

Open Anolis



Open Infrastructure
FOUNDATION

CLOUD NATIVE INFRASTRUCTURES MEETUP

活动交流群



关注OpenAnolis.org



Nydus容器镜像服务及实践

巴德 易观

2020-12-05

01

Nydus镜像服务介绍



02

Nydus在阿里云的应用实践



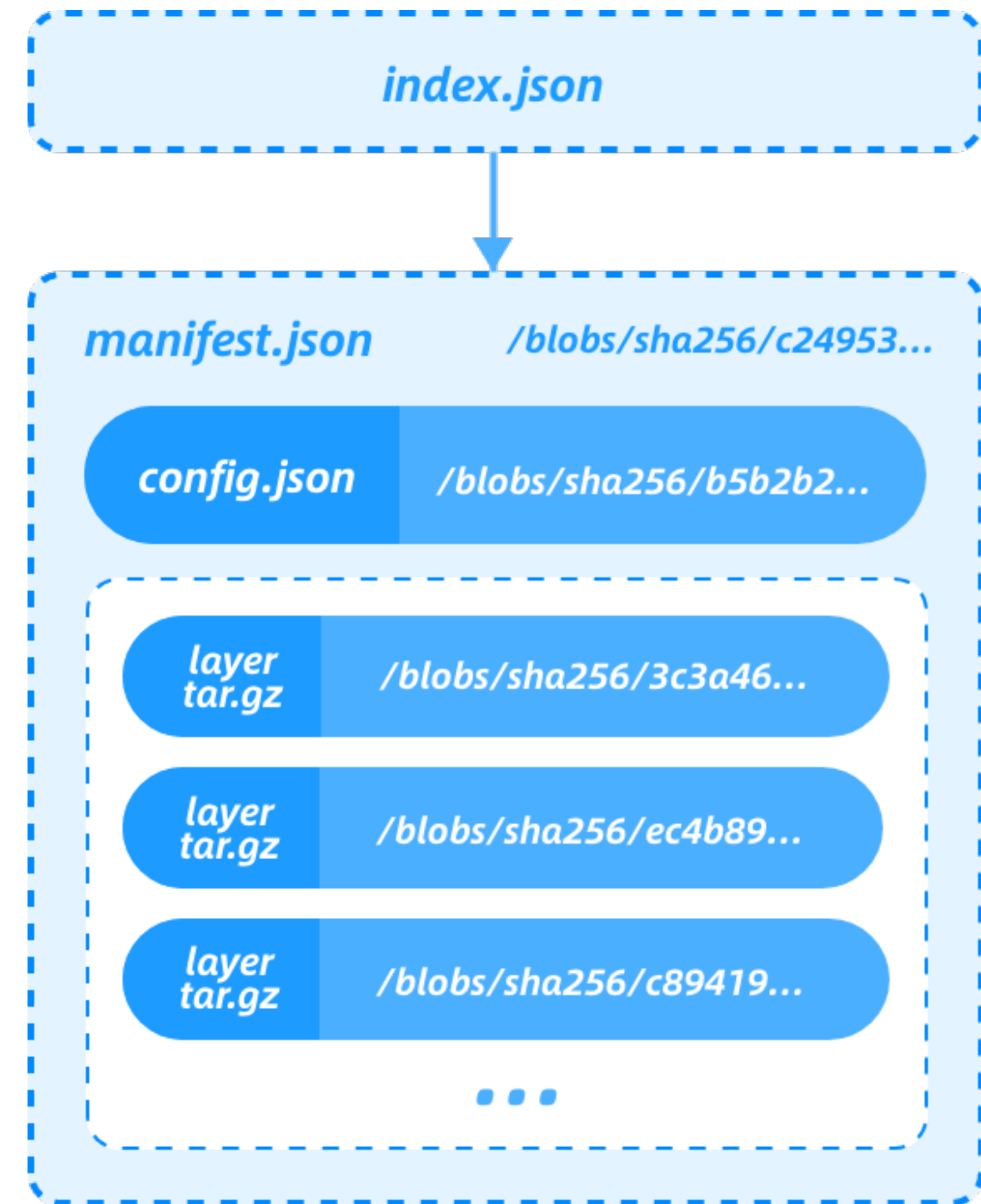
01

Nydus镜像服务介绍



OCI Image Drawbacks

- Before starting a container, images must be pulled and decompressed to the local filesystem
- Deduplication at layer level is less efficient
- <https://hackmd.io/@cyphar/ociv2-brainstorm>



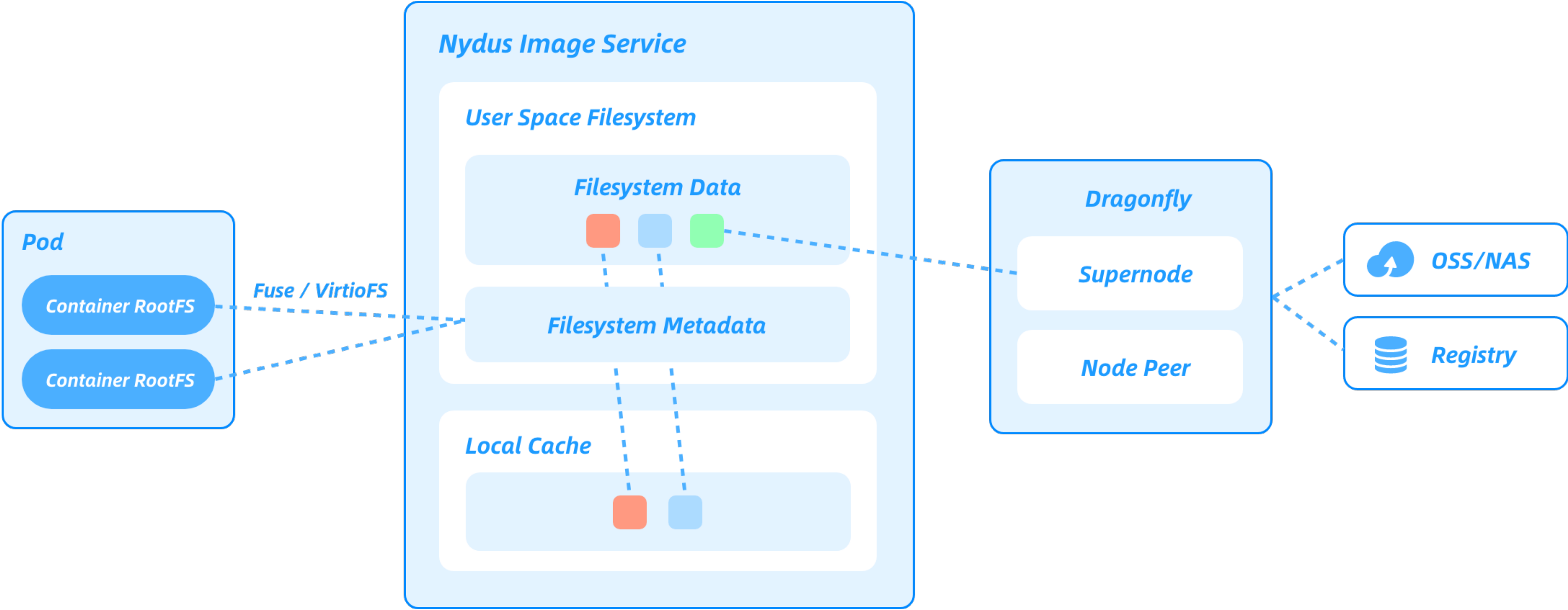
Community Prior Works

- P2P based fast image distribution
 - Dragonfly -- CNCF
 - Kraken -- Uber
- File level local cache
 - CernVM-fs -- CERN
- Rely on storage layer snapshot capability
 - Slacker -- University of Wisconsin
- Hack gzip stream, keeping layering layout
 - CRFS -- Golang Build Team
- Split metadata/data, breaking layering layout
 - Filegrain -- NTT
 - Umoci -- SuSE

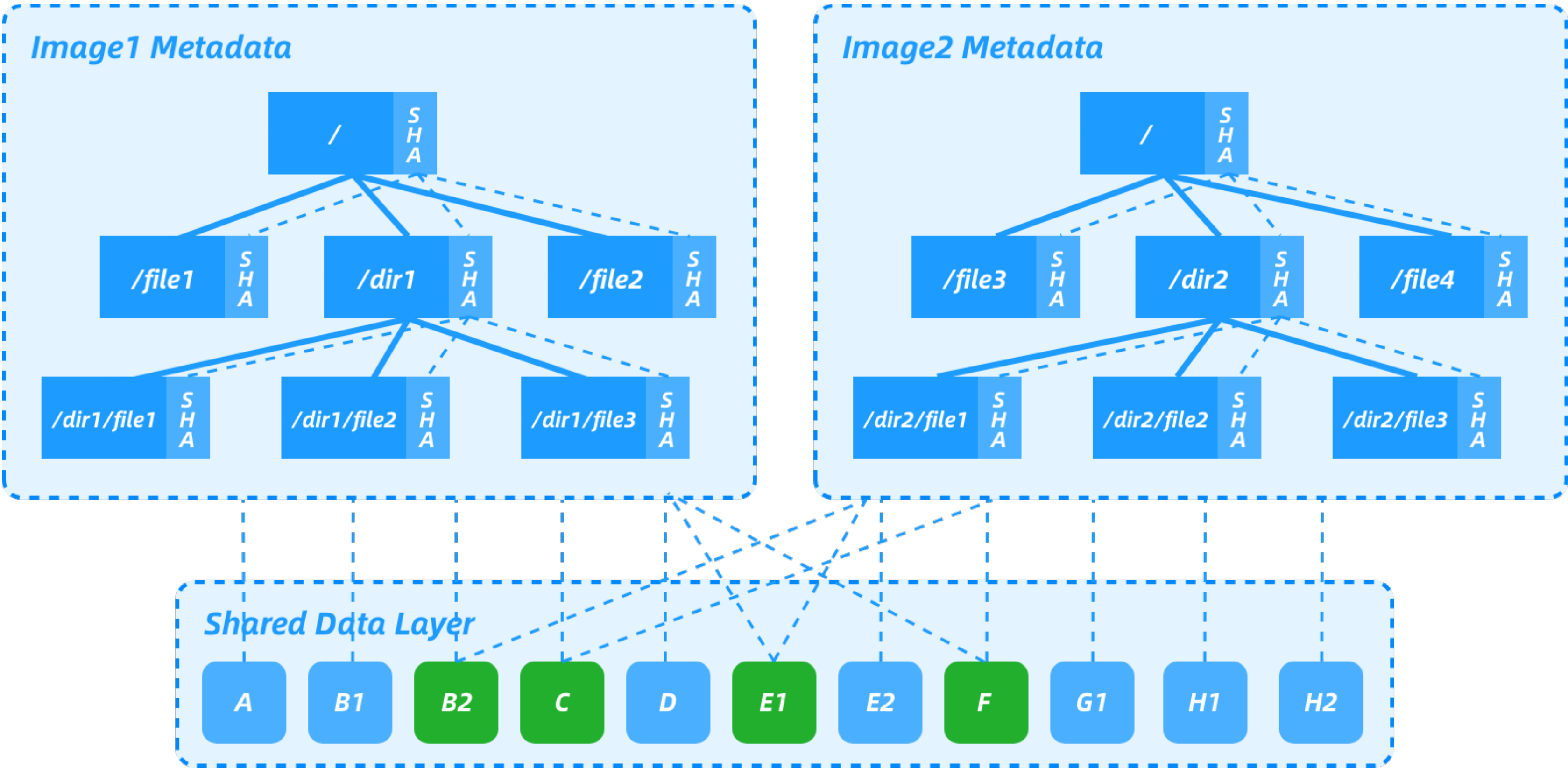
Nydus Image Service

- Container images are downloaded on demand
- Files are split into chunks with chunk level data deduplication
- Flatten image metadata and data to remove all intermediate layers
- End-to-end image data integrity
- Compatible with the OCI artifacts spec and distribution spec

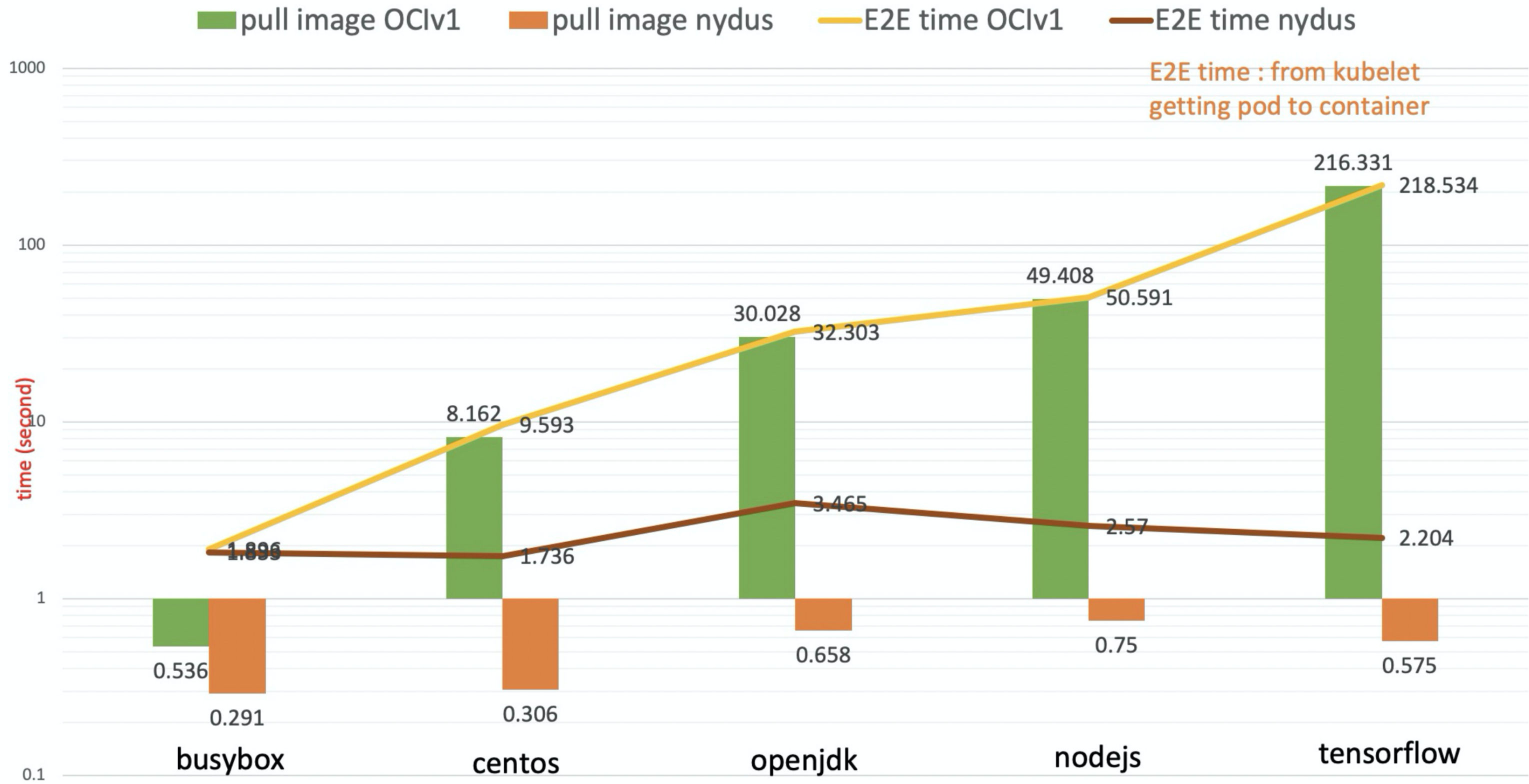
Nydus Architecture



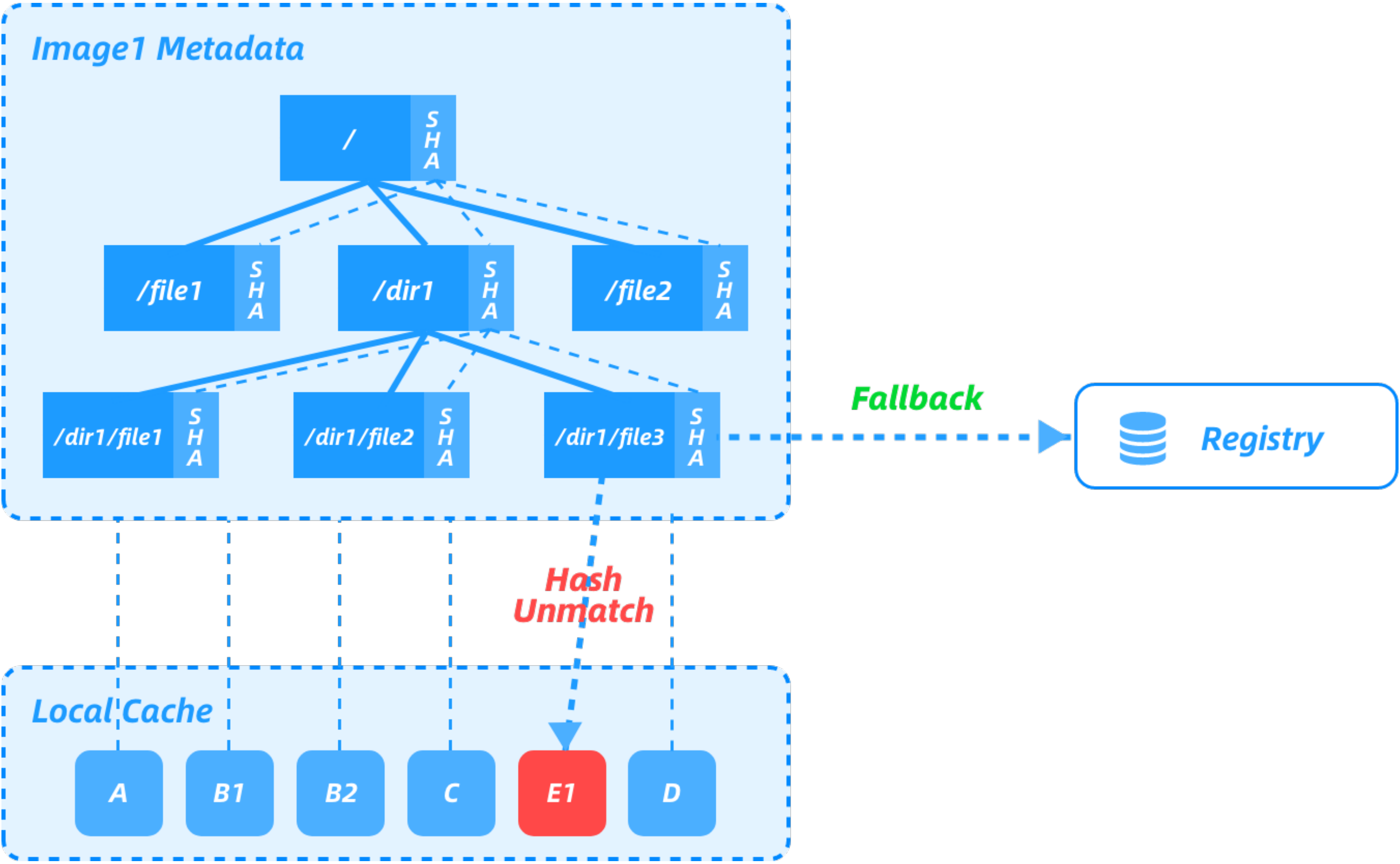
Nydus Image Format



Container Startup Time



End-to-End Data Integrity



Future Works

- Better integration with the container image ecosystem
- Support more compression algorithms
- Flexible chunk size
- Flexible deduplication levels
- Serve as a reference implementation to the OCIv2 image spec
- website: <https://github.com/dragonflyoss/image-service>

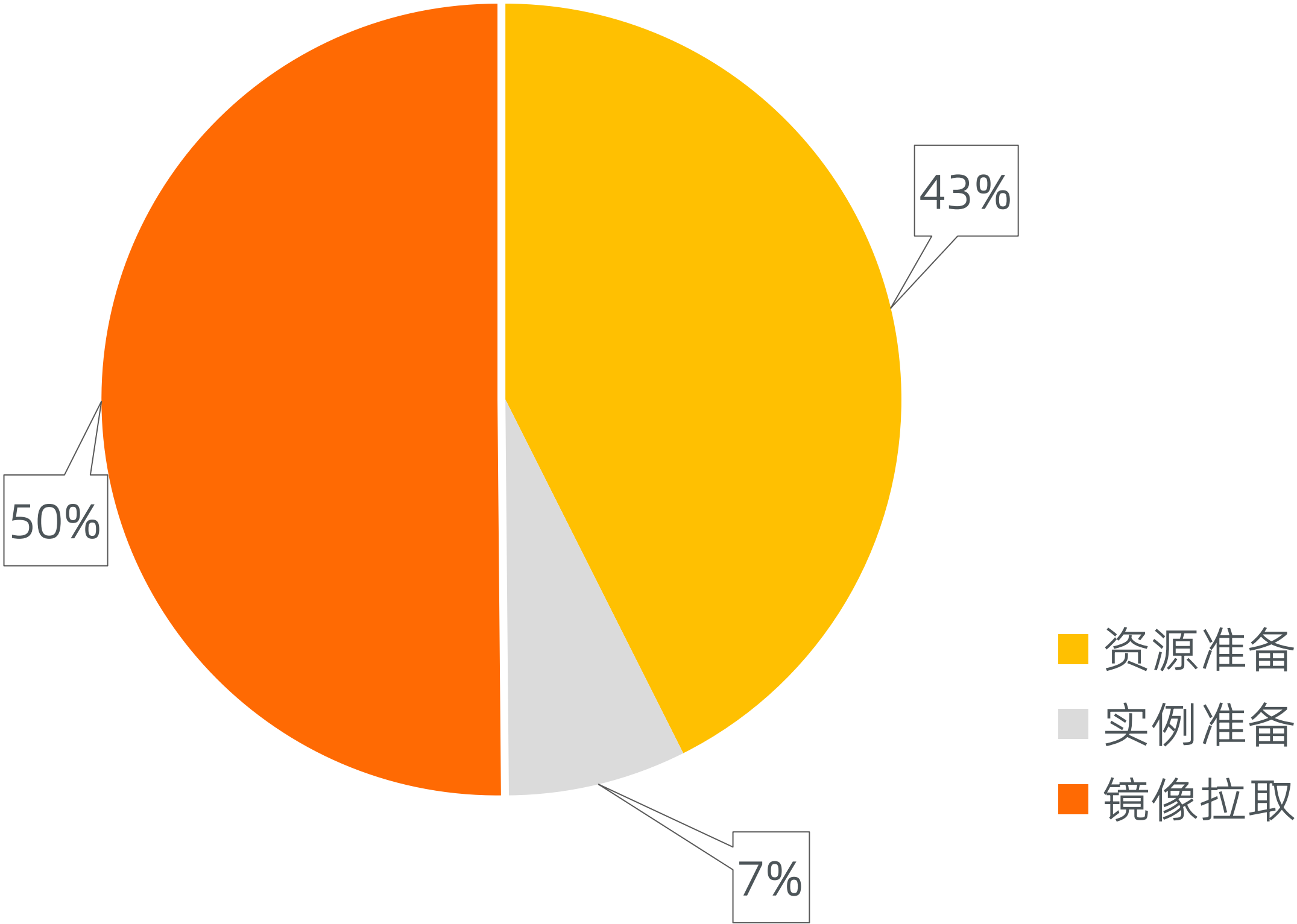
02

Nydus在阿里云的应用实践



线上拉取镜像时间分解

Nginx 容器镜像端到端创建时间分解



镜像缓存使用步骤

- 1.创建镜像缓存
- 2.查询镜像缓存信息
- 3.使用镜像缓存创建实例
(需要用户特殊配置)

创建镜像缓存

- 制作过程中会产生一定的费用。
- 目前单个镜像缓存最多支持包含20个镜像，镜像缓存大小可自定义，默认为20G。
- 镜像缓存支持设置保留时长，过期后自动删除，该功能默认不启用，如需使用请设置保留时长。

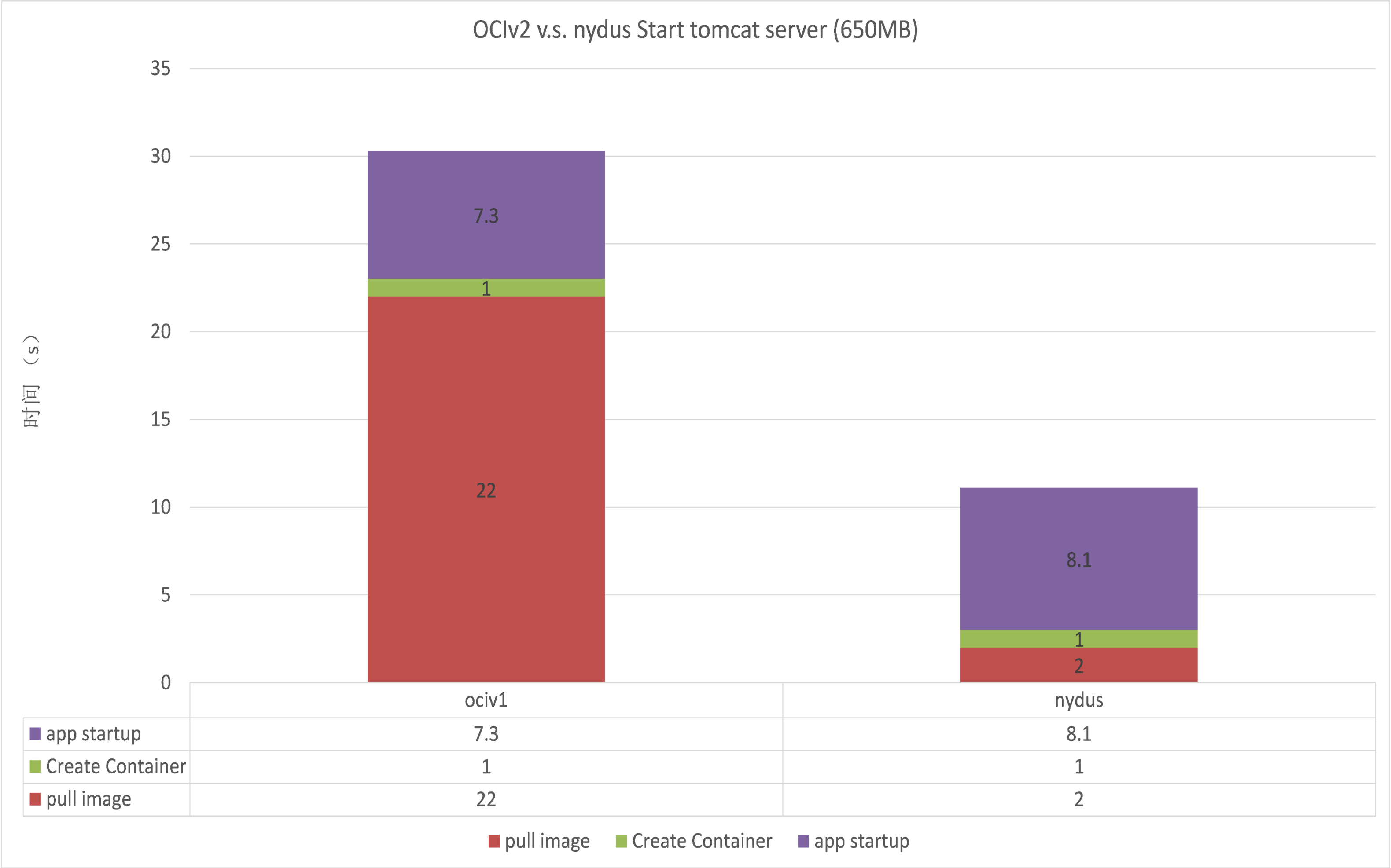
需要部署的组件

- Nydus-snapshotter
 - 为containerd提供snapshot API
- Nydusd
 - 为容器提供rootfs只读层

用户使用流程

1. 镜像格式转换（可用户转换，可registry转换）
2. 使用镜像创建容器（无需添加任何管控开关）
3. 探测到当前镜像支持nydus加速能力并使用nydus格式镜像启动容器，获得镜像加速能力

使用nydus的实际效果



使用nydus的实际效果

du

读取所有镜像文件元数据，是测试元数据性能的极端case。

```
time du -sh /
```

耗时(s)	oci	nydus
	2.6s	1.3s

tar

读取镜像所有数据，是测试读镜像性能的极端case。

```
time tar --exclude="/sys" --exclude="/proc" --exclude="/dev" --exclude="/tmp" -cvf /tmp/rootfs.tar /
```

耗时(s)	oci	nydus
	34.3s	137.1s

Thanks!



奥运会全球指定云服务商