

RAINBOND应用市场解读

好雨交付工程师-郭逊

大纲

1.应用市场功能

2.分享粒度控制

3.如何制作一个rainbond应用

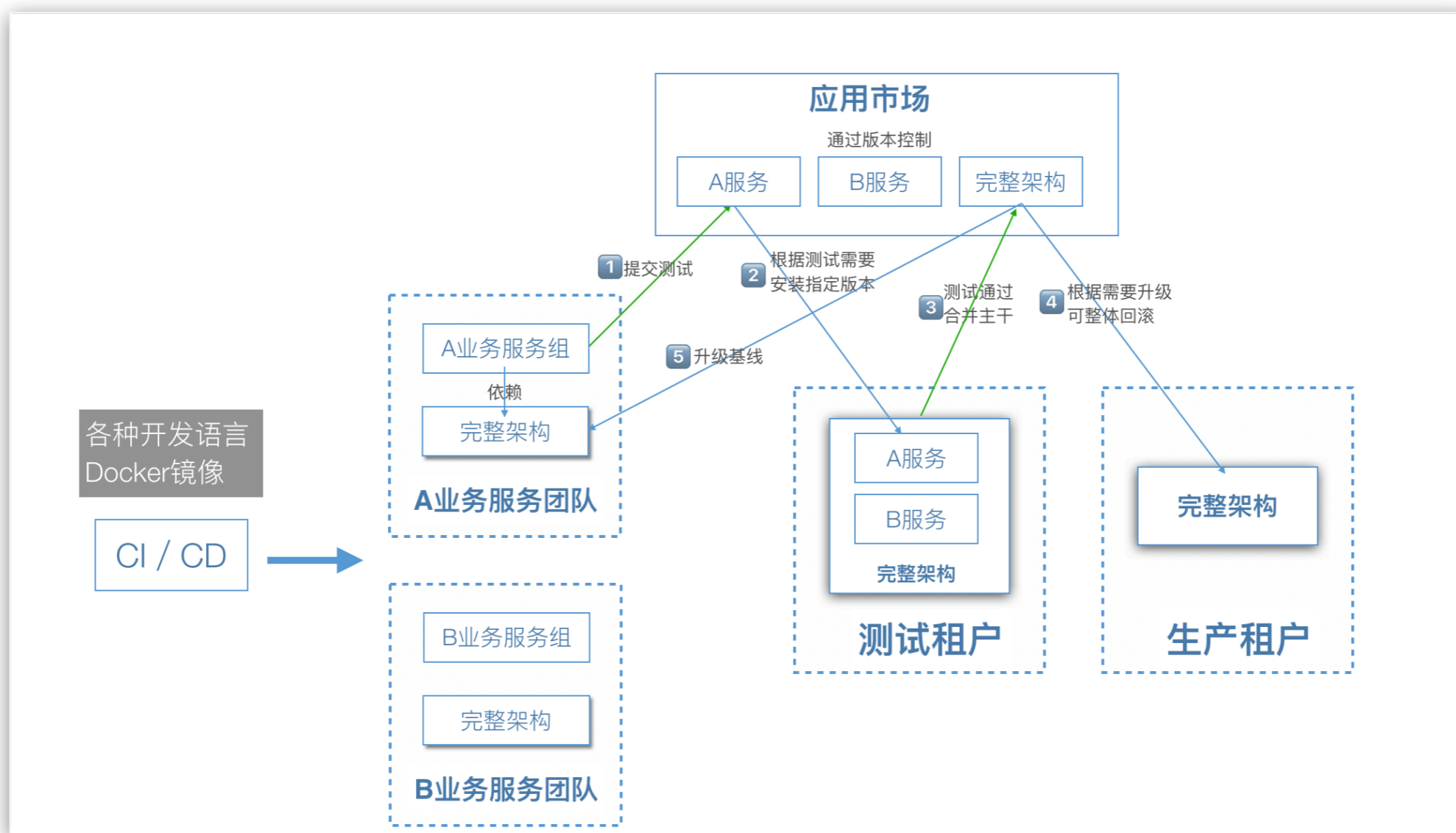
4.演示示例

5.rainbond应用与helm应用的关系

1.应用市场功能

1.1 基于应用市场的开发、测试、生产pipeline

1.2 利于企业应用交付



2. 分享粒度控制

1.1 团队

1.2 公司

1.3 云应用市场

3. 如何制作一个RAINBOND应用

1.1 容器化应用要点

1.2 数据库自动初始化

1.3 依赖关系建立

容器化应用要点

1 不要在容器中存储数据

由于容器是一次性的，当容器被停止、销毁或替换时，应用在容器中存储的数据同样会被销毁。如果应用需要存储数据，需存储在持久化存储中。并且保证数据与代码在目录层面分离。

2 不要在一个容器中运行多个服务

每个容器中只能部署运行一个应用，不要部署多个应用到一个容器中，比如将WEB应用和数据库同时运行到同个容器中。同样的道理，不要将多个不相关的进程运行在同一个容器中。比如在一个容器内同时运行业务进程和日志收集进程。保持容器内容单一，对于管理是非常友好的。

3 清除不必要的包和文件

容器的一个显著特点是秒级启动，一旦制作了一个超大的镜像将难以分发。在镜像中应该确保仅有运行应用/进程的必需的文件和库存在，其他不必要的包、文件等建议应用打包前进行清除。

容器化应用要点

4 使用环境变量对应用进行配置

环境变量应该以传参的形式传入容器，而不是直接写死在容器中，需要的环境变量可写入系统集成文档中。

5 使用非root用户运行

镜像应该使用**USER**命令来指令容器以一个非**ROOT**用户来运行。

6 不要依赖IP地址

重新启动容器，其内部的**IP**有可能会发生变化。所以业务应该避免对**IP**的依赖。可以考虑用域名的方式。而在**RAINBOND**上运行，可以用依赖关系规避。

7 支持自动化升级

当新版本的镜像生成，应该添加判断逻辑，对持久化数据中的特定文件进行检测。来触发升级动作，更新持久化数据中需要改动的地方。

数据库自动初始化

1. 关键目录 /docker-entrypoint-initdb.d

DOCKER HUB官方提供的**MYSQL**镜像默认提供了一个功能——自动初始化。简单说起来，就是提供了一个特殊的目录 **/DOCKER-ENTRYPOINT-INITDB.D**，**MYSQL**初次启动的时候会扫描这个目录，如果存在脚本文件，则会自动执行。借此功能，即可完成自动初始化。

2. Dockerfile关键代码

```
FROM percona  
COPY init.sql /docker-entrypoint-initdb.d
```

3. 意义

自动初始化数据库，可以保证分享出去的应用再安装之后，数据库不必处理，整个业务系统可以直接工作。同样有很多其他**DOCKER HUB**官方提供的镜像都具有此功能。比如**MONGO**、**SOLR**等等，需要大家去探索，仔细阅读官方说明。

建立依赖关系

1. 连接信息讲解

平台上的A应用，在开启对内服务后，会自动在依赖-连接信息处生成规定连接地址、端口的环境变量。在有其他B应用依赖了A应用后，这组变量会自动注入到B应用环境中去，B应用可以通过读取环境变量，来获取A应用的通信地址。

2. 支持拓扑图编辑

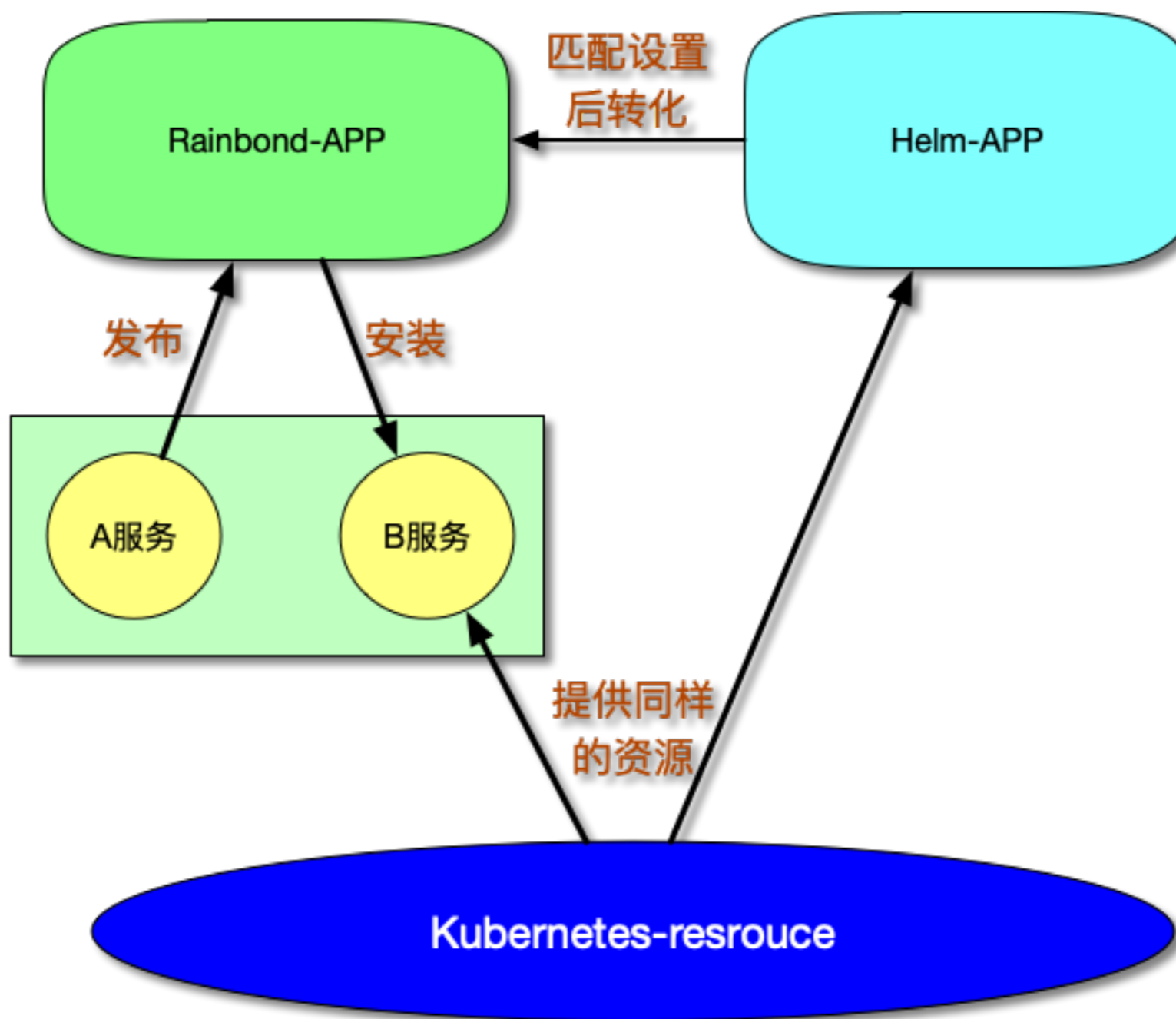
4. 演示示例

OpenSNS微社区

+

MySQL

5. RAINBOND应用与HELM应用的关系



我是郭逊，
好雨交付工程师，
我为交付质量代言 😊



好雨交付工程师-郭逊