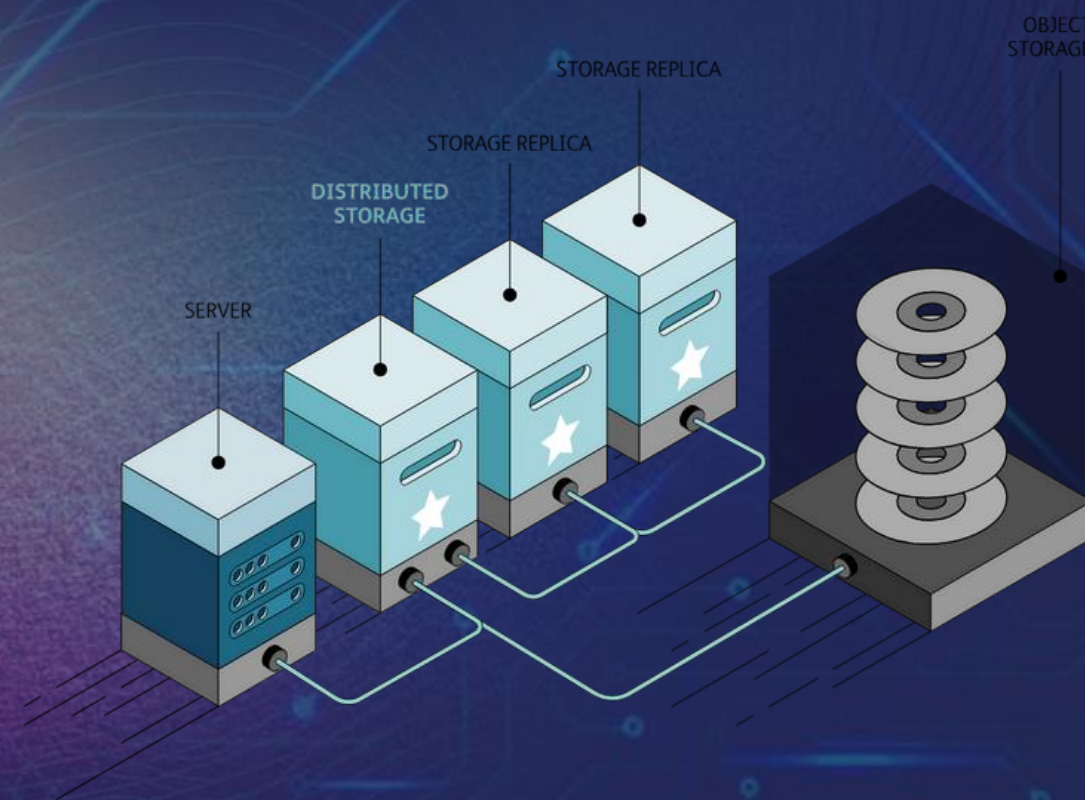


Bringing Object Storage to K8s: A deep dive into SI and its integration at CERN



Types of storage

File storage

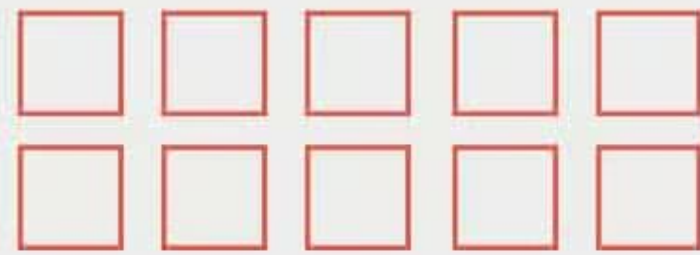


Data is managed as files in a hierarchical structure.

Store and handle data collaboratively, with shared access.

Simple interface to create, delete and organize files.

Block storage

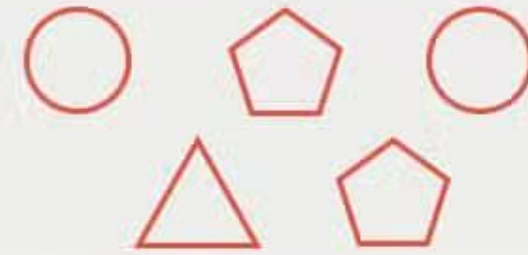


Data is managed as blocks, arbitrarily organized.

Store and handle large volumes of structured data from different units.

Access in different operating systems.

Object storage



Data is managed as objects, stored in a flat structure.

Store and handle large volumes of unstructured data thanks to metadata.

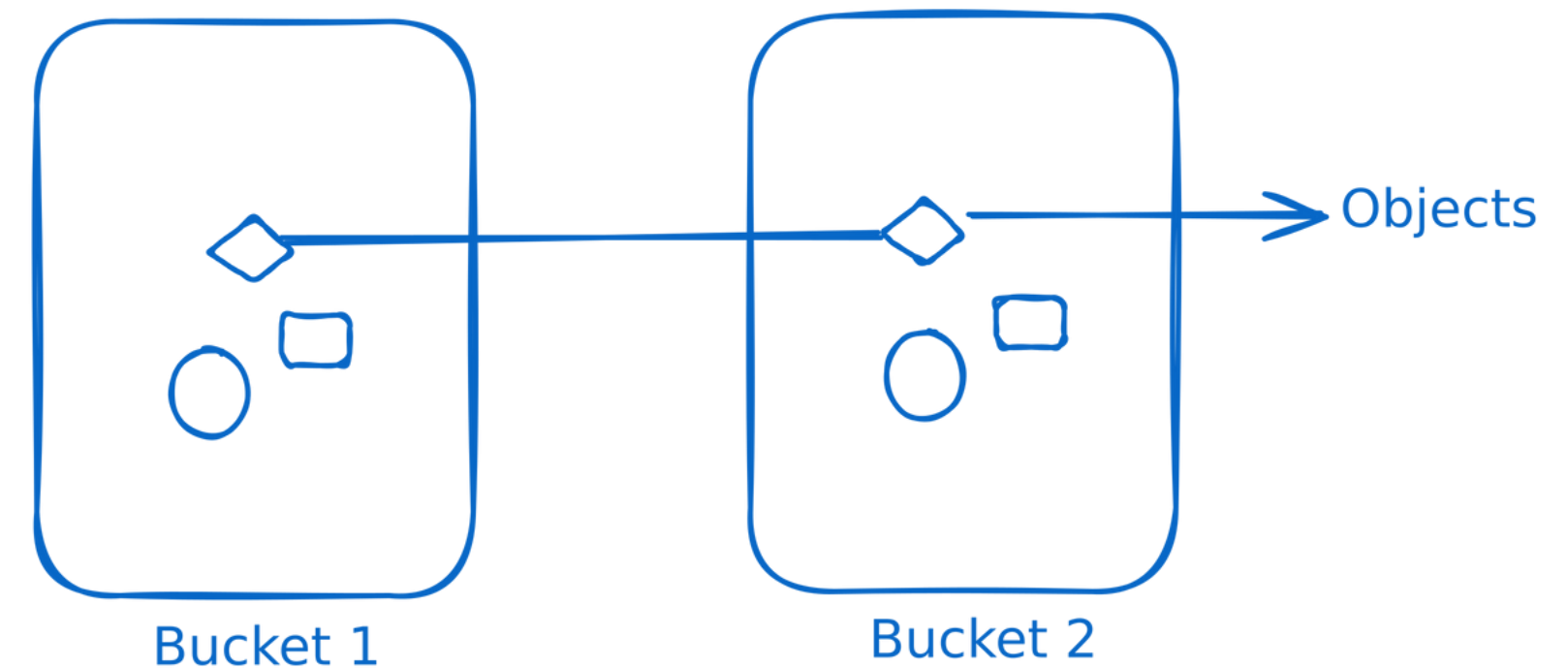
Access to data via an API or user interface.



Object Storage

- Data stored in units known as **Objects** kept inside logical containers known as **Buckets**
- can be **accessed by simple network APIs**
- **cost efficient** and can scale into extremely large quantities while maintaining quick access

Object Storage



HARBOR (Use Case)

Storage in K8s

File and Block Storage:

- CSI (Container Storage Interface) in K8s
- Prior to CSI, connecting to new volumes plugins needed to be directly a part of core Kubernetes.
- CSI moved this logic to separate drivers
- We need something similar for **Object Storage**? an interface to manage buckets in k8s
- Here comes COSI (Container Object Storage Interface)



Manually Creating Buckets

The screenshot displays the OpenStack dashboard interface. On the left, a sidebar lists navigation options: Project, API Access, Compute, Volumes, Container Infra, Network, Object Store, Containers (highlighted), and Share. The main content area shows the 'Containers' page with a breadcrumb 'Project / Object Store / Containers', a '+ CONTAINER' button, and a search bar. A list of containers is visible, including 'test-jack'. A 'Create Container' modal dialog is open in the foreground, containing the following fields and options:

- Container Name**: A text input field with a red asterisk indicating it is required. Below it, a message states: "Container name must not contain '/'".
- Storage Policy**: A dropdown menu currently set to "default-placement".
- Container Access**: Two radio buttons, "PUBLIC" (selected) and "NOT PUBLIC". Below them, a note reads: "A Public Container will allow anyone with the Public URL to gain access to your objects in the container."
- Buttons**: "X CANCEL" and "✓ SUBMIT".

Bucket Provisioning using COSI

User creates a **bucketclaim** with the configuration to create a bucket.

```
kind: BucketClaim
apiVersion: objectstorage.k8s.io/v1alpha1
metadata:
  name: sample-bucket3
  namespace: notea
spec:
  bucketClassName: sample-bcc
  protocols:
  - s3
```

COSI will create a new bucket and manage the access control for it.

Bucket Quotas & Backups

Bucket Quotas:

- Providing limited storage to buckets
- Mutability in storage: increasing storage of buckets, currently in the spec of COSI

Backups:

- Important in case someone accidentally deletes a bucket or if you want to move to a specific version of the bucket.
- Helm Chart for backing up buckets created by COSI from s3.cern.ch.
- https://gitlab.cern.ch/karanjot/s3_cern_backup

Thank you!

Questions?



+



karanjot.singh@cern.ch
github.com/Oxquark