

银数多云数据管家2.4版使用说明书

目录结构

- [1. 银数多云数据管家典型用户场景介绍](#)
 - [1.1 本地Kubernetes集群应用和数据的日常备份与恢复](#)
 - [1.2 在其它Kubernetes集群中恢复应用和数据](#)
 - [1.3 应用的跨云迁移](#)
- [2. 运行环境与兼容性](#)
- [3. 软件配置与授权](#)
- [4. 配置集群与备份仓库](#)
 - [4.1 配置待保护Kubernetes集群](#)
 - [4.2 配置备份仓库](#)
 - [4.3 配置StorageClass转换的ConfigMap](#)
 - [4.4 配置快照](#)
- [5. 备份设置](#)
 - [5.1 创建备份策略](#)
 - [5.2 执行备份任务](#)
 - [5.3 查看备份作业](#)
- [6. 恢复至本集群](#)
 - [6.1 创建应用恢复任务](#)
 - [6.2 执行应用恢复任务](#)
 - [6.3 查看应用恢复作业](#)
- [7. 恢复至其它集群](#)
 - [7.1 创建、执行、查看应用恢复任务](#)
 - [7.2 修改相应应用信息](#)
- [8. 跨集群迁移](#)
 - [8.1 创建迁移任务](#)
 - [8.2 执行迁移任务](#)
 - [8.3 查看迁移作业](#)
 - [8.4 修改相应应用信息](#)
 - [8.5 钩子程序](#)
- [9. YS1000的自备份与恢复](#)
- [10. 产品限制](#)
- [11. 故障与诊断](#)
 - [11.1 日志收集](#)
 - [11.2 常见问题](#)

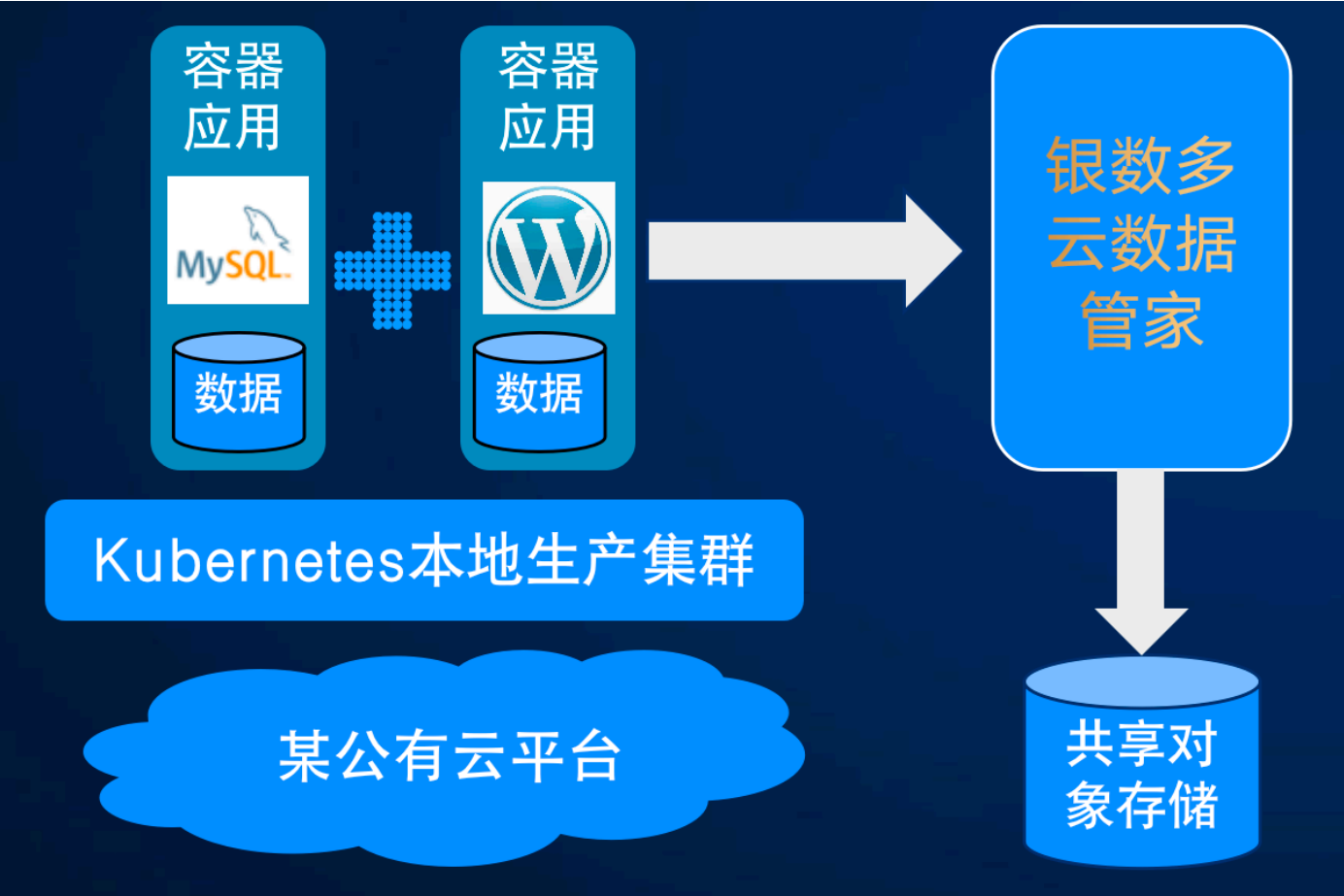
1. 银数多云数据管家典型用户场景介绍

银数多云数据管家（YS1000）是一款创新的云原生软件产品，为企业提供核心业务在多云架构下的备份恢复、应用迁移及容灾保护服务，它可以适用于多个云原生的业务场景中。

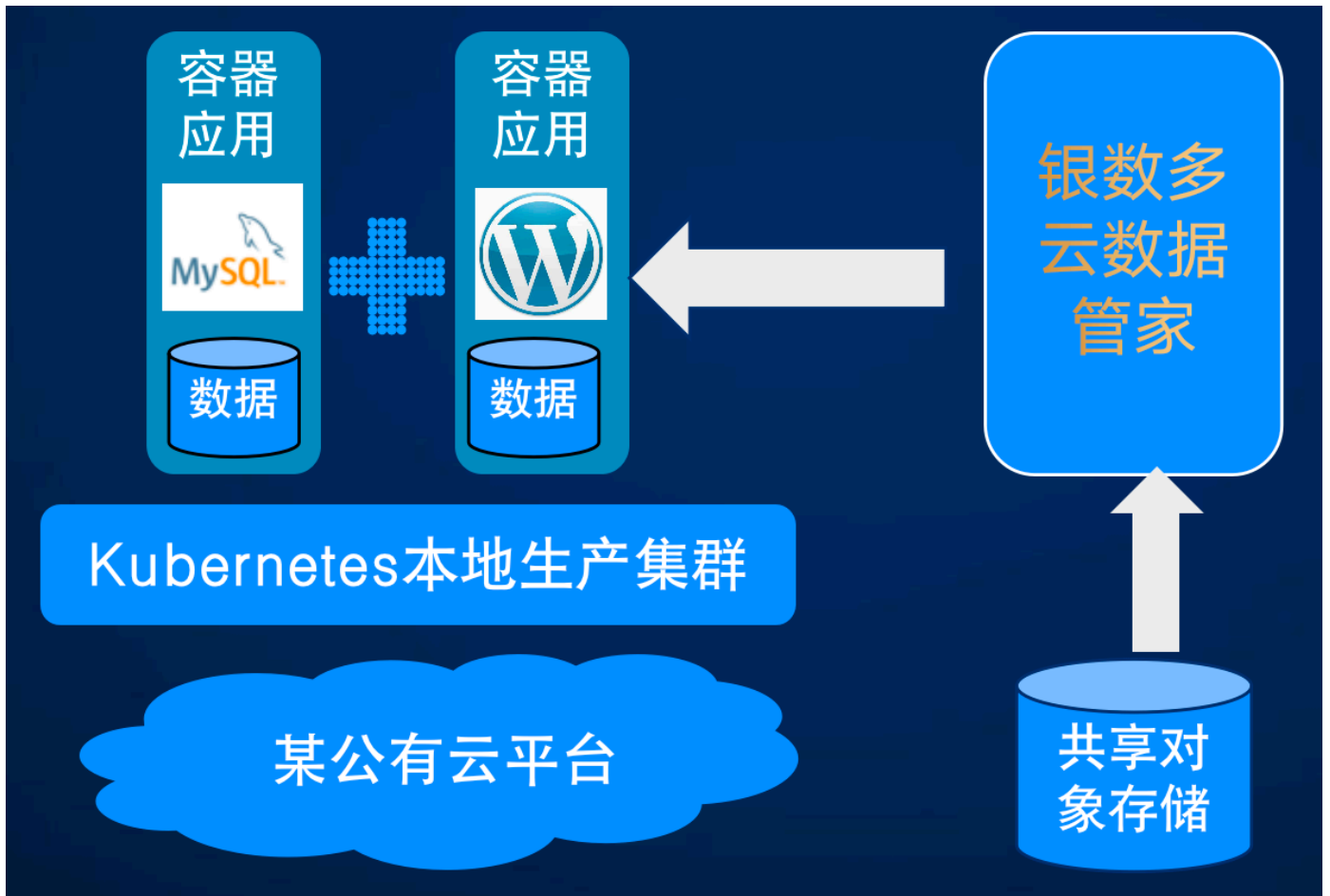
1.1 本地Kubernetes集群应用和数据的日常备份与恢复

通过银数多云数据管家，用户可以设置策略将容器应用和数据进行自动备份，当遇到事故时，可以对容器应用和数据进行一键恢复。

备份示意图：



恢复示意图：

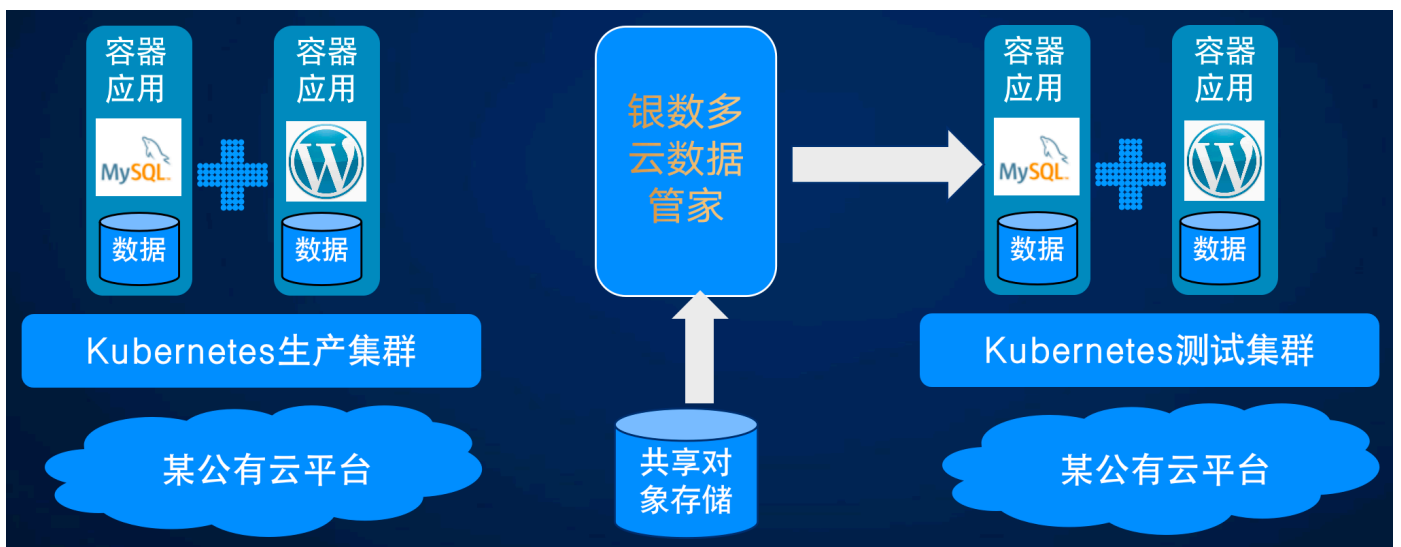


1.2 在其它Kubernetes集群中恢复应用和数据

用户测试环境中，常常希望能使用开发集群或生产集群中的应用和数据副本进行测试，通过银数多云数据管家，用户可以利用开发集群或生产集群中应用和数据的备份，在测试集群中恢复出同样的环境进行测试。

当生产集群遇到软硬件故障或站点故障时，通过银数多云数据管家，用户可以利用生产集群中应用和数据的备份，在灾备站点集群来恢复应用，以提高业务连续性。

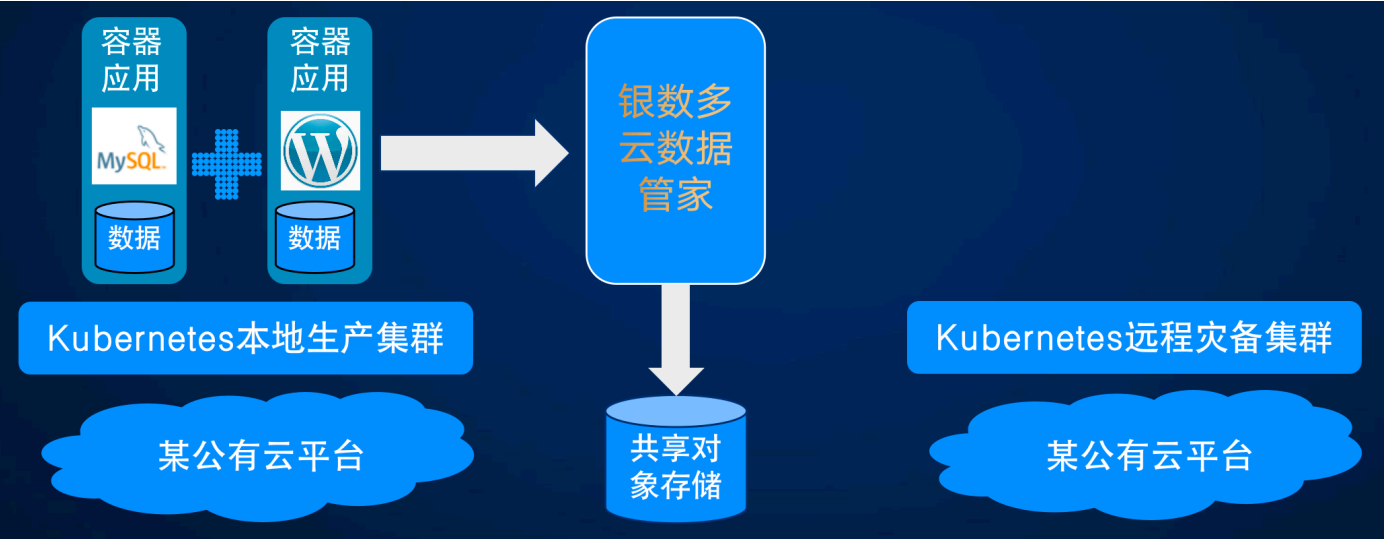
跨集群恢复示意图：



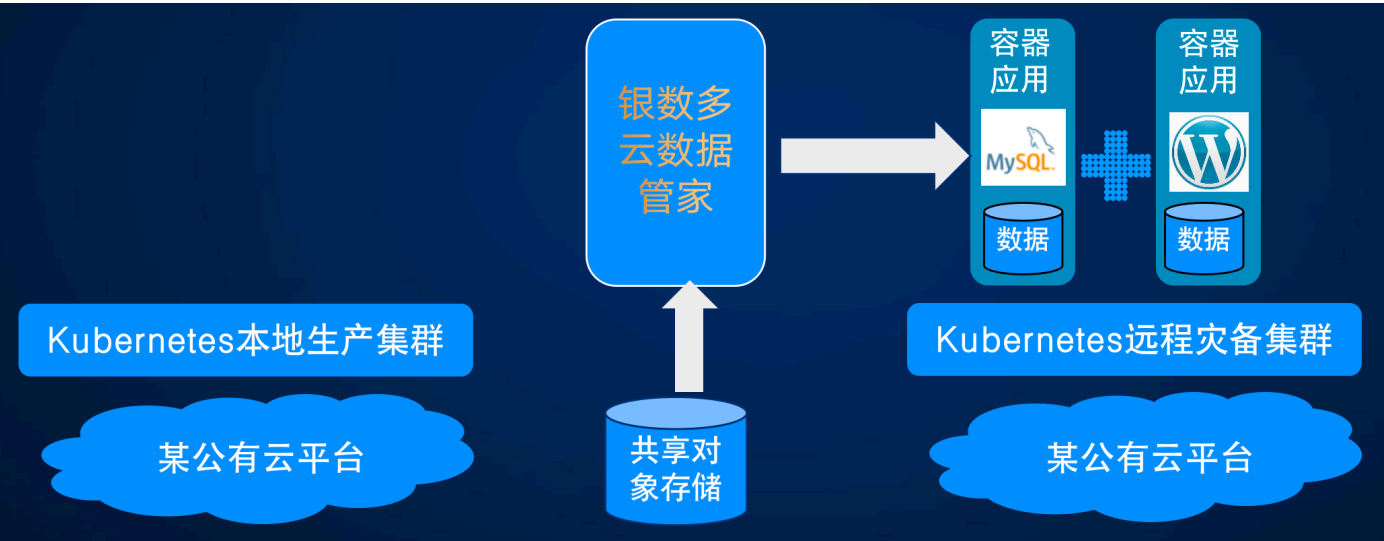
1.3 应用的跨云迁移

通过银数多云数据管家，用户可以一键将容器应用从一个Kubernetes集群迁移至另外一个完全异构的Kubernetes集群，如生产集群到灾备集群的切换，或不同厂商、架构的Kubernetes集群间的应用迁移等。

迁移示意图（一）



迁移示意图（二）



2. 运行环境与兼容性

推荐使用Firefox访问银数多云数据管家2.2版控制台。

目前YS1000 2.2版支持管理的Kubernetes版本、对象存储以及主存储如下表所示：

Kubernetes发行版	S3对象存储	云原生存储	Snapshot CRD
k8s社区版1.17-1.21	S3兼容的对象存储（minio, qingstor）	NFS	v1beta1
		Rook Ceph 1.4-1.8	v1

3. 软件配置与授权

	体验版	企业版	尊享版
集群节点数量	1-10	1-50	1-1000
最大管理集群数量	2	20	100
售后支持	无	24*7远程支持	24*7远程支持
是否需要授权	否	是	是

4. 配置集群与备份仓库

4.1 配置待保护Kubernetes集群

第一步，从左侧菜单栏中选择“集群信息”进入集群配置页面：



第二步，点击“添加集群”按钮进入集群添加页面：

添加集群

✕

集群名称 *

请输入集群名称

URL *

集群API服务器URL地址

账号令牌 *

复制并粘贴集群账号令牌

保存

检查连接

取消

“集群名称”请输入待保护Kubernetes集群名称。

“URL”请输入待保护Kubernetes集群的API服务器地址。

“账号令牌”栏，请创建一个具有 cluster admin 权限的 service account，并获取对应的token，或者使用已有的token。以下是创建的命令：

```
kubectl create serviceaccount k8sadmin -n kube-system
kubectl create clusterrolebinding k8sadminrb --clusterrole=cluster-admin --
serviceaccount=kube-system:k8sadmin
```

用下面命令拿到这个token：

```
kubectl -n kube-system describe secret $(sudo kubectl -n kube-system get secret | (grep
k8sadmin | | echo "$_") | awk '{print $1}')) | grep token: | awk '{print $2}'
```

第三步，点击“保存”按钮，YS1000会自动对待保护Kubernetes集群进行连接测试，如果连接成功，在状态栏会显示“连接成功”。

4.2 配置备份仓库

银数多云数据管家支持兼容S3接口的对象存储作为数据备份仓库。

第一步，从左侧菜单栏中选择“数据备份仓库”进入数据备份仓库配置页面：



第二步，点击“创建备份仓库”按钮进入备份仓库添加页面：

创建备份仓库

备份存储类型 *

S3

备份仓库名称 *

test-repo

S3 存储空间(Bucket) 名称 *

test1

S3 存储空间(Bucket) 区域(Region)

cn

S3 访问域名(Endpoint) *

http://ip:port

S3 访问秘钥(Access Key) *

.....

S3 访问秘钥口令(Secret Access Key) *

.....

☐ 使用SSL认证

选择 '使用SSL认证' 来验证对象存储的SSL连接安全性. 如果您正在使用个人证书 (self-signed certificate), 您需要关闭SSL验证或者上传您的证书.

提交

检查连接

取消

选择备份仓库类型，输入数据备份仓库名称，S3存储空间名称，S3存储空间区域，访问密钥及访问密钥口令，若选择的备份仓库类型为S3，则还需要输入访问域名。

第三步，点击“提交”按钮，银数多云数据管家会自动对数据备份仓库进行访问测试，如果测试成功，在状态栏会显示“连接成功”。

4.3 配置StorageClass转换的ConfigMap

在创建一个恢复任务之前，如果应用需要将PVC的StorageClass进行转换，可以在恢复的目标集群创建一个ConfigMap，以下是一个Ceph到NFS的一个yaml文件示例：

```
apiVersion: v1
kind: ConfigMap
metadata:
  # any name can be used; Velero uses the labels (below)
  # to identify it rather than the name
  name: sc-change-config
  # must be in the velero namespace
  namespace: qiming-backend
  # the below labels should be used verbatim in your
  # ConfigMap.
  labels:
    # this value-less label identifies the ConfigMap as
    # config for a plugin (i.e. the built-in restore item action plugin)
    velero.io/plugin-config: "true"
    # this label identifies the name and kind of plugin
    # that this ConfigMap is for.
```



```

    velero.io/change-storage-class: RestoreItemAction
data:
  # add 1+ key-value pairs here, where the key is the old
  # storage class name and the value is the new storage
  # class name.
  rook-ceph-block: managed-nfs-storage
  # managed-nfs-storage: rook-ceph-block

```

这种方法目前有几个限制：

- 这个StorageClass的转换作用域是ConfigMap所在的整个集群
这就意味着，例如，如果只想对某一个历史备份作业进行恢复，那恢复作业启动后，要确保没有其他恢复作业在运行，否则其他的恢复作业也会进行PVC的StorageClass转换。
- 这种转换方式暂时不支持针对CSI快照的备份恢复方式

4.4 配置快照

配置快照是为了支持用CSI快照方式来进行备份与恢复。下面以Ceph为例来描述如何配置快照。

第一步，创建Ceph的SnapshotClass，以下是一个snapshotclass.yaml的示例：

```

apiVersion: snapshot.storage.k8s.io/v1beta1
kind: VolumeSnapshotClass
metadata:
  name: csi-rbdplugin-snapclass
  labels:
    velero.io/csi-volumesnapshot-class: "true"
driver: rook-ceph.rbd.csi.ceph.com
parameters:
  # Specify a string that identifies your cluster. Ceph CSI supports any
  # unique string. When Ceph CSI is deployed by Rook use the Rook namespace,
  # for example "rook-ceph".
  clusterID: rook-ceph
  csi.storage.k8s.io/snapshotter-secret-name: rook-csi-rbd-provisioner
  csi.storage.k8s.io/snapshotter-secret-namespace: rook-ceph
deletionPolicy: Retain

```

其中，SnapshotClass的 `deletionPolicy` 必须是 `Retain`，并且加上velero需要的label (`velero.io/csi-volumesnapshot-class: "true"`)，这样后面在配置备份时候，会协调velero来产生并备份PV的快照。

第二步，检查Storageclass和Volumesnapshotclass对应关系。

查看Storageclass的provisioner名字, 这里是 `rook-ceph.rbd.csi.ceph.com`

```

bash# kubectl get sc rook-ceph-block -oyaml |yq .provisioner
rook-ceph.rbd.csi.ceph.com

```

查看Volumesnapshotclass的driver名字, 这里是 `rook-ceph.rbd.csi.ceph.com`

```
bash# kubectl get volumesnapshotclasses csi-rbdplugin-snapclass -oyaml |yq .driver
rook-ceph.rbd.csi.ceph.com
```

如果Storageclass的provisioner名字和Volumesnapshotclass的driver名字相同(例如ceph), 则跳到第三步; 如果不同(例如华为云csi-disk)则需要在Volumesnapshotclass添加annotation (`velero.io/csi-volumesnapshot-class-provisioner`), 对应的值为storageclass的provisioner名字。例子如下:

```
apiVersion: snapshot.storage.k8s.io/v1beta1
kind: VolumeSnapshotClass
metadata:
  name: csi-disk-snapclass
  annotations:
    velero.io/csi-volumesnapshot-class-provisioner: "everest-csi-provisioner"
  labels:
    velero.io/csi-volumesnapshot-class: "true"
driver: disk.csi.everest.io
parameters:
  ...
deletionPolicy: Retain
```

第三步, 配置Volumesnapshot CRD。

首先检查集群是否已经配置了Volumesnapshot CRD, 如果已经配置, 则跳过此步骤。

目前, 银数多云数据管家支持的Snapshot CRD版本为v1beta1。

1. 获取external-snapshotter的代码仓库:

```
git clone https://github.com/kubernetes-csi/external-snapshotter.git
```

2. 进入external-snapshotter目录, 切换到 `release-4.1` 分支

```
git checkout release-4.1
```

3. 执行以下命令来创建CRD:

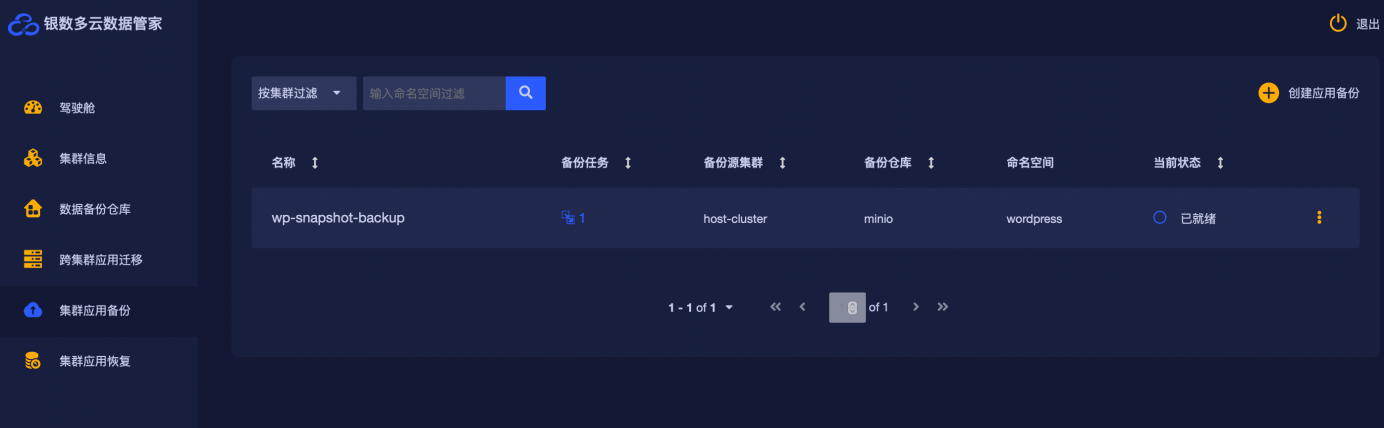
```
kubectl create -f config/crd
```

4. 执行以下命令来创建snapshot controller:

```
kubectl create -f deploy/kubernetes/snapshot-controller/
```

5. 备份设置

在银数多云数据管家左侧菜单栏中选择“集群应用备份”进入备份页面。



5.1 创建备份策略

第一步，点击“创建应用备份任务”按钮进入备份任务添加页面：

创建一个备份任务

1 通用

2 命名空间

3 持久卷

4 复制选项

5 数据库备份一致性保护

6 钩子程序

通用

创建应用备份任务

应用备份名称 *

wordpress

选择备份集群

备份集群 *

host-cluster

选择备份仓库

备份仓库 *

minio

选择备份方法

下一步

上一步

取消

用户需要输入备份任务的名称，选择待保护的集群，以及备份目标仓库。

创建一个备份任务

1 通用

2 命名空间

3 持久卷

4 复制选项

5 数据库备份一致性保护

6 钩子程序

minio

选择备份方法

☒ 文件拷贝

☐ 快照拷贝

选择备份方式

☒ 按需备份

☐ 定时备份

选择备份频率

备份频率(单位: 小时)

3

选择备份保留时长

备份保留时长(单位: 天)

1

下一步

上一步

取消

同时，用户需要选择备份方法、备份方式并根据备份方法选择对应参数（这里我们选择直接用文件拷贝）。

备份方式支持按需备份和定时备份（这里我们选择直接用按需备份）。

- 按需备份由用户按照需求来触发备份作业。
- 定时备份由系统按照用户指定的备份频率自动触发备份作业。选择定时备份时，必须指定备份频率。

当备份数据超过指定备份保留时长后，相关的备份数据将被系统自动删除。

第二步，点击“下一步”选择需要备份的命名空间。

用户可以通过“筛选”按钮对命名空间进行筛选，或者通过搜索栏搜索相关名称快速找到需要备份的命名空间。



第三步，点击“下一步”确认需要备份的持久卷。

系统会自动选择出用户指定命名空间中使用到的持久卷，用户可以进行确认。



第四步，点击“下一步”确认持久卷的备份方法。

第五步，点击“下一步”选择备份前保护数据库一致性（目前仅支持mysql，后续开放mongo和postgresql）。

第六步，跳过钩子程序直接点击完成（YS1000 2.2版本提供了数据一致性保护的默认钩子，但目前无法与自定义钩子程序同时使用，如需使用则需跳过数据一致性保护后创建）

点击“完成”按钮后，备份任务创建成功，系统会自动对备份任务进行验证。

YS1000 2.2版本的备份方式除了仍支持基于存储快照的备份和基于文件拷贝的备份，还新增了存储快照备份+后台数据导出的高级模式。

如需使用快照备份+后台数据导出的方式：

- 1、备份方法选择“快照拷贝”，备份方式可以选择“按需备份”或“定时备份”，“是否导出快照”选择“是”。
- 2、若选择“按需备份”，则后台导出该备份任务按需备份的快照。
- 3、若选择“定时备份”，则需要设置“导出频率”，按照频率导出该时间段内新增的所有快照。

5.2 执行备份任务

对于定时备份策略，系统会自动按照定时设定进行备份。同时，用户可以选择备份任务手动触发备份作业。

在备份页面中，选择对应备份任务的“”列，在操作中选择“备份”，即可触发备份作业。

点击“确定”按钮，备份作业即开始运行。

5.3 查看备份作业

在备份页面中，点击“备份任务”栏的链接，即可查看备份作业的执行情况。

6. 恢复至本集群

从备份恢复应用至本集群一般在本地应用出现故障时使用（如命名空间被意外删除等），恢复往往无需进行应用资源本身相关的修改（如对外服务的域名和端口等）。

在YS1000左侧菜单栏中选择“集群应用恢复”进入恢复页面。

6.1 创建应用恢复任务

第一步，点击“创建应用恢复任务”按钮进入应用恢复任务添加页面：

输入应用恢复任务名称，并选择目标恢复集群，此处选择本地kubernetes集群。

这里可以勾选对命名空间进行修改（注意功能只适用于单个命名空间的备份）。

第二步，点击下一步并选择一个备份任务。

第三步，点击“下一步”选择需要一个已完成的备份作业。

用户可以根据备份作业对应的时间、命名空间等信息选择自己需要的备份数据。

第四步，点击“下一步”选择应用恢复后可以执行的钩子程序。

点击“完成”按钮后，恢复任务创建成功，系统会自动对恢复任务进行验证。

6.2 执行应用恢复任务

在应用恢复页面中，选择恢复任务列的链接，在相应恢复作业的“”列操作中选择“激活”，即可触发任务恢复作业。

6.3 查看应用恢复作业

在应用恢复页面中，点击恢复任务栏的链接，即可查看恢复作业的执行情况。

7. 恢复至其它集群

从备份恢复应用至其它集群一般在远程容灾场景、开发/测试场景、数据重用场景中使用，恢复完成后可能需要进行应用资源相关的修改（如对外服务的域名和端口等）。

同恢复至本地集群一样，在银数多云数据管家左侧菜单栏中选择“集群应用恢复”进入恢复页面。

7.1 创建、执行、查看应用恢复任务

对于创建、执行以及查看恢复任务的操作，可以参考5.1-5.3的内容，这里唯一的区别就是在选择恢复的集群时，选择的是跟备份集群不同的集群。

7.2 修改相应应用信息

在其它集群（如测试集群）恢复应用后，对于和原备份集群有冲突的资源需要进行相应的更改，然后才能在其它集群完全恢复应用。

如wordpress在新集群恢复后，仍旧会绑定备份集群中使用的域名，这时需要管理员将域名指向新集群IP，或者执行特定脚本来更改新域名。

8. 跨集群迁移

在银数多云数据管家左侧菜单栏中选择“跨集群应用迁移”进入应用迁移页面。

8.1 创建迁移任务

【注意】 在开始执行跨集群迁移任务前，要确认迁移的目标集群不存在资源冲突。例如，如果迁移一个命名空间A到集群B，则要确认集群B不存在命名空间A。

第一步，点击“创建迁移任务”按钮进入迁移任务添加页面：

用户需要输入迁移任务名称，选择源端集群和目标端集群，以及备份仓库。

第二步，点击“下一步”选择需要迁移的命名空间。

用户通过多选框选择需要迁移的命名空间，目前银数多云数据管家2.2版支持同时迁移多个命名空间。

用户可以通过“筛选”按钮对命名空间进行筛选，或者通过搜索栏搜索相关名称快速找到需要备份的命名空间。

第三步，点击“下一步”确认需要迁移的相关持久卷。

系统会自动选择出用户指定命名空间中使用到的持久卷，用户可以进行确认。

第四步，点击“下一步”选择持久卷的拷贝方法。

银数多云数据管家2.2版本支持文件系统拷贝的方式进行跨集群迁移。用户需要选择目标集群上应用恢复时需要使用的StorageClass。目标端集群使用的StorageClass和源端集群使用的StorageClass可以不同。

8.2 执行迁移任务

在应用迁移页面中，选择对应迁移任务的“”列，在操作中选择“一键迁移”，即可触发迁移作业。

迁移过程默认会停掉源集群中选定命名空间内的应用，以保证数据一致性。

用户可以选择迁移过程中是否停止源集群中的应用运行。例如对于支持宕机一致性（crash-consistency）的应用，如果不希望在迁移过程中停止源集群中的应用运行，用户可以勾选此选项。

点击“迁移”按钮，迁移作业即开始运行。

8.3 查看迁移作业

在迁移页面中，点击迁移任务栏的链接，即可查看迁移作业的执行情况。

8.4 修改相应应用信息

在目标端集群重新启动应用后，对于和源集群有冲突的资源需要进行相应的更改，然后才能在目标端完全恢复应用。如wordpress在目标端重启启动后，仍旧会绑定源端使用的域名，这时需要管理员将域名指向目标端集群IP，或者执行特定脚本来更改新域名。

8.5 钩子程序

钩子程序 (Hook) 提供在备份、恢复以及迁移应用执行前后添加应用自定义逻辑的功能，满足不同应用的需求。简单的自定义逻辑支持通过ansible playbook 脚本来实现，或者用户也可以根据应用需要开发自定义的钩子程序（容器镜像）。

钩子程序将会以 kubernetes 作业 (jobs.batch) 的形式在指定的备份、恢复以及迁移前后阶段执行。

常见的场景如下：

1. 应用数据一致性保证

- 应用备份之前，通过自定义的 "prebackup" 钩子程序，调用应用的静默 (quiesce) 接口，暂停应用并将应用内存数据以及文件系统缓存刷入磁盘
- 应用备份之后，通过自定义的 "postbackup" 钩子程序，调用应用的恢复 (unquiesce) 接口，恢复应用执行

2. 数据验证

- 应用备份之后，通过自定义的 "postbackup" 钩子程序，验证数据完整性

3. 环境修改

- 应用恢复到远程集群以后，需要对环境配置进行修改，比如通过 ingress 修改用户访问 web URL 地址

下面的例子以 wordpress 应用在跨集群迁移的场景下进行说明，wordpress 部署代码见 [wordpress example](#)

- 迁移之前应用通过 <https://wp-demo.jibudata.com:30165> 访问 wordpress
- 因业务需求，wordpress 需要迁移到远端目标集群，通过 <https://blog.jibudata.com:30165> 访问原 wordpress

1. 参考 [7.1 创建迁移任务](#)，在创建迁移任务最后一步，如下图，点击 **添加钩子程序**

2. 输入钩子程序名称，此处为 `postrestore-change-ingress-wp-url`

3. 在 `Ansible playbook` 区域选择上次Ansible playbook 脚本或者直接将脚本内容复制到文本框中。

此例的 ansible playbook 脚本为 [wp_migration_url_update.yaml](#)。

脚本根据wordpress 应用迁移要求，修改 mysql 数据库中的文章和站点的Web URL 地址，同时修改对外访问的 ingress 地址。

4. 选择钩子程序执行的运行式环境并点击**创建** 按钮。

此例中，Ansible playbook 脚本在目标集群中执行，脚本示例采用在 wordpress mysql pod上直接运行命令进行数据库修改。

填入钩子程序作业运行所需的服务账户以及所在的命名空间，这里使用 `qiming-migration`（velero 安装的命名空间）以及 `velero` 服务账户来运行任务。

选择执行阶段为 "PostRestore".

注意：

钩子程序运行操作需要确保指定的服务账户具有运行所需要的权限。

此例中 `velero` 服务账户具有 cluster admin 权限，因此可以通过 ansible 脚本登录到目标容器 (wordpress命名空间中的 mysql pod) 进行操作。

5. 创建完成后，返回迁移任务向导页面，在钩子程序页面显示创建结果，之后点击 **完成** 按钮，参考 [7.2 执行迁移任务](#) 执行迁移操作

6. 完成迁移之后，访问新的 web URL 地址，可观察到 wordpress 页面以更新为新的域名地址

迁移之前：

迁移之后

9. YS1000的自备份与恢复

- 第一步，在宿主集群上打开自备份功能

```
kubectl -n qiming-migration edit migconfigs.migration.yinhestor.com qiming-config
```

修改参数（自备份间隔时长 [5, 1440] minutes），下例为12小时自备份间隔

```
backupIntervalMinutes: 720
```

- 第二步，查看自备份job

```
kubectl -n qiming-migration get migconfigs.migration.yinhestor.com qiming-config -o yaml
```

```
images: {}
lastBackupItems:
- backupTime: "2021-11-15T09:17:34Z"
  name: ys1000-backup-1636967854
  status: Completed
- backupTime: "2021-11-15T09:22:35Z"
  name: ys1000-backup-1636968155
  status: Completed
- backupTime: "2021-11-15T09:27:35Z"
  name: ys1000-backup-1636968455
  status: Completed
lastBackupName: ys1000-backup-1636968455
lastBackupTime: "2021-11-15T09:27:35Z"
phase: Ready
```

- 第三步，在新集群上成功安装ys1000并配置指向原数据备份仓库S3之后，使用 `self-restore.sh` 脚本进行恢复

<https://github.com/jibutech/docs/tree/release-2.2/self-restore>

```
[root@remote-master home]# ./self-restore.sh ys1000-backup-1636968455
2021年 11月 15日 星期一 17:55:02 CST Trigger velero restore on backup ys1000-backup-1636968455 ...
Restore request "restore-1636970102" submitted successfully.
Run `velero restore describe restore-1636970102` or `velero restore logs restore-1636970102` for more details.
Error from server (NotFound): restores.velero.io "restore-1636970102" not found
2021年 11月 15日 星期一 17:55:03 CST Velero restore status ...
2021年 11月 15日 星期一 17:55:08 CST Velero restore status InProgress ...
2021年 11月 15日 星期一 17:55:14 CST Velero restore status InProgress ...
2021年 11月 15日 星期一 17:55:19 CST Velero restore status InProgress ...
2021年 11月 15日 星期一 17:55:24 CST Velero restore completed.
```

10. 产品限制

- PVC的类型暂时不支持Host Path方式
- 如果PVC的dataSource是VolumeSnapshot，无法迁移或恢复到异地
- 如果Pod自带有emptyDir类型的Volume，备份会出错

解决方法：对要备份的Pod加一个annotation：

```
kubectl -n <namespace> annotate pod/<podname> backup.velero.io/backup-volumes-excludes=<volumename>
```

11. 故障与诊断

11.1 日志收集

请参考 https://github.com/jibutech/docs/tree/main/log_collection#readme 部署日志收集容器镜像，执行日志收集命令后发送给技术支持人员。

11.2 常见问题

- 快照备份不工作
可能原因：快照的SnapshotClass没配好，比如没有加所需要的label。
解决方法：请参考3.4节的“配置快照”，把相应的配置做好。
- 快照恢复失败
可能原因：快照的SnapshotClass的 `deletionPolicy` 不是 `Retain`。
解决方法：用 `kubectl` 查看相应的 `volumesnapshotcontents` CR，看 `deletionPolicy` 是不是 `Retain`，如果不是，请参考3.4节的“配置快照”，并修改SnapshotClass的yaml文件，重新apply。
- 备份/恢复/迁移任务卡在50%左右一直不动
可能原因：当前集群前面有备份/恢复一直完成不了，卡在Velero的队列中。
解决方法：查看是否有一个备份/恢复一直在进行，等前一个备份完成，或者超时（现在大约要4小时）后，当前这个任务就会开始。如果不想等，可以重启Velero的Pod来观察问题是否解决。
- 恢复很慢，花了比预期多很多的时间
可能原因：如果是异地恢复，可以去查看恢复的命名空间，看Pod是不是Image Pull失败，或者有什么异常情况，导致Pod起来太慢。

- 执行应用迁移时，待迁移的应用的PVC的 `DataSource` 不能是 `VolumeSnapshot`，否则迁移会一直卡在目标集群的恢复阶段。

原因：如果待迁移的应用是通过CSI快照方式恢复出来的，PVC的 `DataSource` 就会变成 `VolumeSnapshot`，这时候再迁移就会出问题。

解决方法：要确保待迁移的应用没有被恢复过，或者是文件系统的方式恢复的。

- 部署到k8s集群时，velero不能正常运行，并且报 `unexpected directory structure for host-pods volume ...` 的错误

原因：这是由于k8s在安装时，没有使用标准的 `/var/lib/kubelet/pods/` 的目录格式。

解决方法：执行 `mount -l | grep kubelet`，找到集群实际的Pod的路径，例

如 `/var/k8s/kubelet/pods/`，然后执行：

```
kubect1 patch ds/restic --namespace qiming-migration --type json -p
' [{"op": "replace", "path": "/spec/template/spec/volumes/0/hostPath", "value": {
"path": "/var/k8s/kubelet/pods"}} ] '
```

- 备份一个k8s版本 ≥ 1.21 集群上的应用，再恢复到一个 k8s版本 ≤ 1.20 的集群上后，应用中pod无法正常running的问题，参考：

<https://velero.cn/d/37-k8s-121-beta-feature-boundserviceaccounttokenvelerorestic>