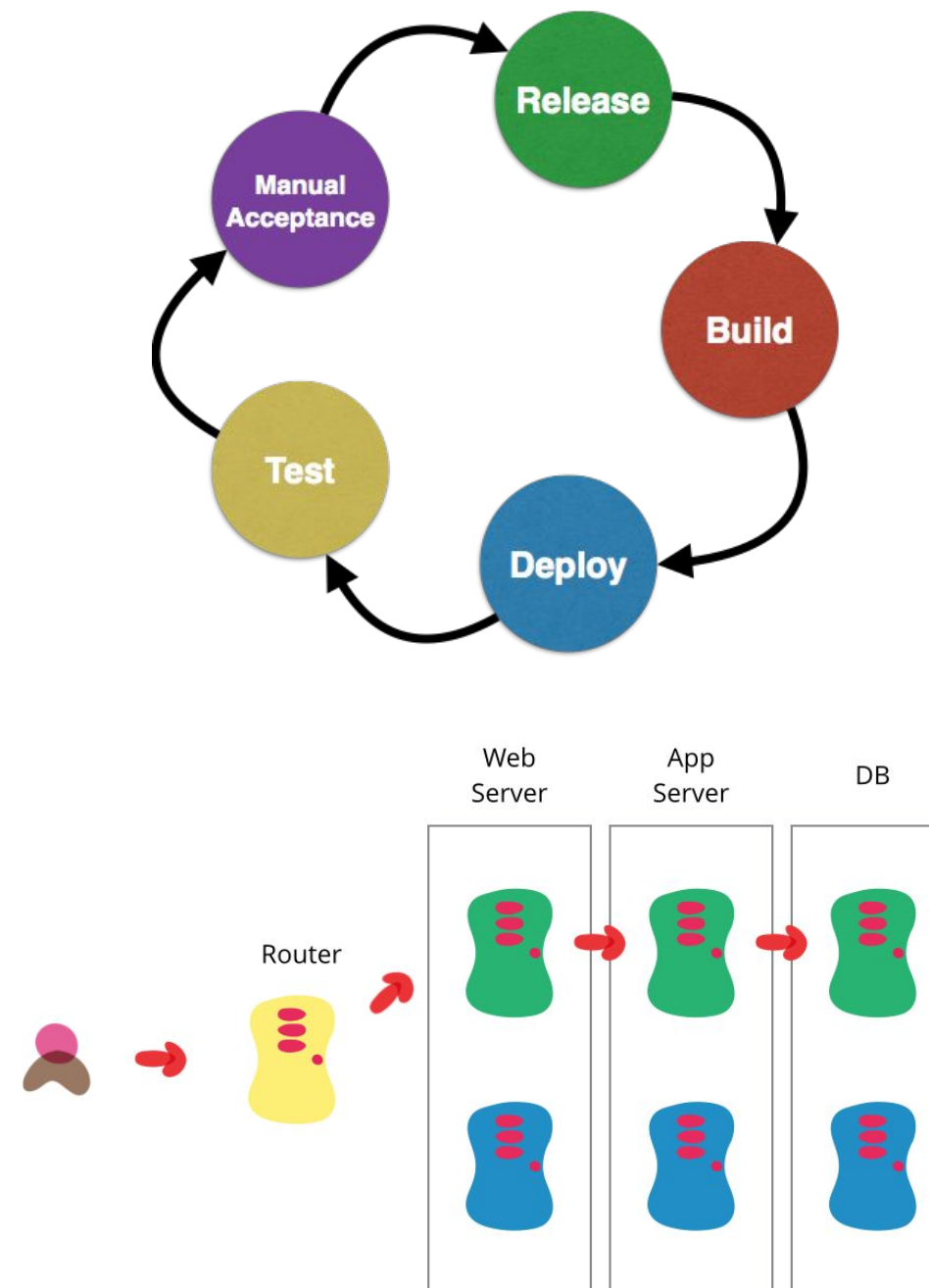
Унификация обслуживания и увеличение плотности сервисов

- Единообразный API для всех контейнеров
- Плотность контейнеров по сравнению с VM примерно 10/1
- Ограничение - только RHEL x86_64

Организация DevOps

Организация DevOps

- CI/CD (непрерывная разработка и внедрение)
- Быстрое обнаружение и исправление ошибок
- Самообслуживание
- 100%-ная автоматизация
- Ввод в промышленную среду
- Проактивный мониторинг и отчетность
- Blue/Green развертывание и A/B тестирование



Организация DevOps

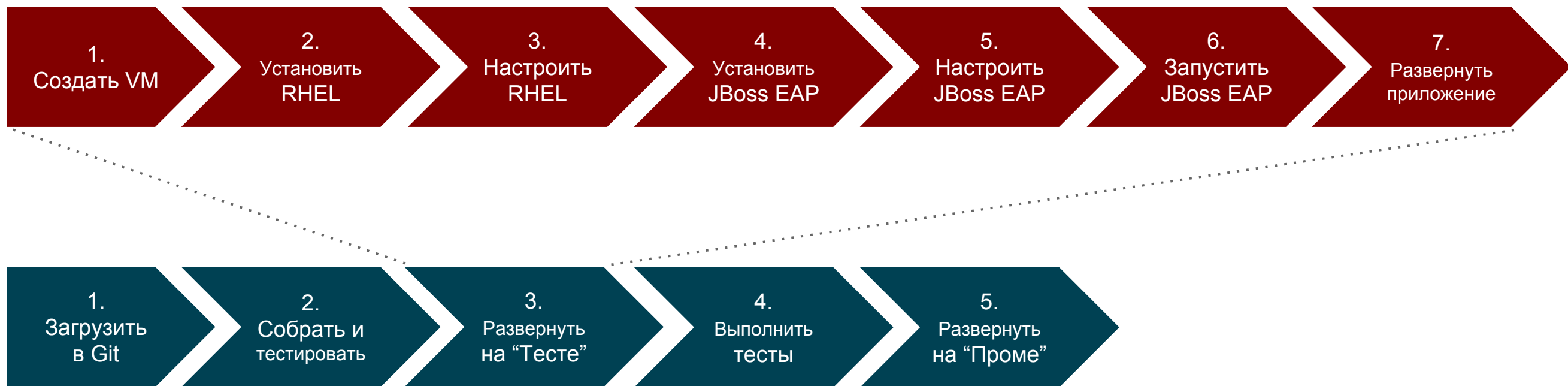
Традиционный подход

Используемые средства

A N S I B L E

by Red Hat®

RED HAT®
CLOUDFORMS



Организация DevOps

Подход OpenShift

Используемые средства



1.
ос new-app

1.
Загрузить
в Git

2.
Собрать и
тестировать

3.
Развернуть
на "Тесте"

4.
Выполнить
тесты

5.
Развернуть
на "Проме"

Что входит в подписку

Поставки OpenShift

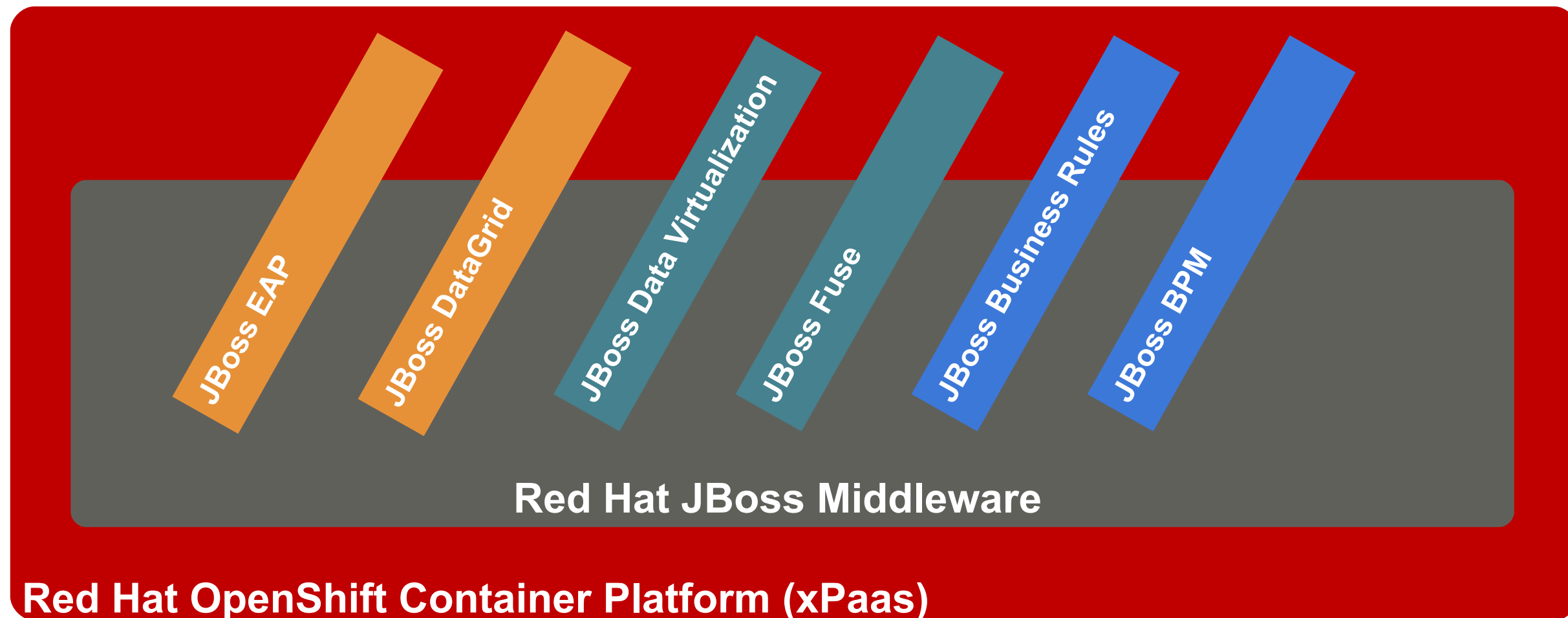
- OpenShift Origin
 - Разрабатывается и поддерживается сообществом
 - Большое количество разработчиков из Red Hat
- OpenShift Container Platform (ранее — Enterprise)
 - Основан на исходном коде OpenShift Origin
 - Находится на поддержке компании Red Hat
 - Развертывается на дата-центре предприятия
- OpenShift Online и Dedicated
 - Поддерживается компанией Red Hat
 - Развернут в публичном облаке (Amazon)
 - Dedicated — создается под заказчика

Что еще в подписке OpenShift

- OpenShift Container Platform предлагает в рамках подписки поддерживаемые шаблоны и образы
 - Ruby, PHP, Python, Perl, Apache HTTPD, Java (OpenJDK 1.8),...
 - PostgreSQL, MySQL, MariaDB, MongoDB,...
 - .Net Core CLR (1.0, 1.1, 2.0), JBoss WEB
 - Jenkins
- Дополнительно совершенно безвозмездно можно использовать образы из RHEL SCL (Software Collection Library)
 - RHEL
 - NodeJS
 - Ruby on Rails
 - Nginx

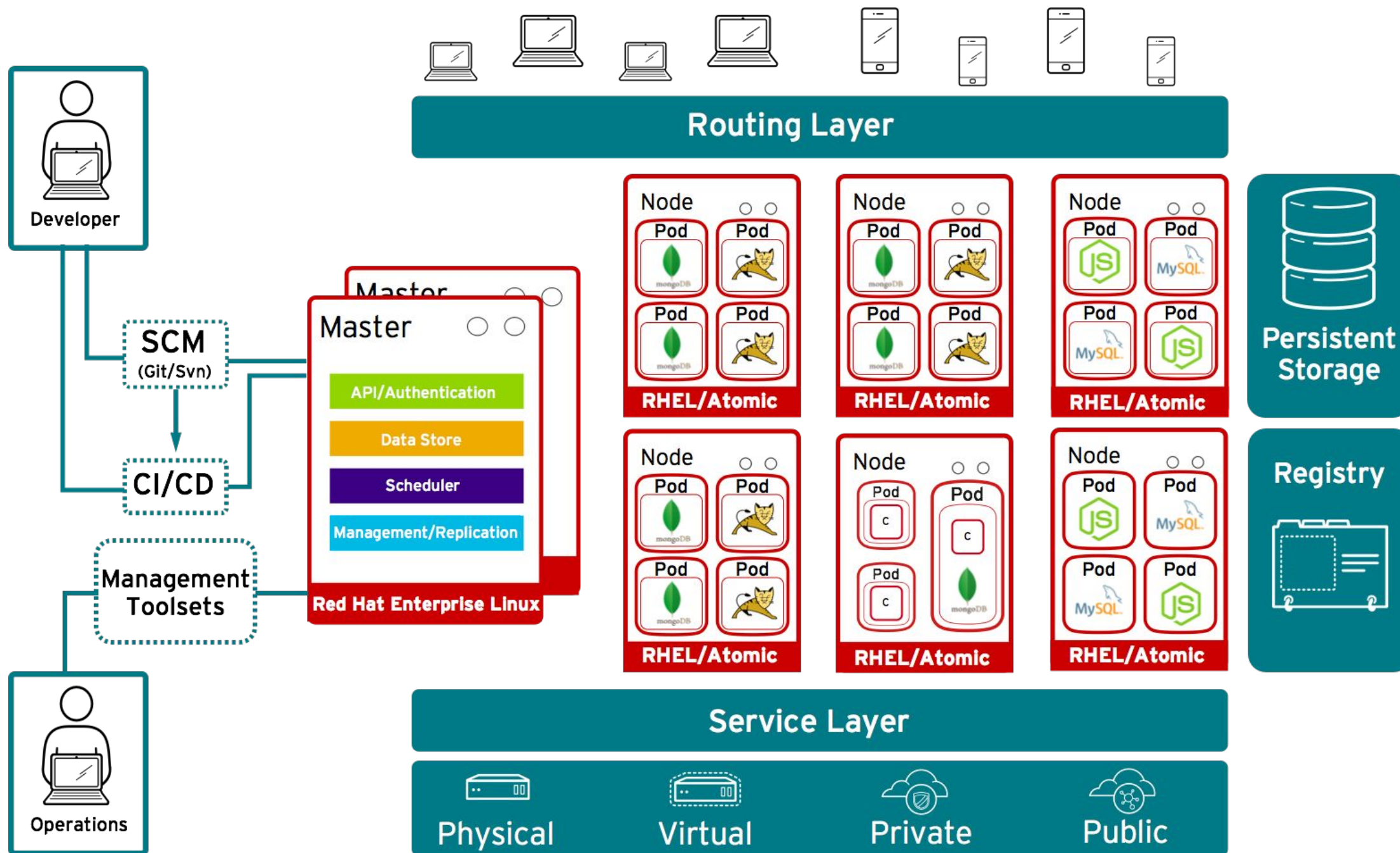
JBoss Middleware и OpenShift

OpenShift представляет JBoss Middleware в виде шаблонов



Как работает OpenShift v3

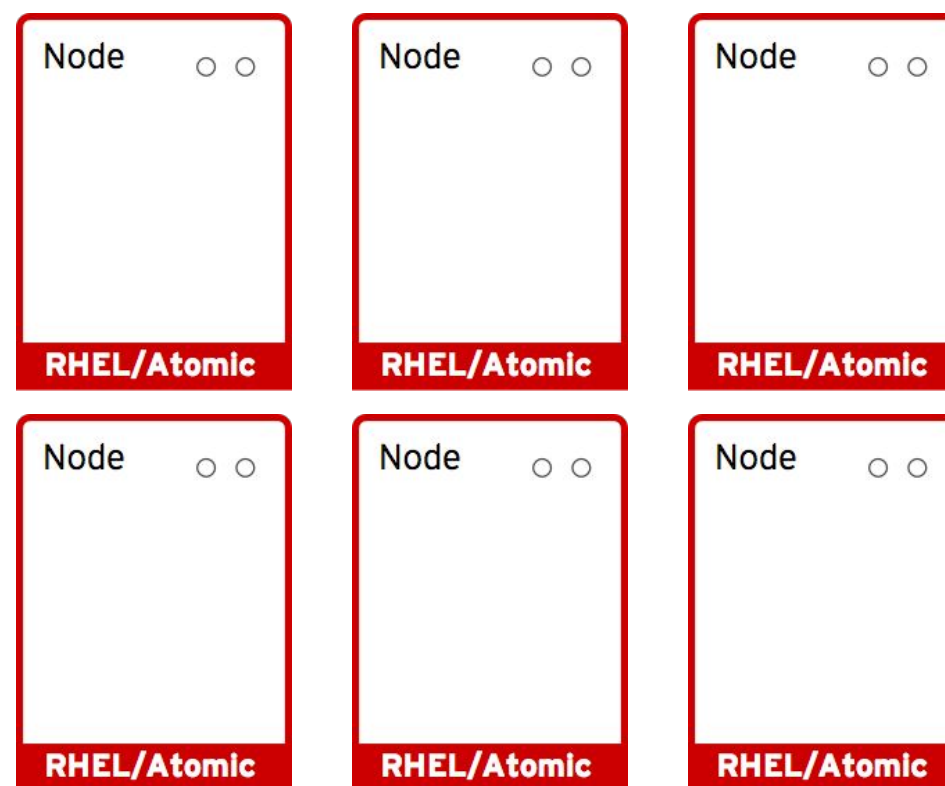
Компоненты и их взаимодействия



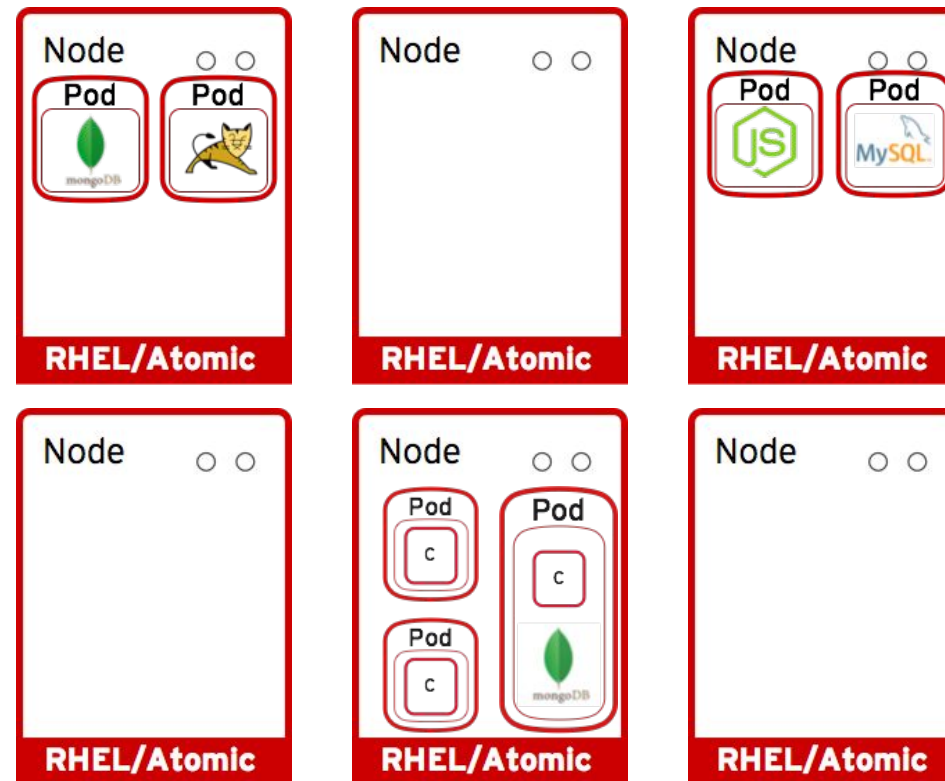
OpenShift работает на любой инфраструктуре



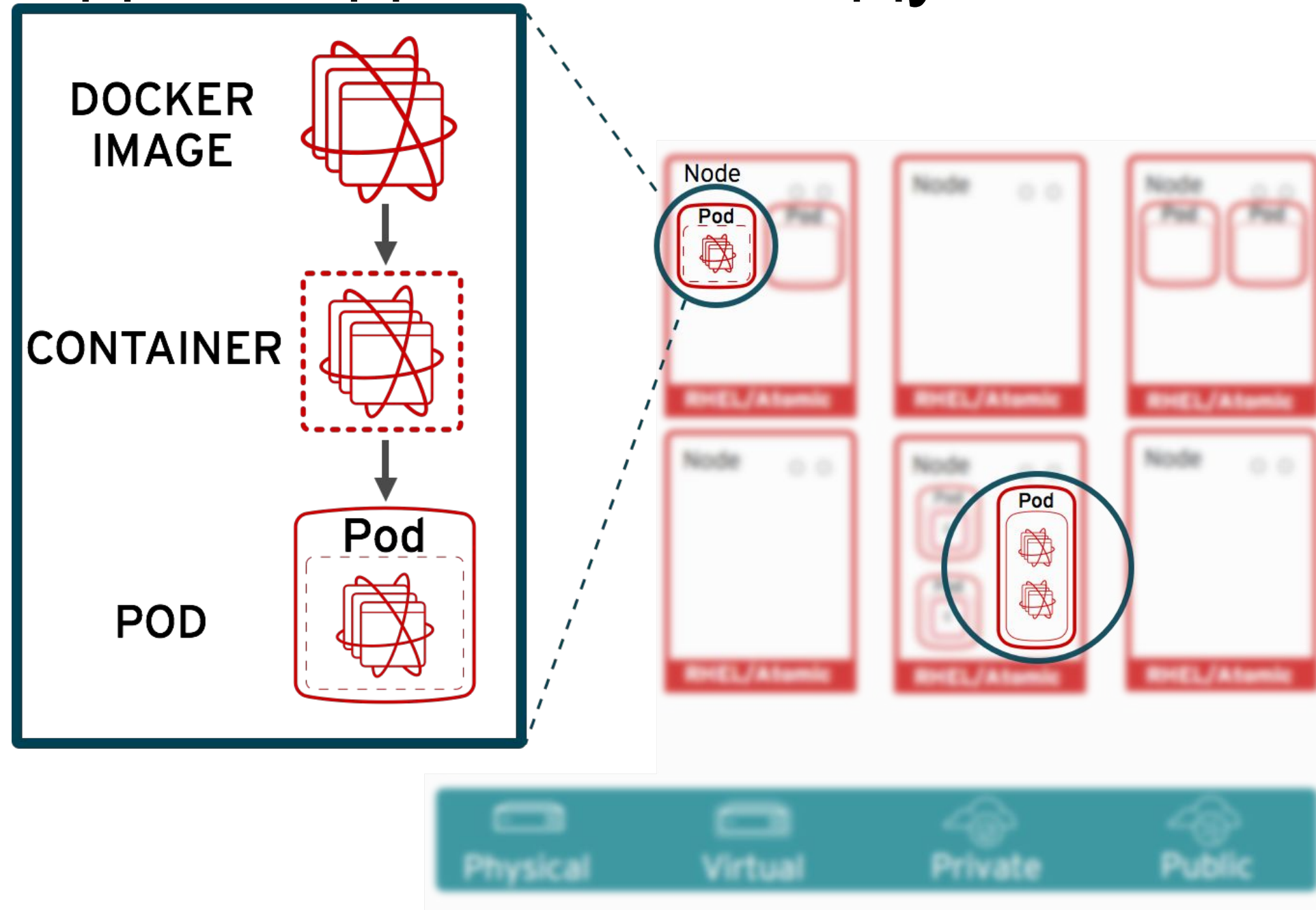
Узлы - экземпляры RHEL, где работают приложения



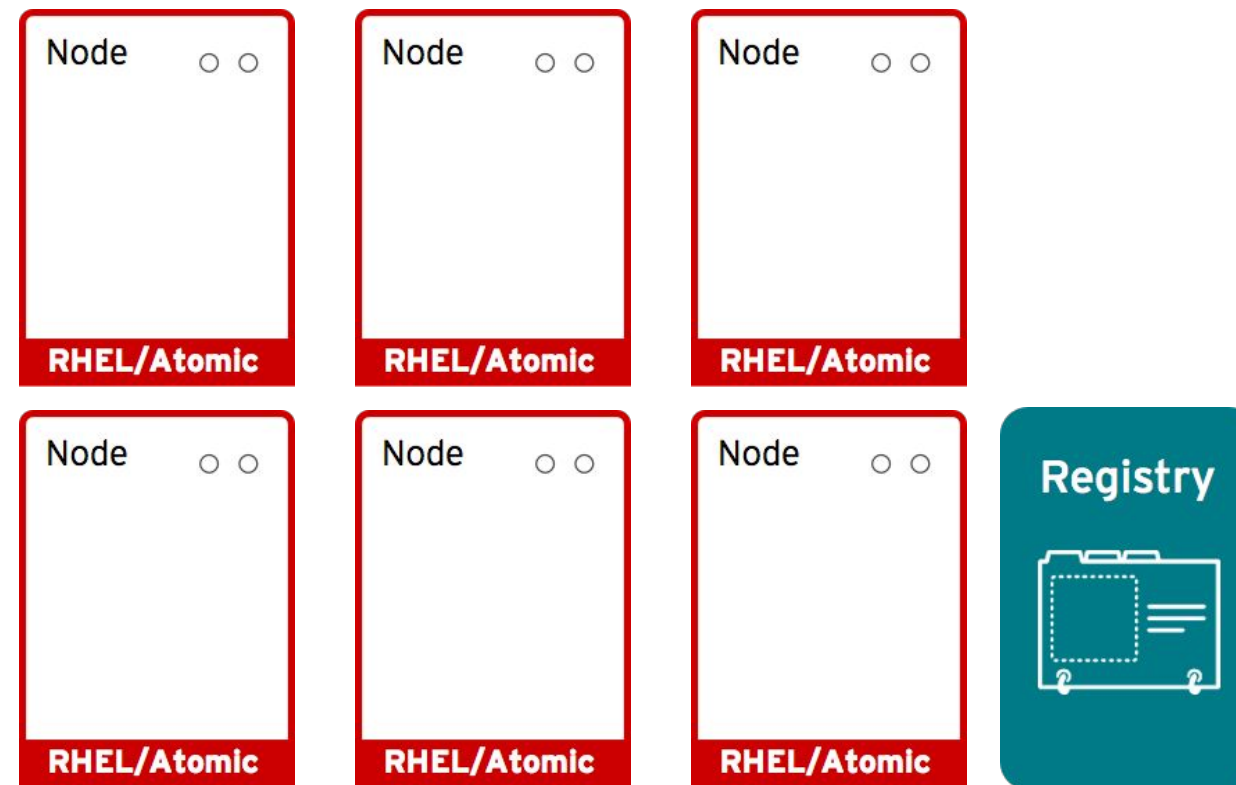
Приложения работают в контейнерах на каждом узле



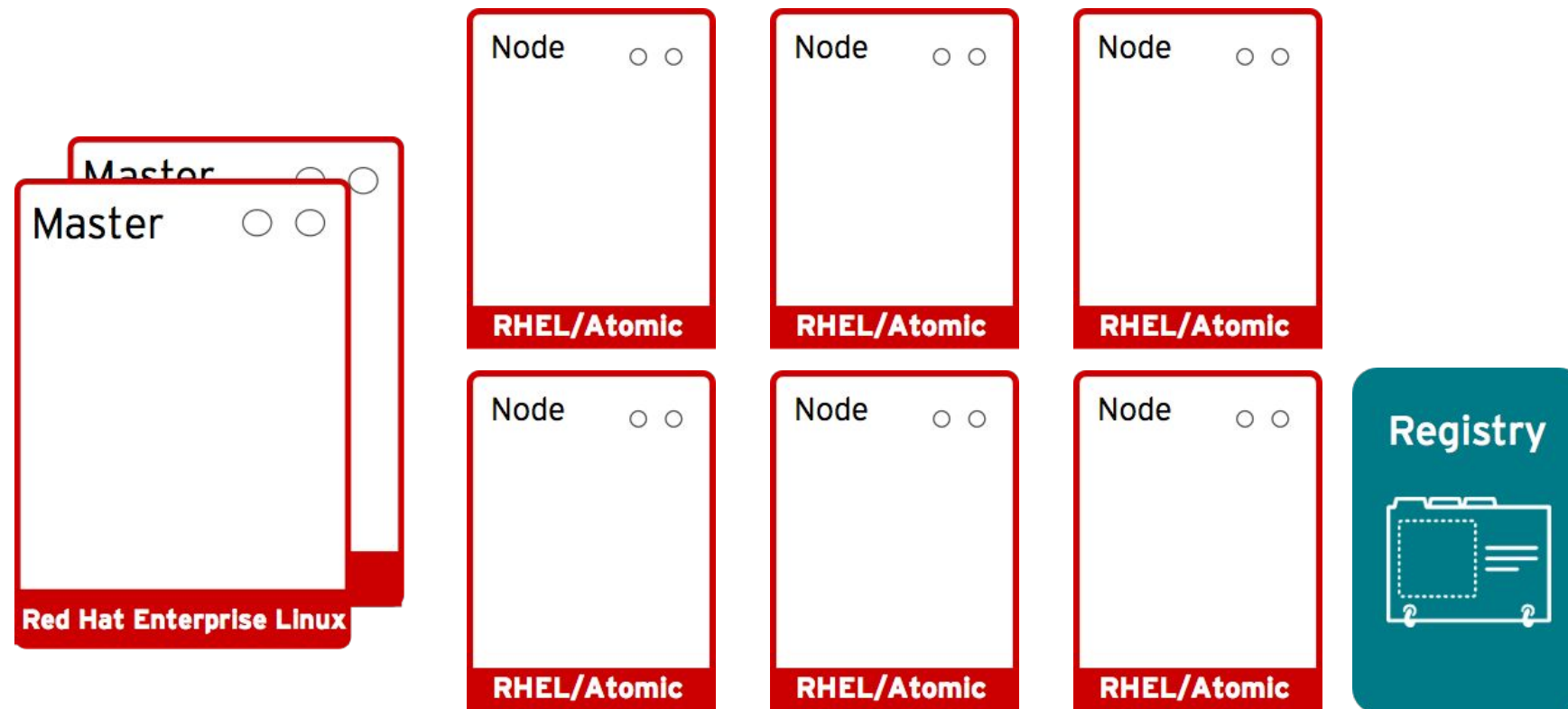
POD выполняет один или более контейнеров, как один неделимый модуль



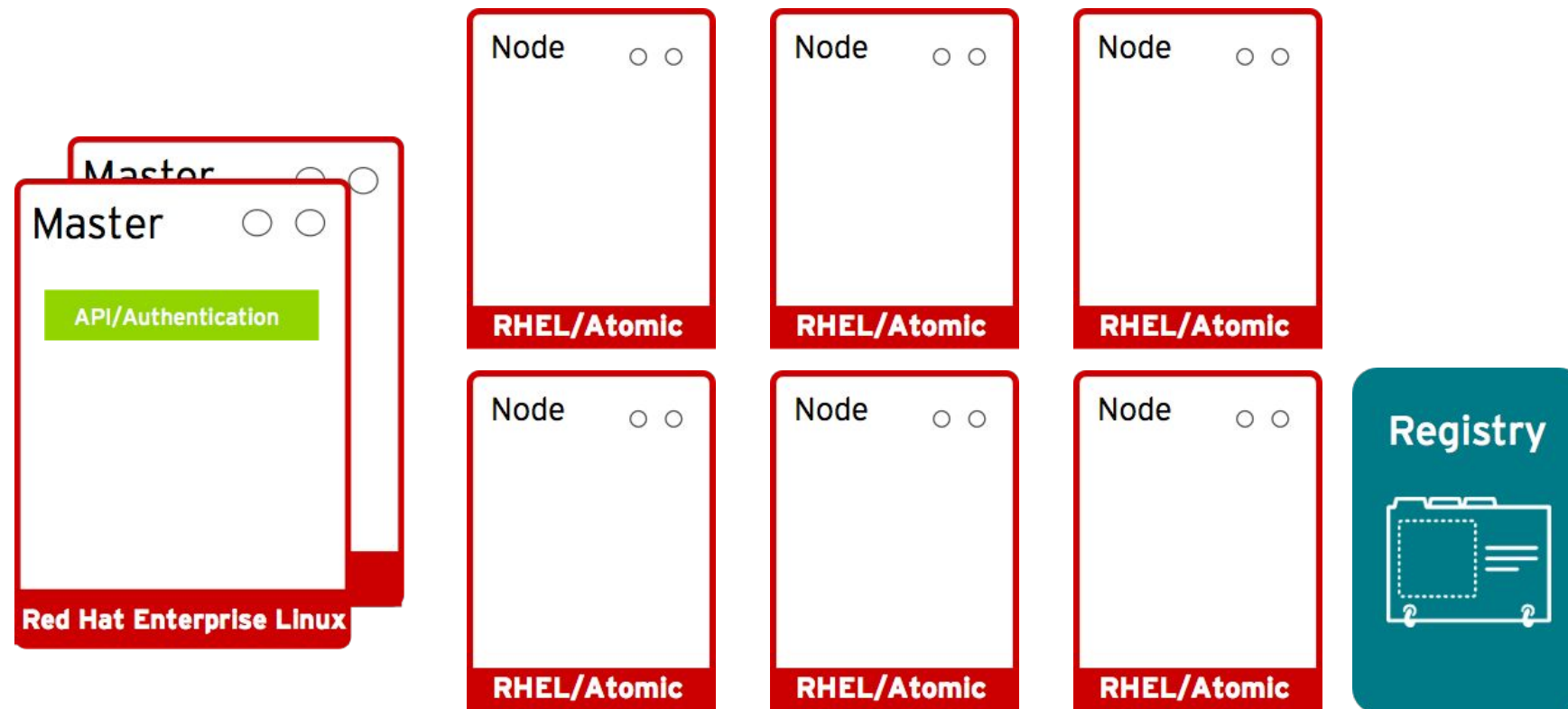
Образы приложений хранятся в Реестре



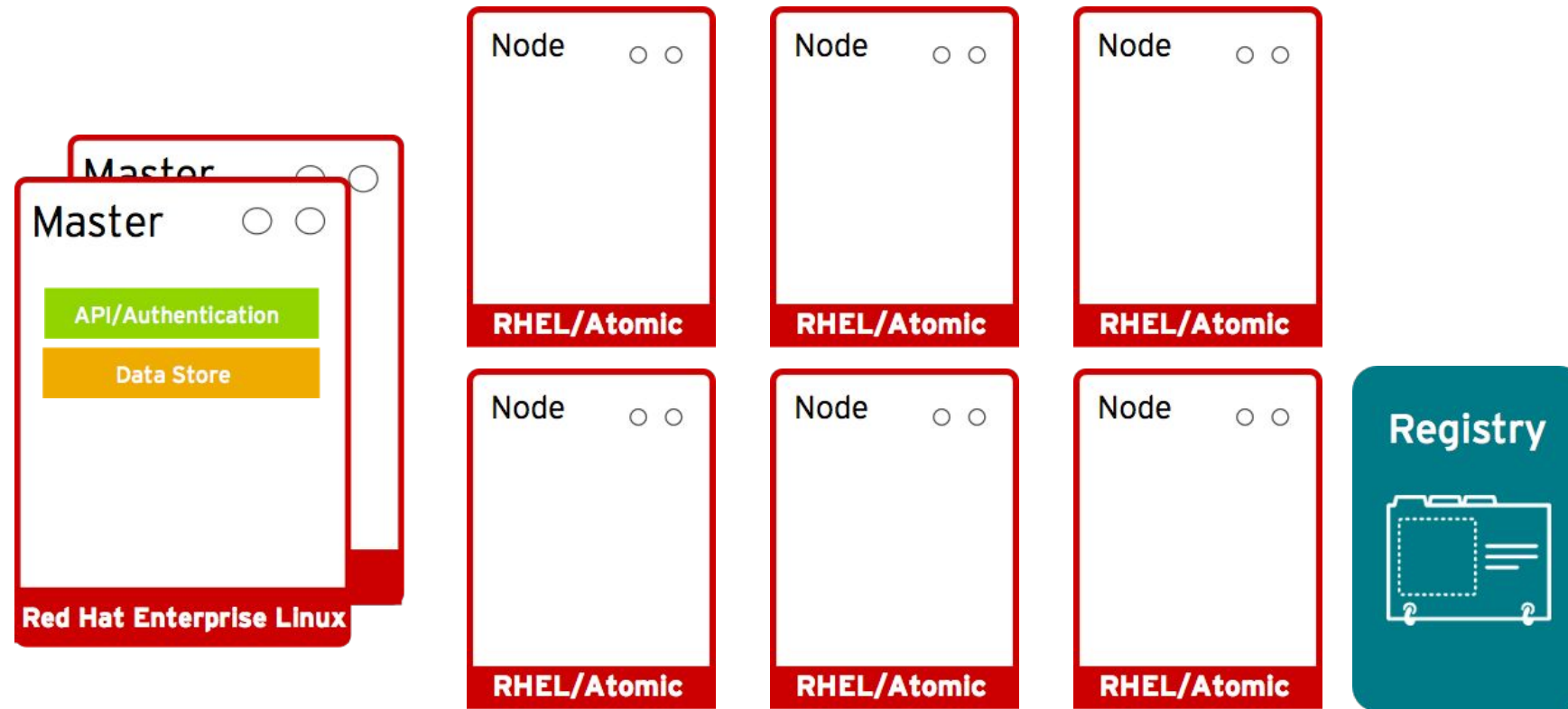
Мастер используют Kubernetes для управления узлами и приложениями



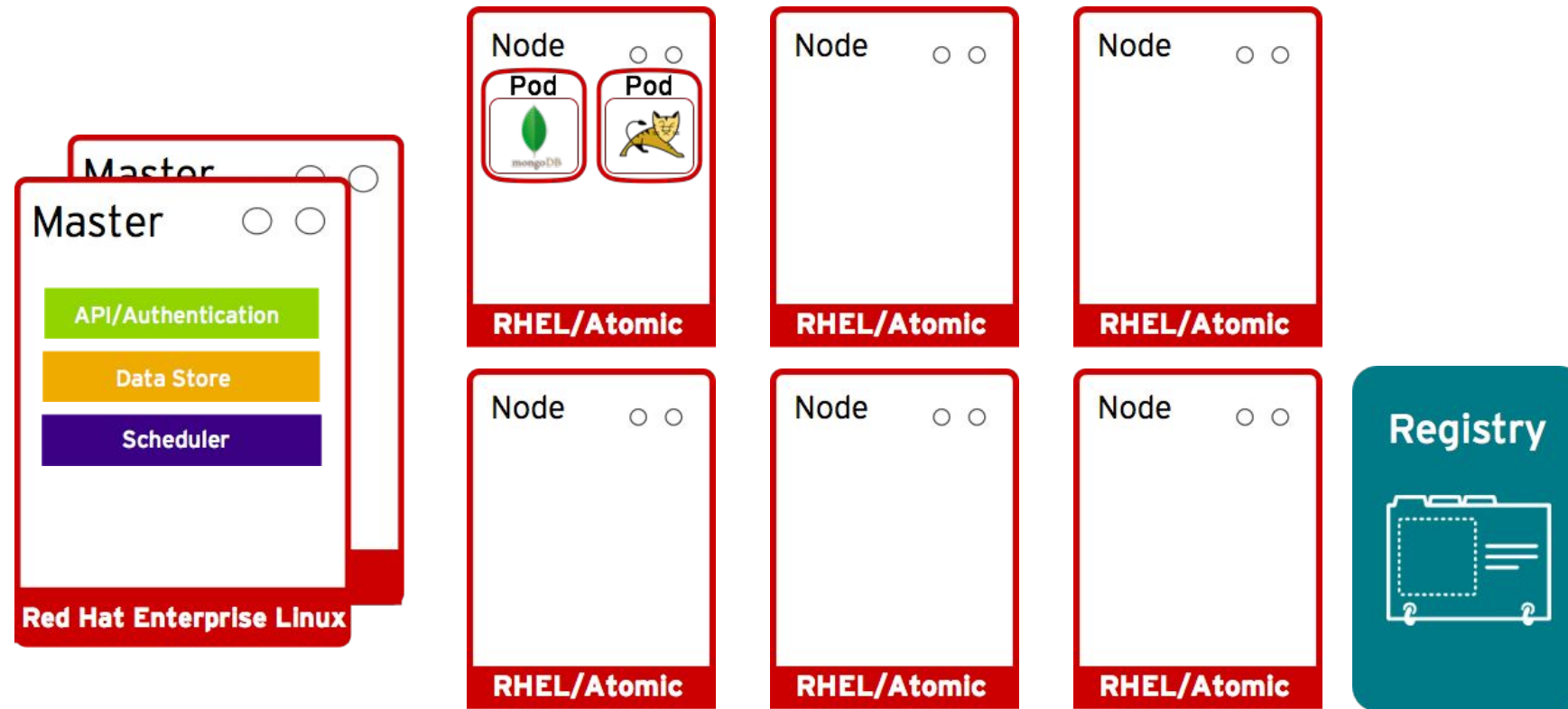
Мастер обеспечивают аутентификацию



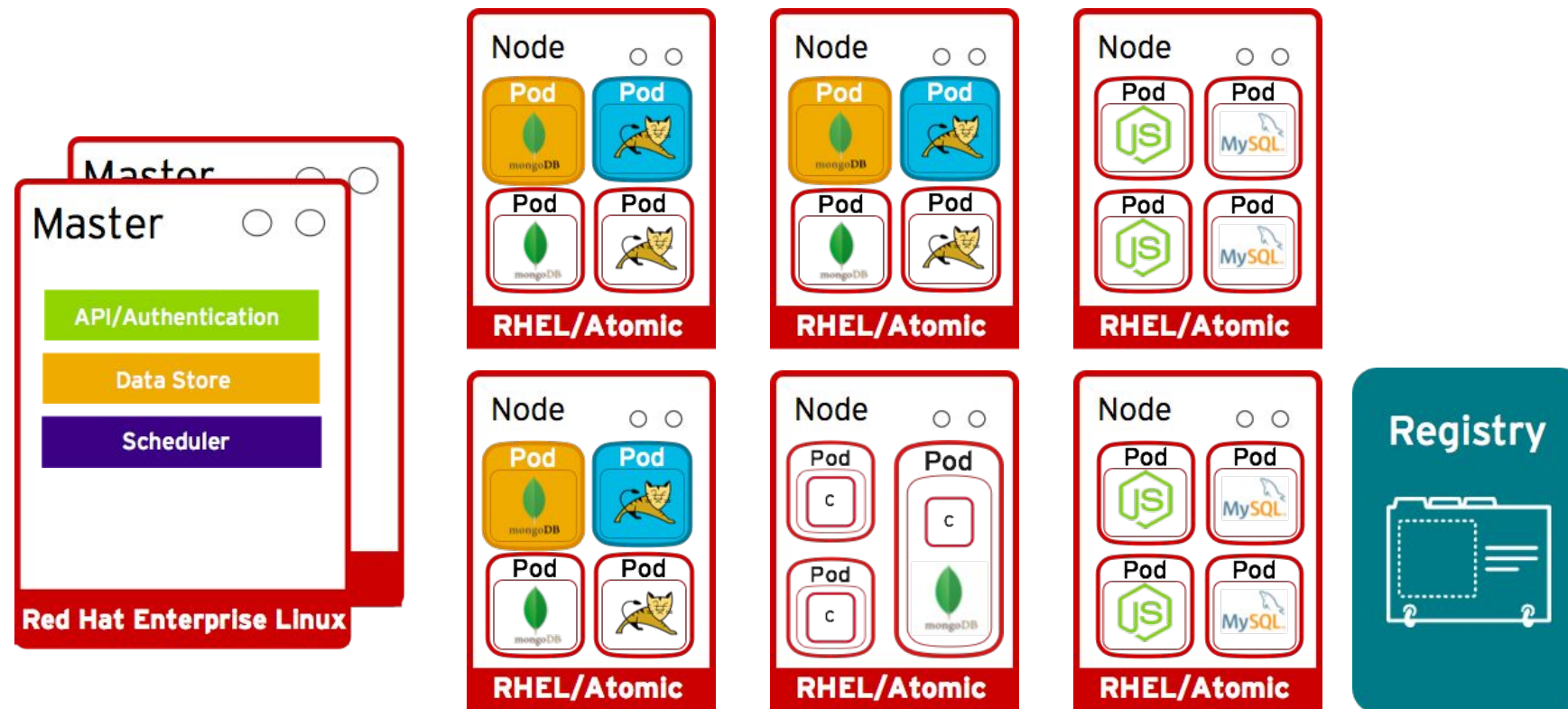
Мастер использует ETCD - хранилище метаданных в виде ключ-значение



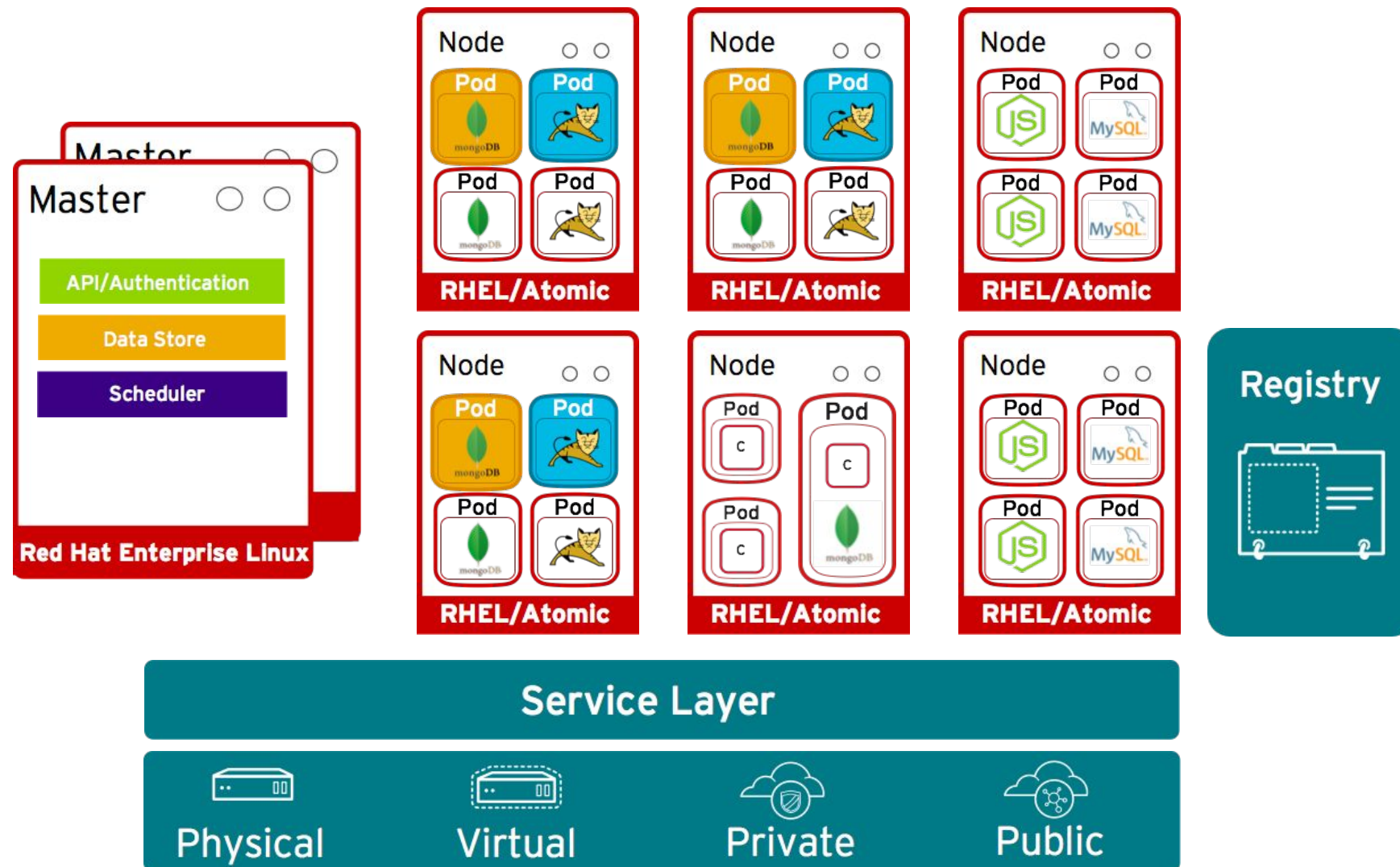
Мастер содержит планировщик задач - размещает POD'ы на узлах



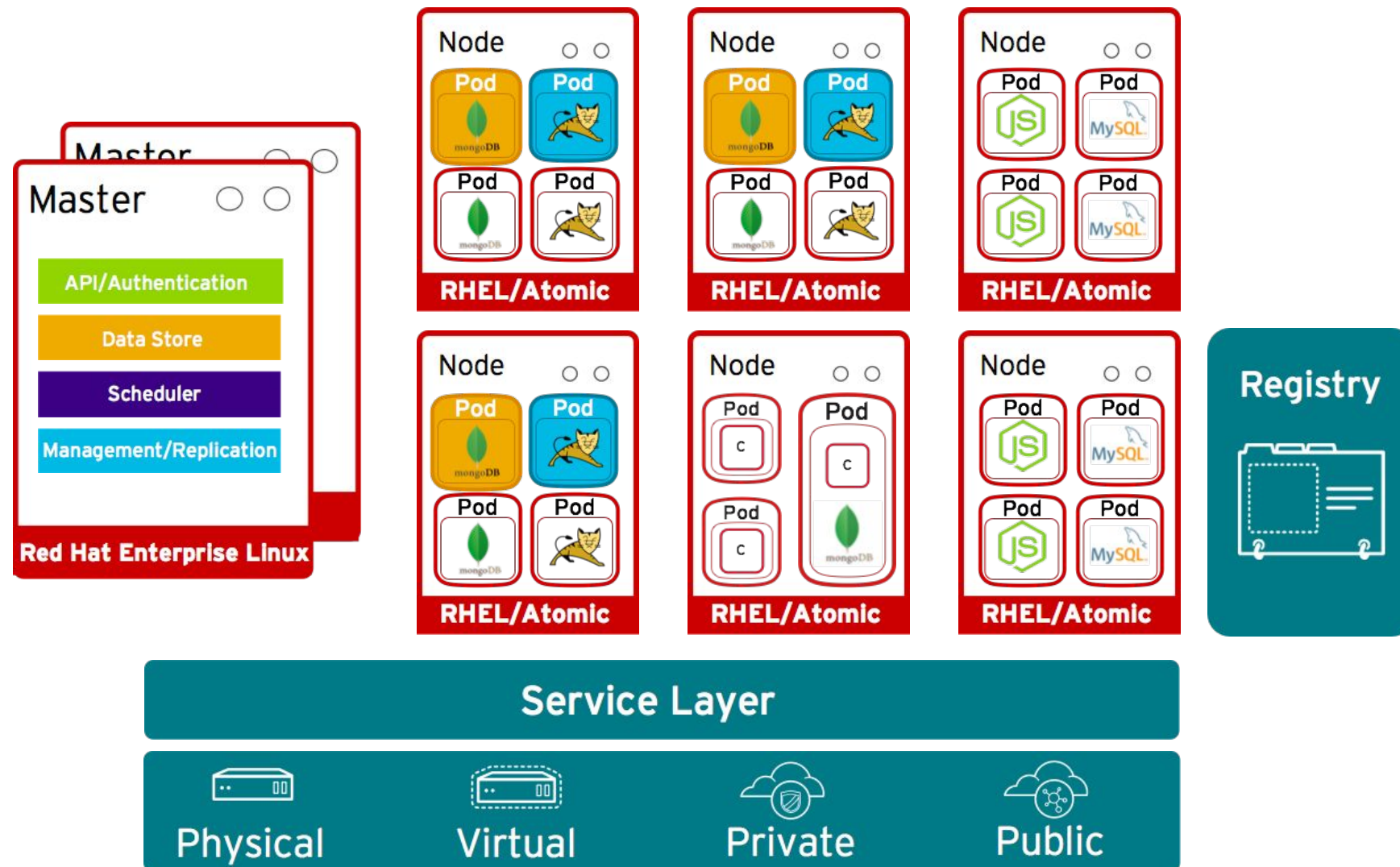
Планировщик использует политики размещения



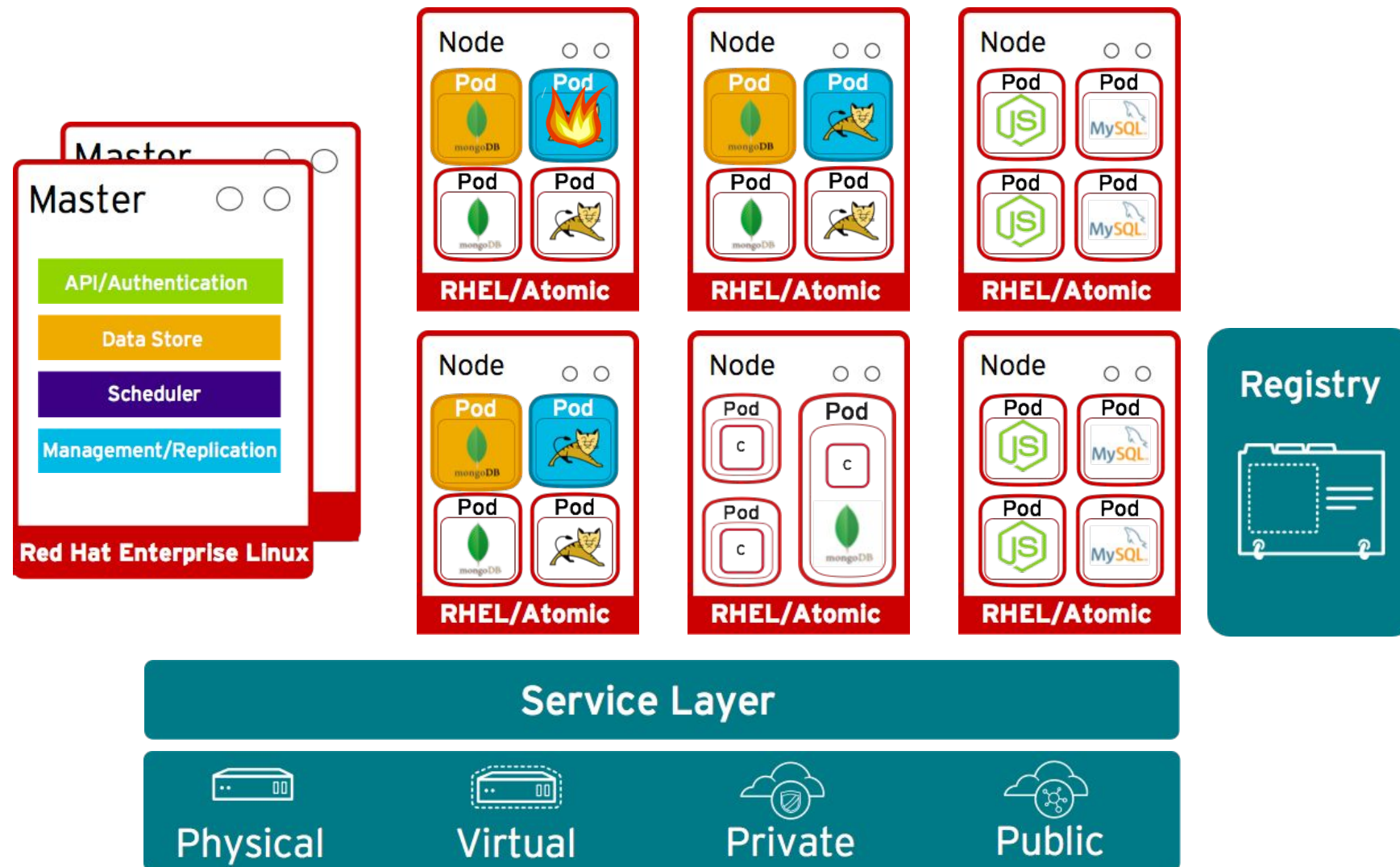
Сервисы абстрагируют доступ к POD



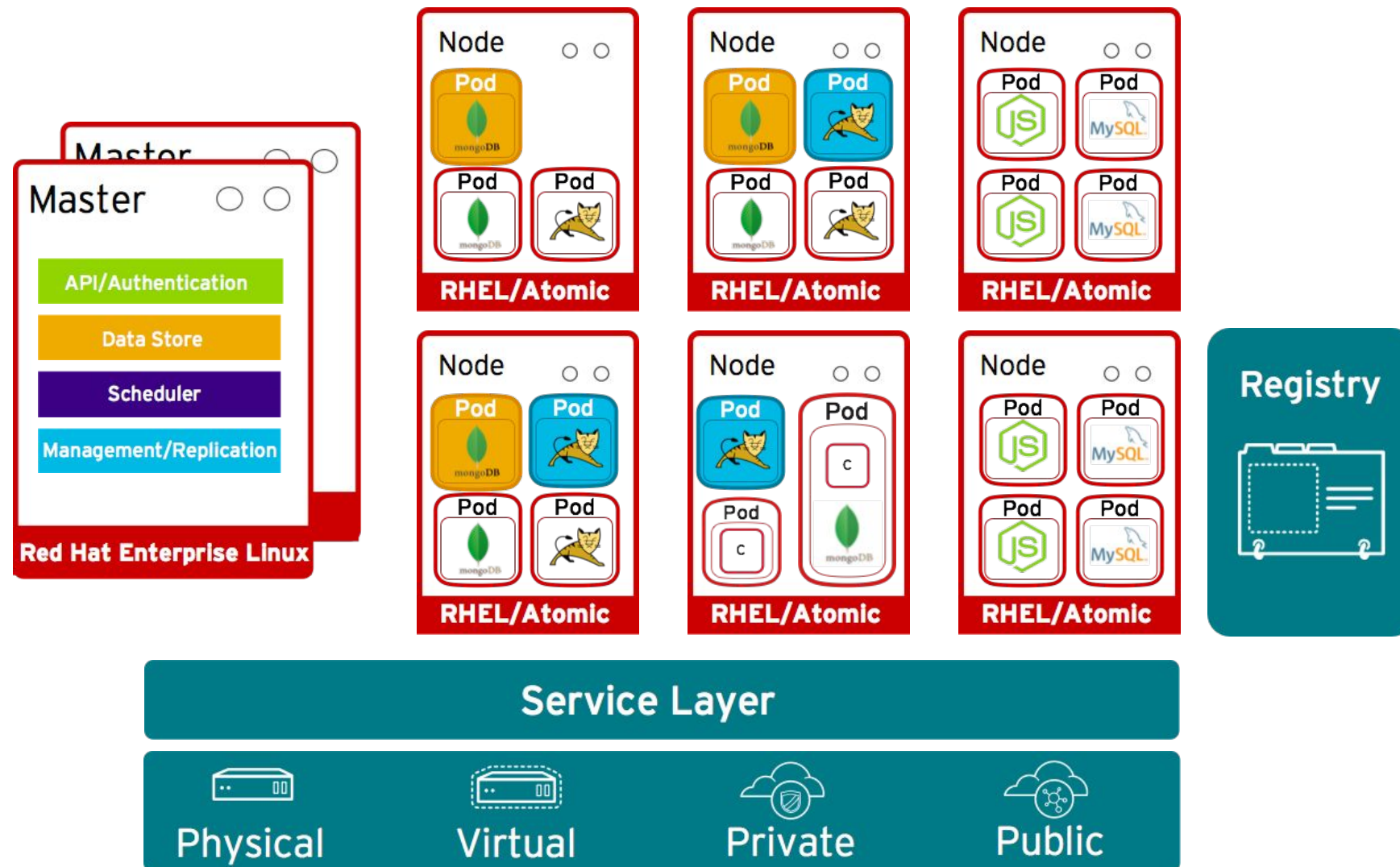
Replication Controller управляет жизненным циклом POD



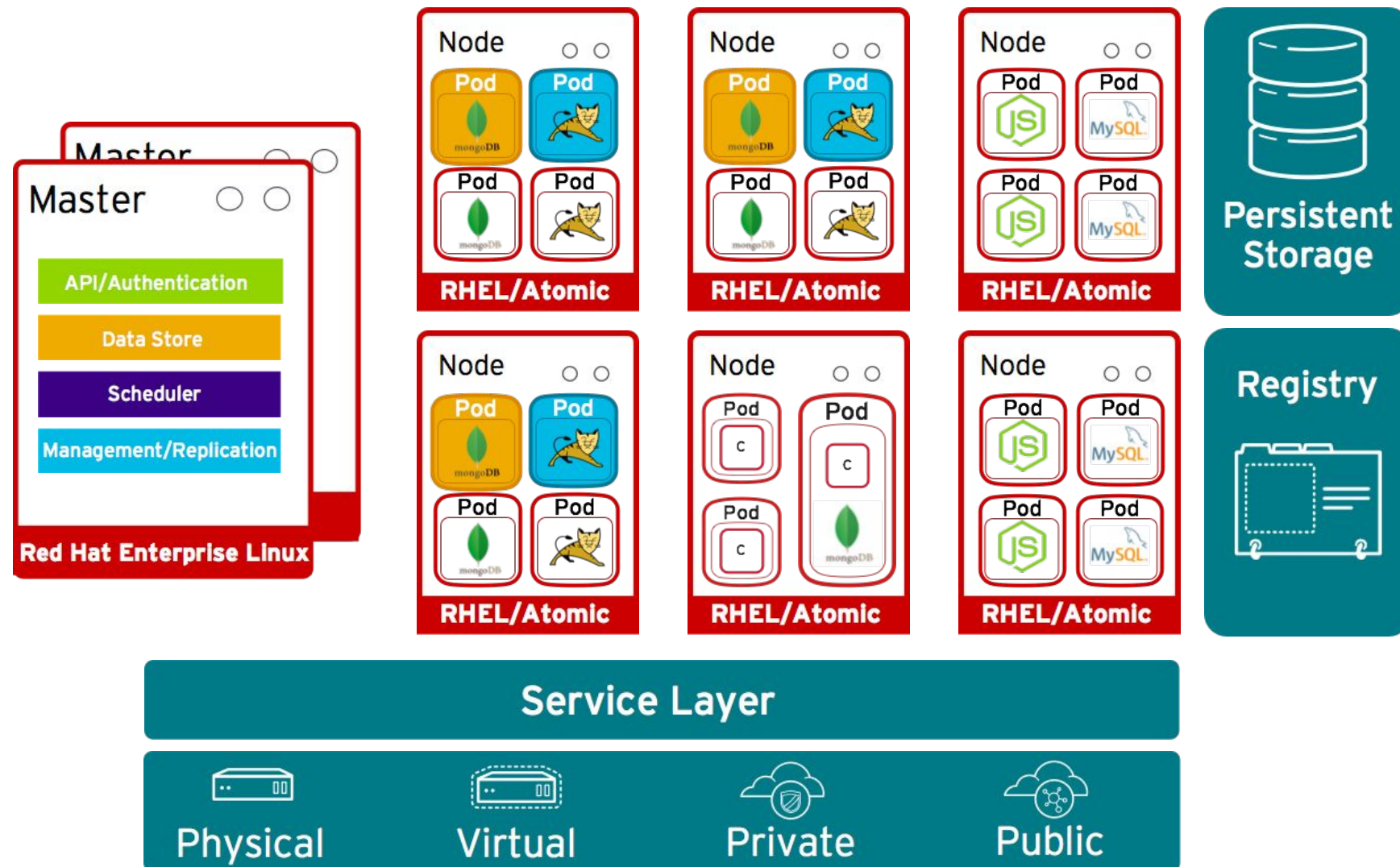
Если POD перестает функционировать



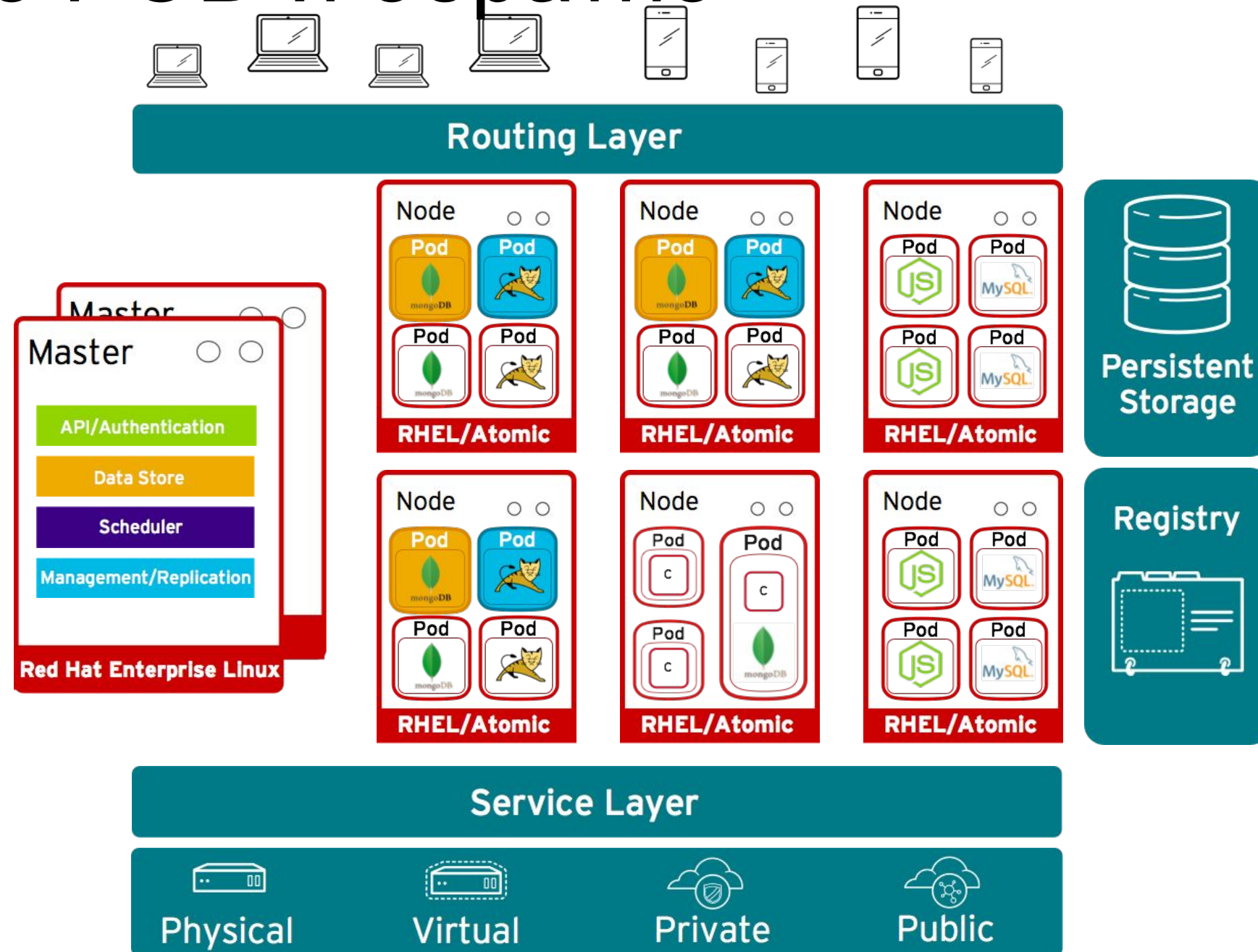
OpenShift автоматически разворачивает новый POD, восстановив работоспособность



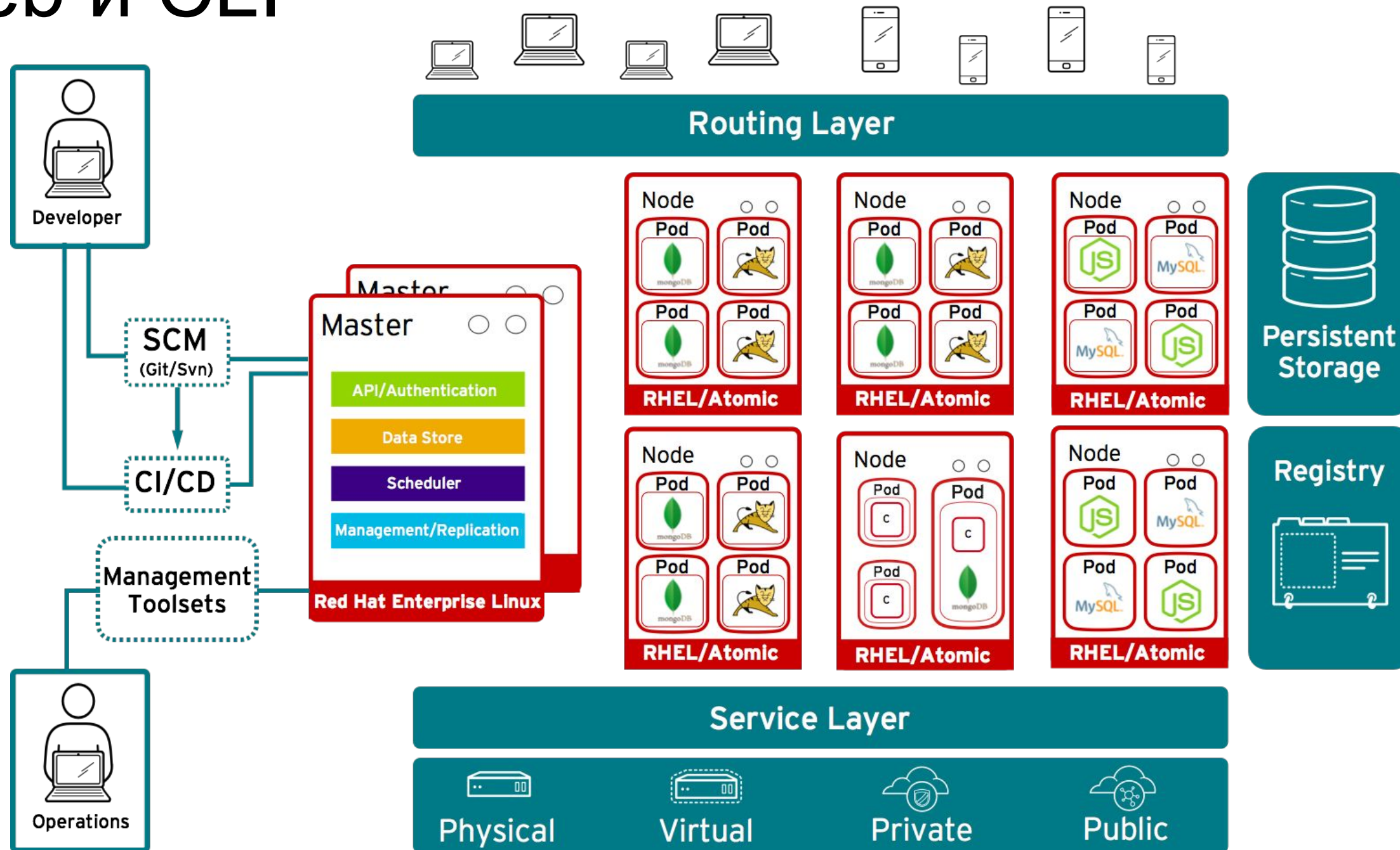
POD может иметь доступ к СХД для хранения результатов работы



Сервис Маршрутизации доставляет внешние запросы до POD и обратно

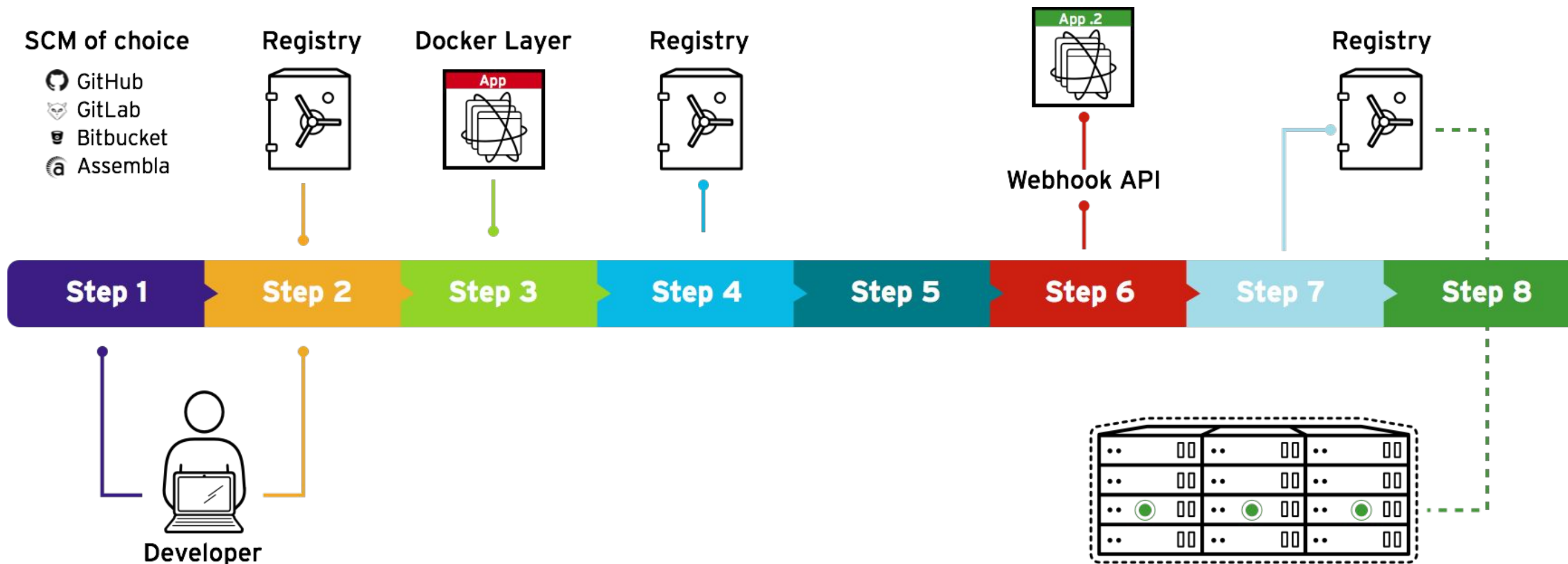


Разработчики имеют доступ к OpenShift через Web и CLI







Простейший процесс CI/CD

Встроенный CI/CD процесс



Разработчик указывает репозиторий Git





SCM of choice

-  GitHub
-  GitLab
-  Bitbucket
-  Assembla

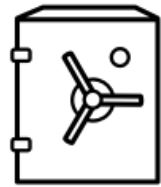


Разработчик выбирает образ из реестра

SCM of choice

-  GitHub
-  GitLab
-  Bitbucket
-  Assembla

Registry

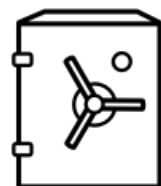


Новый слой добавляется к образу

SCM of choice

- GitHub
- GitLab
- Bitbucket
- Assembla

Registry



Docker Layer



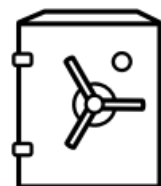
Developer

Образ помещается в реестр

SCM of choice

- GitHub
- GitLab
- Bitbucket
- Assembla

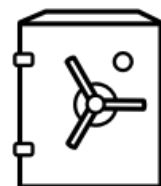
Registry



Docker Layer



Registry

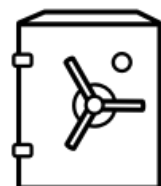


Образ планируется и разворачивается

SCM of choice

- GitHub
- GitLab
- Bitbucket
- Assembla

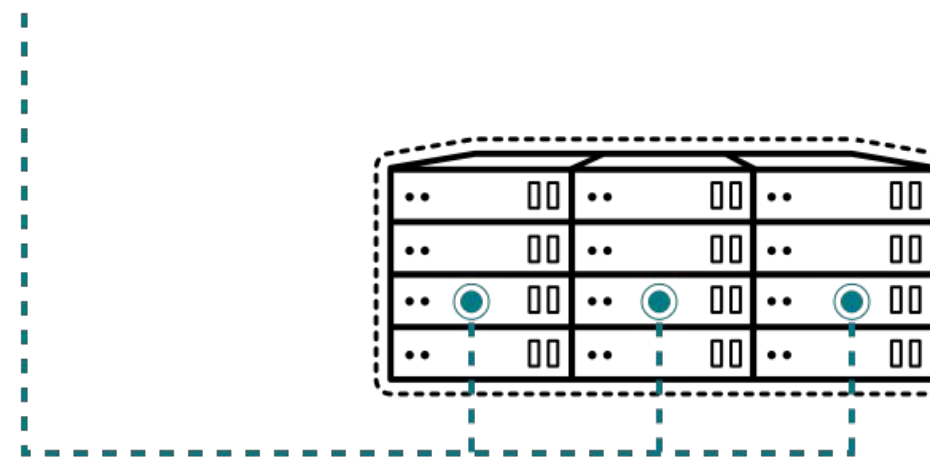
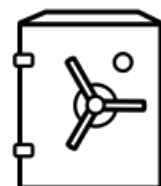
Registry



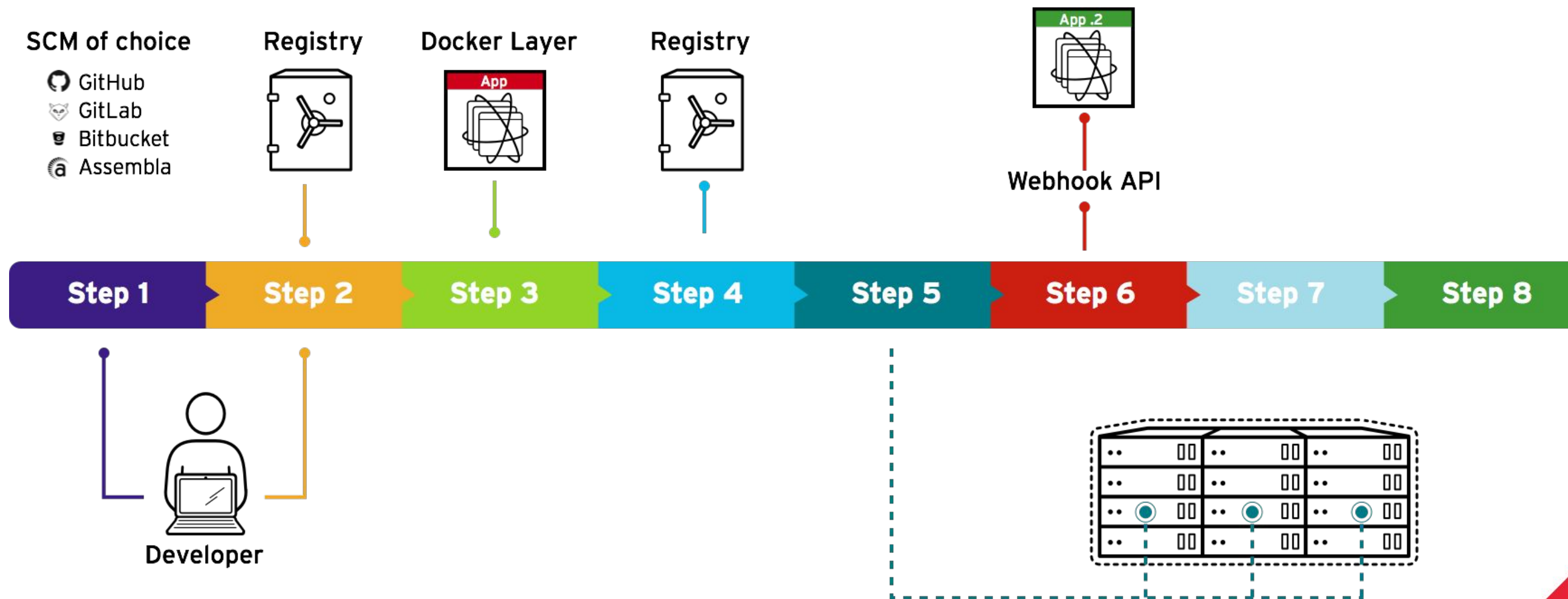
Docker Layer



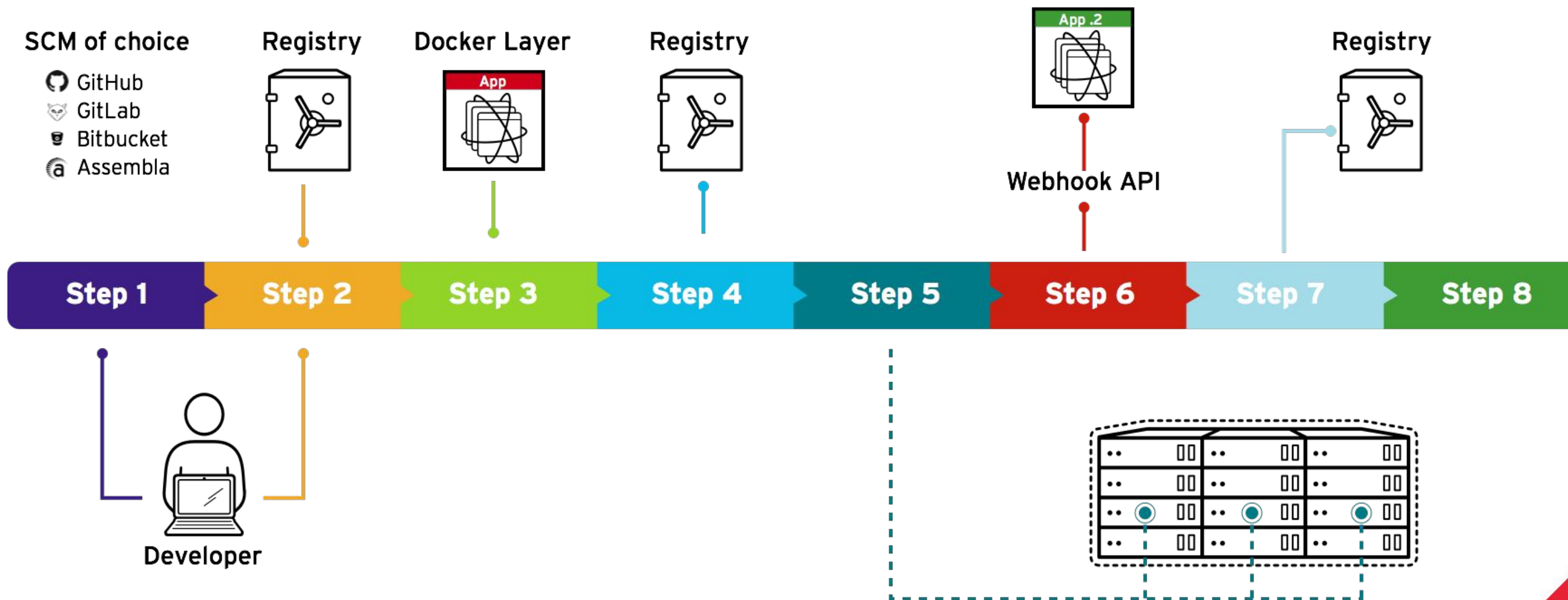
Registry



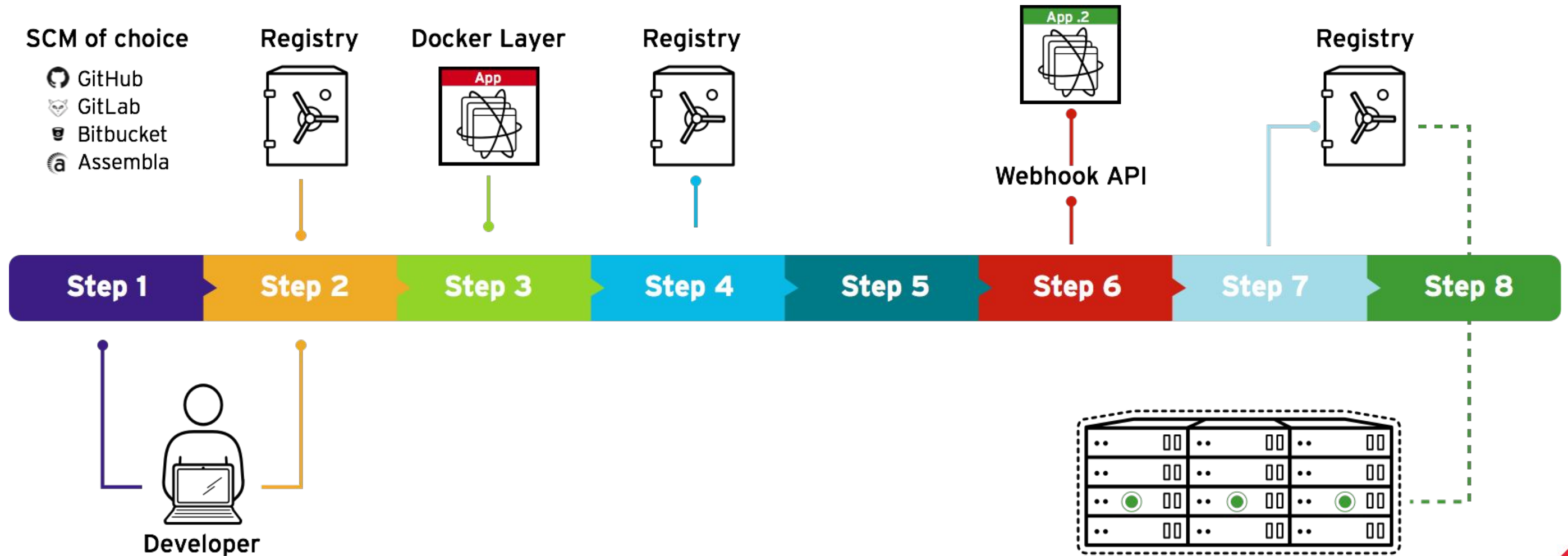
Разработчик указывает Web-Hook



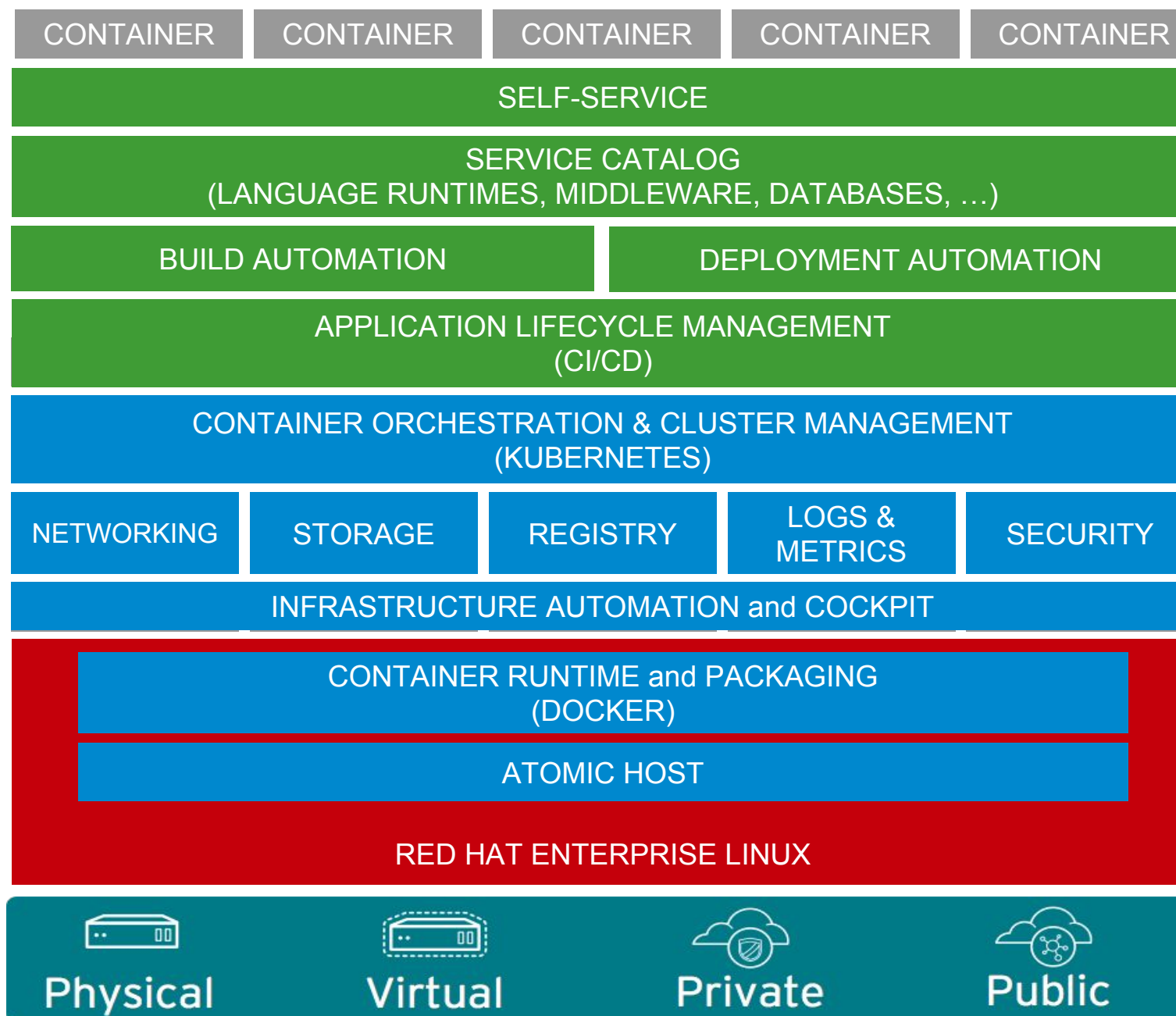
Новый образ загружается опять в реестр



Новый образ разворачивается как обновление



Что такое OpenShift



ССЫЛКИ

- <https://docs.openshift.com/container-platform/3.9/welcome/index.html>
- https://docs.openshift.com/container-platform/3.9/scaling_performance/index.html
- [Канал на Youtube - OpenShift](#)
- [Книга DevOps with OpenShift](#)
- [Книга OpenShift for Developers](#)
- [Книга Deploying to OpenShift](#)
- <http://blog.openshift.com>
- [Интерактивный учебник по OpenShift](#)
- Лабораторные работы
 - <http://redhatgov.io>
 - <https://github.com/osevg/workshopper-content>
 - <https://github.com/RedHatWorkshops/openshiftv3-workshop>
 - <https://github.com/openshift-roadshow/cloud-native-guides>
 - <https://github.com/mann1mal/aws-loft-2017-container-lab>



redhat.®

Наши заказчики

PAYCHEX[®]

T Systems



FICO[™]

accenture



Что такое OpenShift

Технологический аспект

- Это сложная сбалансированная технологическая система, развивающаяся по эволюционному пути
- Как превратить карету в тыкву?
 - Рассматривать OpenShift, как классическую систему
 - Разрозненный набор сервисов
 - Загрузка приложений не готовых к облаку
 - Отказаться от синергетического эффекта
 - Замена встроенного Atomic Registry
 - Замена OpenShift SDN третье сторонними модулями
 - Загрузка готовых образов, минуя сборки