



KubeCon



CloudNativeCon

Europe 2026

#KubeCon #CloudNativeCon

Snapshots Gone Wild: Taming Multi-PVC Chaos with VolumeGroupSnapshot

Shubham Pampattiwar & Scott Seago
Engineering at Red Hat
Velero Maintainers



About Us



KubeCon



CloudNativeCon

Europe 2026



Shubham Pampattiwar

- Principal Software Engineer, Red Hat
- OpenShift Engineering team
- Velero maintainer
- Data on Kubernetes Ambassador
- GitHub: @shubham-pampattiwar



Scott Seago

- Principal Software Engineer, Red Hat
- OpenShift Engineering team
- Velero maintainer
- GitHub: @sseago

What is Velero ?

Backup, restore, and disaster recovery for Kubernetes

- Back up all cluster resources and persistent volumes
- Multiple volume backup methods: CSI snapshots, file-system backup, cloud provider snapshots
- Scheduled backups, pre/post hooks, data mover for cross-cloud DR

Velero's CNCF Sandbox application has been approved by the TOC!

"Backups should be simple, reliable, and ideally so boring they never make headlines."



KubeCon



CloudNativeCon

Europe 2026

Agenda

- **The Problem:** Why multi-volume backups break
- **VolumeGroupSnapshot:** The Kubernetes answer
- **Bringing VGS to Velero:** Design and implementation
- **Real-World Challenges:** War stories from CSI drivers
- **Results & Blueprint:** How to adopt VGS today



Act 1: The Problem

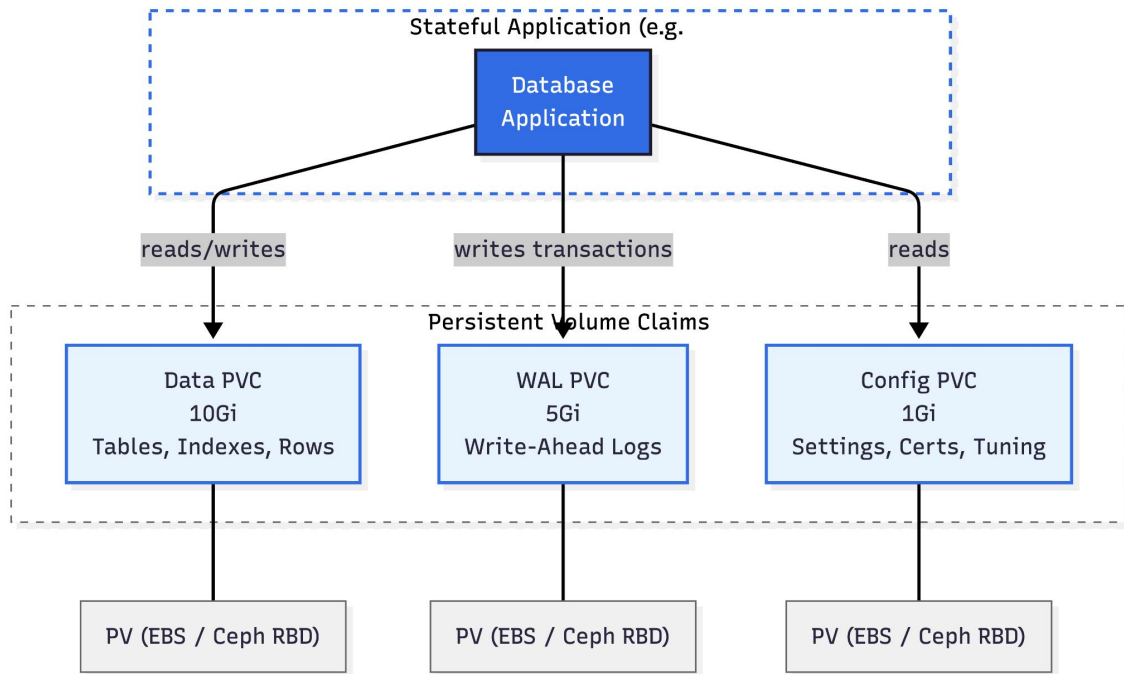


KubeCon



CloudNativeCon

Europe 2026



Databases, message queues, analytics workloads, they all spread data across multiple volumes.

Data + Write-Ahead Logs + Configuration = One logical application

Individual Snapshots: The Chaos

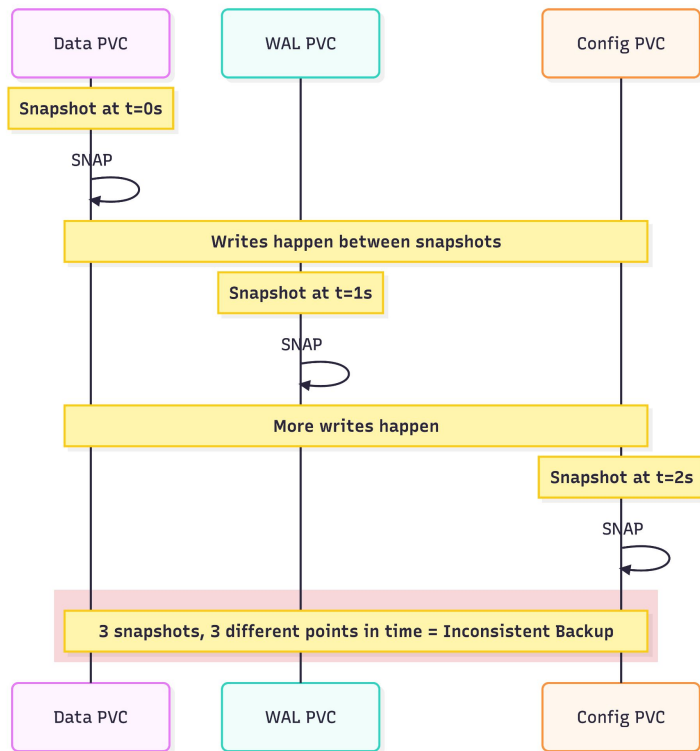


KubeCon



CloudNativeCon

Europe 2026



The Problem: Each snapshot captures a different point in time.

- Data PVC snapshot: **before** Write X
- WAL PVC snapshot: **after** Write X, **before** Write Y
- Config PVC snapshot: **after** Write Y

Result: Your "backup" contains an impossible state that never existed.

What Can Go Wrong?



KubeCon



CloudNativeCon

Europe 2026

Scenario	What happens on Restore
DB data + WAL out of sync	Corruption. WAL references missing data
Kafka data + metadata out of sync	Lost messages or duplicate delivery
Distributed DB (Cassandra)	Split-brain. Nodes have inconsistent state
App config + data mismatch	Crashes on startup, wrong schema version

The deeper issue: Individual snapshots give you a **false sense of security**.
Your backup succeeded. Your restore will fail.

What We Actually Need

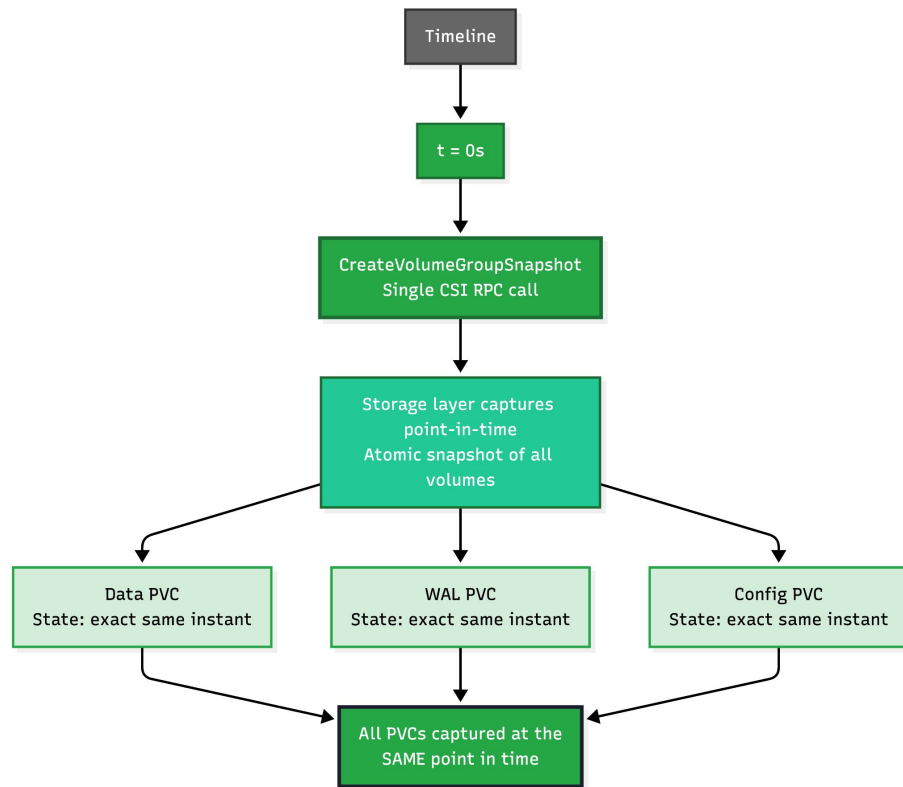
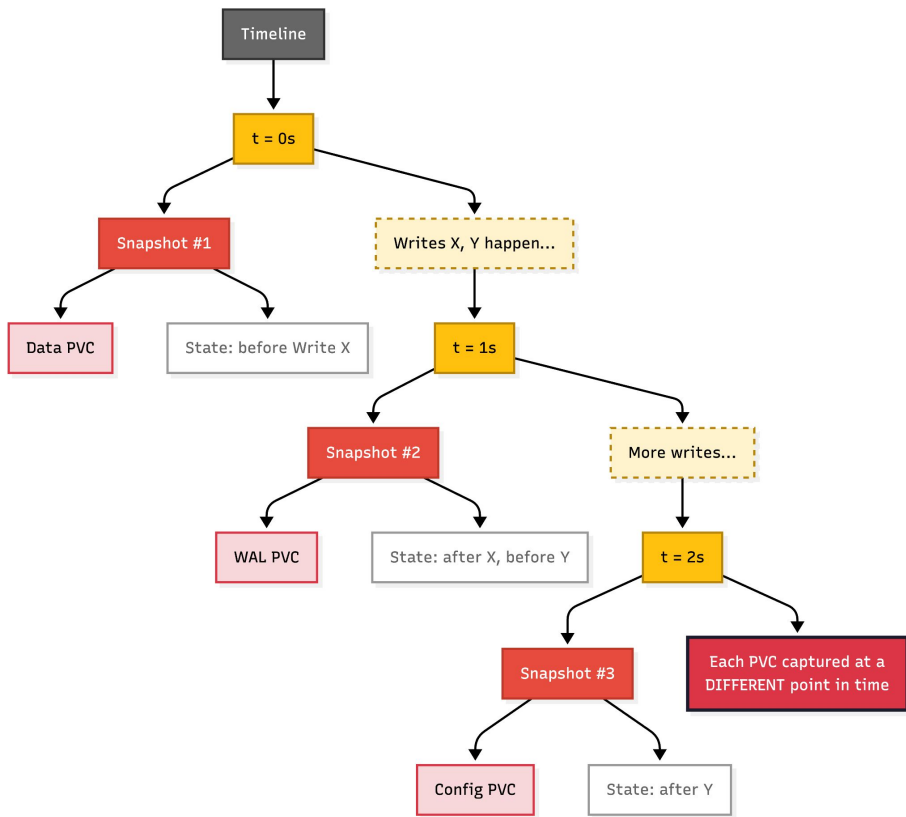


KubeCon



CloudNativeCon

Europe 2026



Act 2: VolumeGroupSnapshot

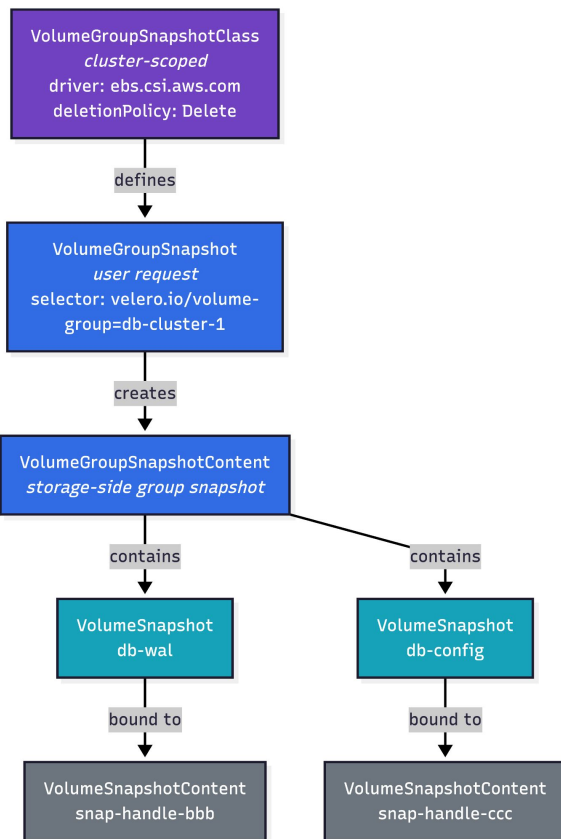


KubeCon



CloudNativeCon

Europe 2026



The Kubernetes Answer (KEP-3476)

Mirrors the existing snapshot pattern:

VolumeGroupSnapshotClass ~ VolumeSnapshotClass

VolumeGroupSnapshot ~ VolumeSnapshot

VolumeGroupSnapshotContent ~ VolumeSnapshotContent

How It Works Under the Hood

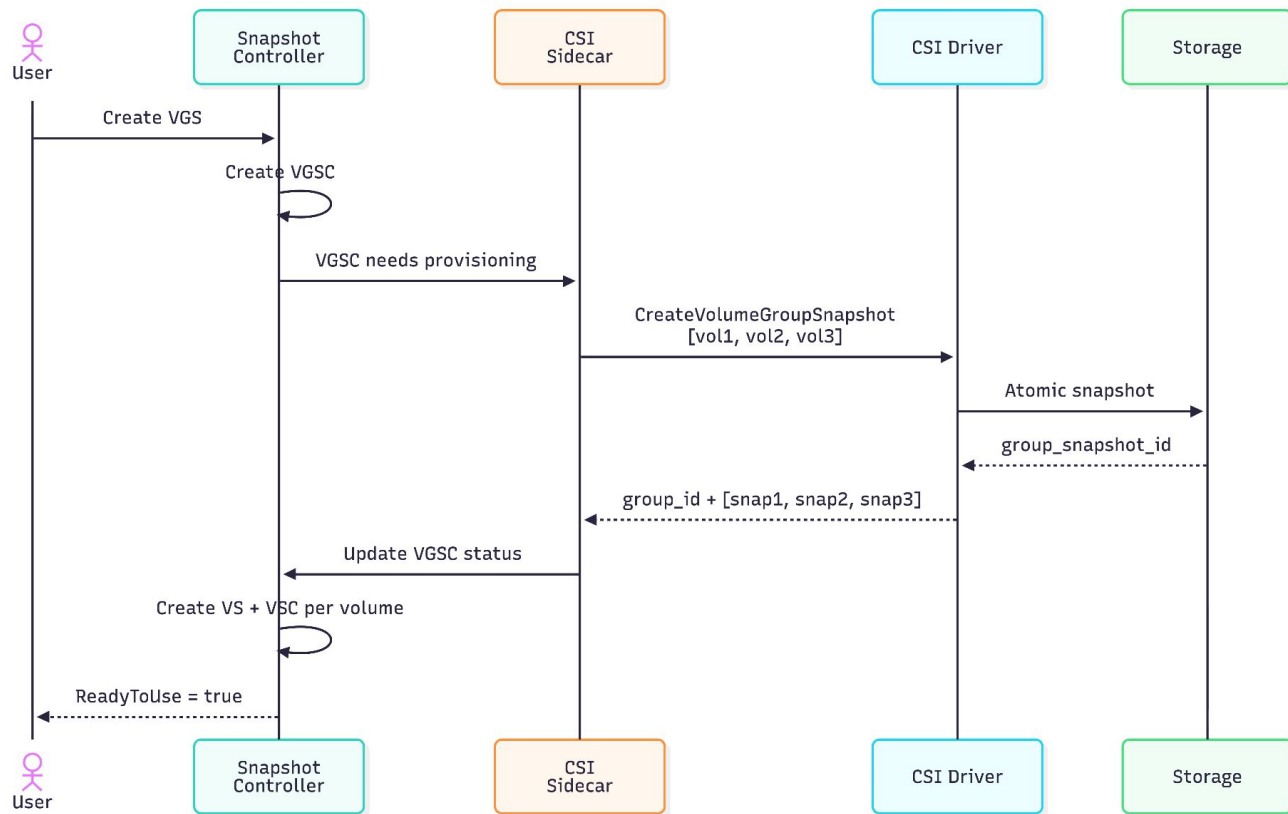


KubeCon



CloudNativeCon

Europe 2026



Crash Consistency vs. Application Consistency



KubeCon



CloudNativeCon

Europe 2026

Crash Consistency (VGS):

- All volumes captured at **same instant**
- Equivalent to pulling the power plug
- Recovery relies on app's crash recovery (WAL replay, fsck, journal replay)
- **Automatic, no app cooperation needed**

Application Consistency:

- Application quiesces writes first
- Flushes buffers, closes transactions
- Snapshot captures "clean" state
- **Requires app-specific hooks**

Best practice: Use VGS for crash-consistent snapshots (nightly).
Add pre/post hooks for application-consistent snapshots (weekly).
VGS handles the **storage** consistency. Hooks handle the **application** consistency.

Act 3: Bringing VGS to Velero - From Design Doc to Production



KubeCon



CloudNativeCon

Europe 2026

The Design Challenge: Velero already had a mature CSI snapshot pipeline.

PVC --> VolumeSnapshot --> VolumeSnapshotContent --> Storage Snapshot

The question: How do we add group snapshot support without breaking the existing flow?

Design principle: VGS should be an **opt-in extension**, not a replacement.

- No VGS label? Everything works exactly as before.
- Add a label? You get atomic group snapshots.
- Same restore flow either way.

“The key insight was: VGS is an orchestration mechanism, not a new backup format.”

Velero Backup Workflow Modes

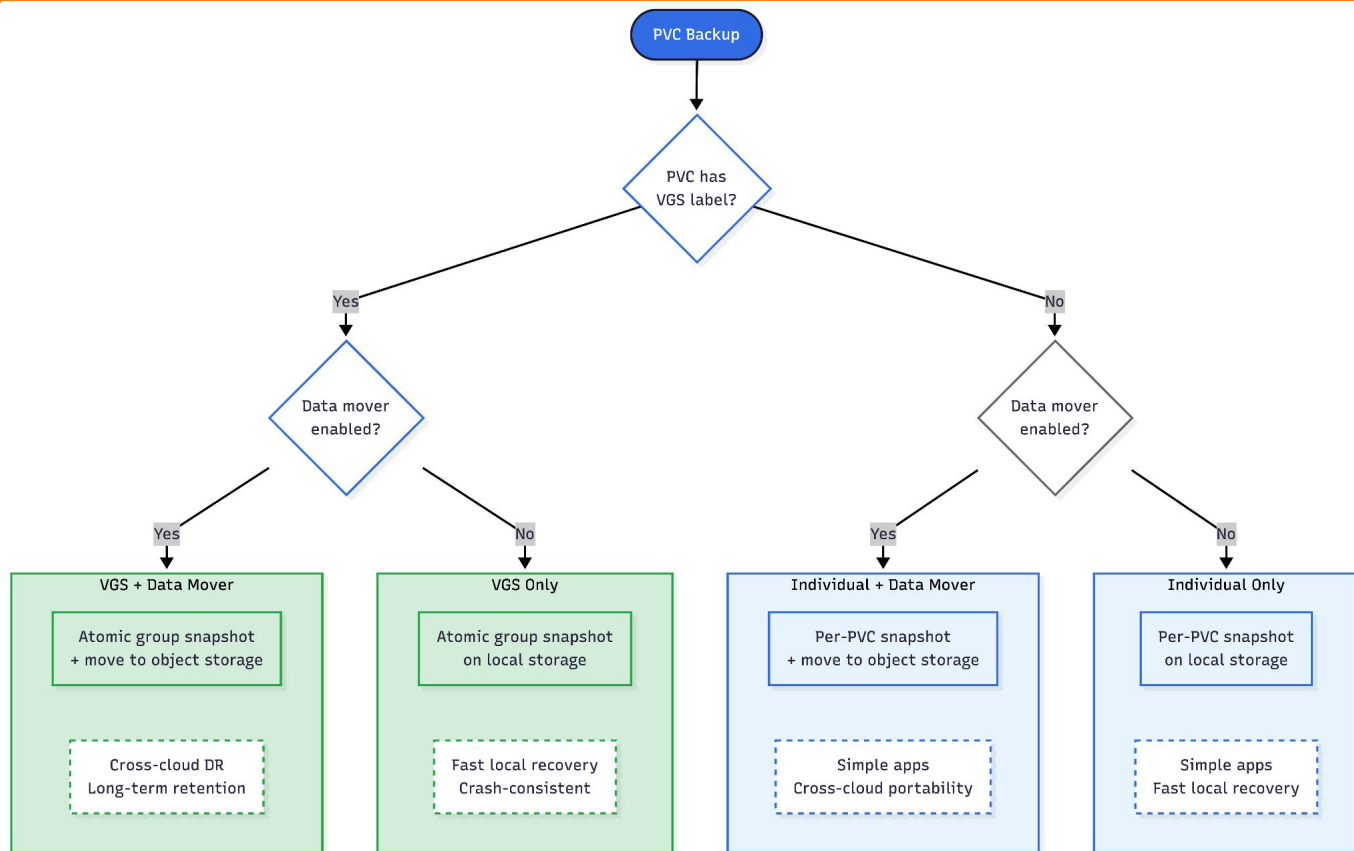


KubeCon



CloudNativeCon

Europe 2026



Velero VGS Backup Workflow Overview

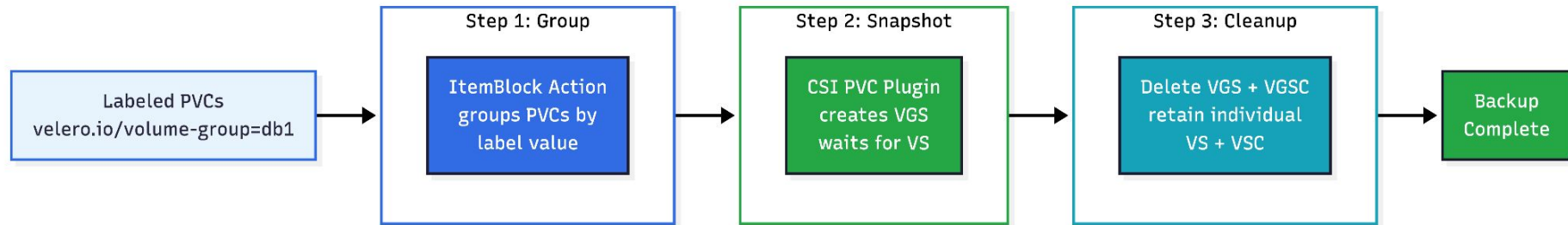


KubeCon



CloudNativeCon

Europe 2026



`velero backup create my-db-backup --include-namespaces my-database`

“Same CLI. No new flags. Just a label on your PVCs.”

VGS Backup Workflow Step 1: Grouping



KubeCon



CloudNativeCon

Europe 2026

PVC: db-data
label: velero.io/volume-group=db1

PVC: db-wal
label: velero.io/volume-group=db1

PVC: db-config
label: velero.io/volume-group=db1

ItemBlock
(processed together)

Create VGS

PVC: app-logs
no VGS label

Individual
Snapshot

```
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: db-wal
  namespace: my-database
labels:
  velero.io/volume-group: db-1 # <-- Same label = same group
spec:
  accessModes: [ReadWriteOnce]
  resources:
    requests:
      storage: 5Gi
```

```
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: db-data
  namespace: my-database
labels:
  velero.io/volume-group: db-1 # <-- Same
label = same group
spec:
  accessModes: [ReadWriteOnce]
  resources:
    requests:
      storage: 10Gi
```

VGS Backup Workflow Step 2: Snapshot

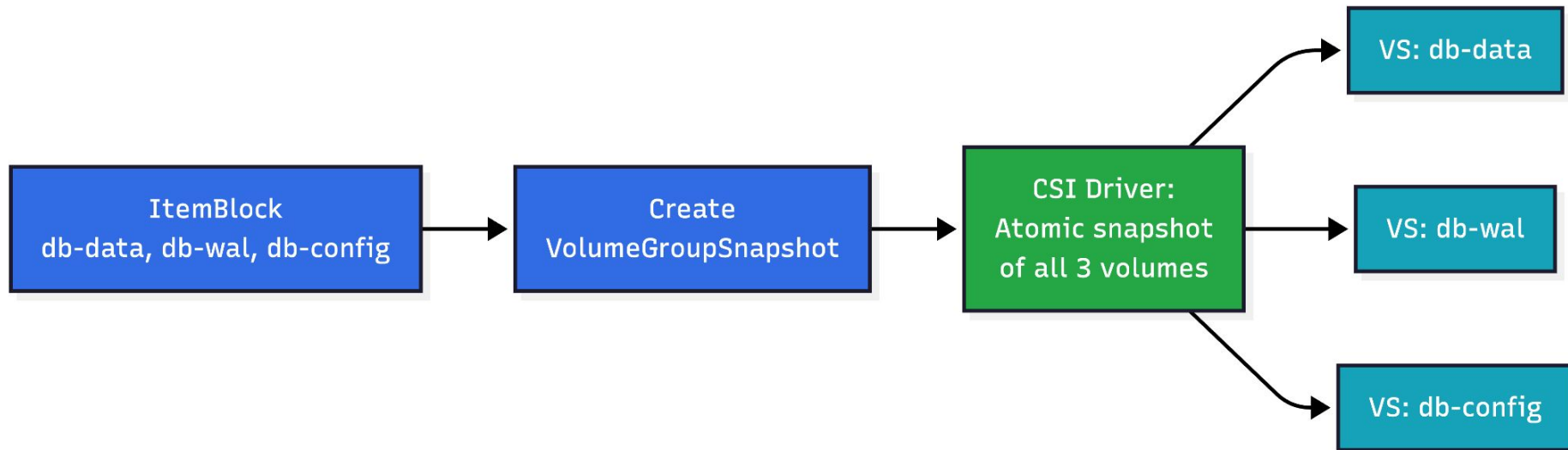


KubeCon



CloudNativeCon

Europe 2026



One CSI call. All volumes. Same instant.

VGS Backup Workflow Step 3: Cleanup



KubeCon



CloudNativeCon

Europe 2026

During Backup

VGS + VGSC
(temporary)

contains

VS + VSC
(per volume)

Delete
VGS + VGSC

After Cleanup

VS + VSC
(per volume)
Standard Velero lifecycle

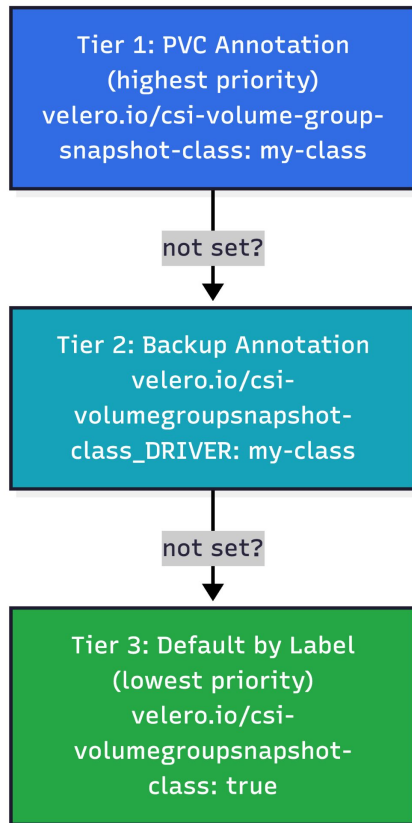
Restore portability: Target cluster typically doesn't need VGS CRDs

Lifecycle management: VS/VSC follow existing Velero deletion flow

No Dangling Orphan Objects: If we kept VGS around, deleting it would cascade destroy all the child snapshots, we don't want that

“VGS is transient by design. It does its job during backup and then gets out of the way”

VGSClass Selection: Three-Tier Precedence



Most users just label their VGSClass (Tier 3) and it works automatically.

Volume Policies + VGS



KubeCon



CloudNativeCon

Europe 2026

What if a group has PVCs on different storage backends?

PVC	Storage	Result
db-data	EBS CSI	Snapshot
config-nfs	NFS	Skip
db-wal	EBS CSI	Snapshot

- **Without volume policies:** Backup fails, "multiple CSI drivers found"
- **With volume policies:** NFS PVC filtered out before grouping. VGS proceeds.

Volume policies are evaluated BEFORE VGS grouping.

Act 4: Real-World Challenges



KubeCon



CloudNativeCon

Europe 2026

War Stories from CSI Drivers

The Ceph RBD Surprise



KubeCon



CloudNativeCon

Europe 2026

We deployed VGS backup on a Ceph RBD cluster. **Backup worked perfectly.**

Then we tried to **restore**.

VolumeSnapshotContent reconciliation failed:

```
"volumeGroupSnapshotHandle" is set but no matching  
VolumeGroupSnapshotContent found"
```

The problem: Ceph RBD's CSI driver populates `volumeGroupSnapshotHandle` on every `VolumeSnapshotContent` that was created via a group snapshot.

During restore, the CSI snapshot controller sees this field and **requires** a matching `VolumeGroupSnapshotContent` to exist, but we deleted it during backup cleanup!

The Solution: Stub VGSCs



KubeCon



CloudNativeCon

Europe 2026

During Backup:

VS backup item action plugin captures VolumeGroupSnapshotHandle as annotation

During Restore:

VS restore item action detects the annotation

Creates a "stub" VGSC with:

- Matching group snapshot handle
- Individual snapshot handles
- Retain deletion policy
- Restore-name label for cleanup

After Restore (Finalization):

Wait for all VSCs to become ReadyToUse

Delete stub VGSCs (no longer needed)

Result: Ceph RBD restore works. Stub VGSCs are temporary scaffolding they exist just long enough for the CSI controller to reconcile, then they're cleaned up.

The Lesson

CSI drivers are not uniform.

The VolumeGroupSnapshot API defines **what** to do.

Each CSI driver decides **how** to do it.

- Some drivers populate `volumeGroupSnapshotHandle`. Some don't.
- Some drivers require VGSC during VSC reconciliation. Some don't.
- Some drivers return snapshot handles immediately. Some are async.

Test across multiple CSI drivers. The API spec is necessary but not sufficient.

Act 5: Results & Blueprint



KubeCon



CloudNativeCon

Europe 2026

Making VGS Work for You

Individual vs. Group Snapshots



KubeCon



CloudNativeCon

Europe 2026

Metric	Individual Snapshot	VolumeGroupSnapshot
Consistency	Per-volume only	Crash-consistent across group
API Calls	N snapshot calls	1 group snapshot call
Failure Isolation	Per-volume	All-or-nothing
CSI Driver Required	VolumeSnapshot support	VolumeGroupSnapshot support
User Effort	None	Add one label per PVC

Multi-volume stateful app? Use VGS.
Single volume or independent volumes? Use individual snapshots.

VGS Performance: Real-World Benchmarks

PVCs	Individual VS	VGS	Speedup
3	19s	15s	1.3x
10	56s	37s	1.5x
20	107s	69s	1.6x
30	160s	108s	1.5x
50	266s	184s	1.4x

Consistency:

- **Individual:** Snapshots captured over 4m 26s at 50 PVCs -- writes between first and last snapshot create inconsistent state
- **VGS:** Single CSI call captures all volumes atomically at the storage layer -- crash-consistent, zero inconsistency window

K8s 1.32, CephFS, 1Gi PVCs, Velero with VGS support

VGS Backup Demo



KubeCon



CloudNativeCon

Europe 2026

```
VGS Backup Monitor

Backup: demo-egs-backup InProgress [4s]
Namespace: demo-namespace

└─ Create ──► VGS ──► Snapshots ──► Cleanup ──► Finalize ──► Done

VGSClass: demo-egs-backup-creating-groovy-classes [velero.io/csi]-volume-group-snapshot-class=itag

VGSs [velero.io/volume-group-snapshot=3]
├─ db-config Sound
├─ db-data Sound
├─ db-sql Sound
└─

└─ VGS: velero-csi-snapshot-1-k8s ReadyToUse: Pending (transient)
  └─ VGS: groovy-egs-backup-335720cb-2b29-42ff.... DeletionPolicy: Delete (transient)

└─ VolumeSnapshots (2 created, 1 pending)
  └─ snapshot-b52a07f2a65e99ad1c88b1d03a79.... ReadyToUse: Pending
  └─ snapshot-23d672a2b63b84e0e0e1e73e2a465.... ReadyToUse: Pending

4 VGSs + VGS created, waiting for snapshots...
```

Blueprint: Adopting VGS Today



KubeCon



CloudNativeCon

Europe 2026

Prerequisites Checklist

1. Kubernetes 1.32+ (VGS API beta)

kubectl version

2. VGS CRDs installed

kubectl get crd | grep volumegroup

3. CSI driver supports VGS

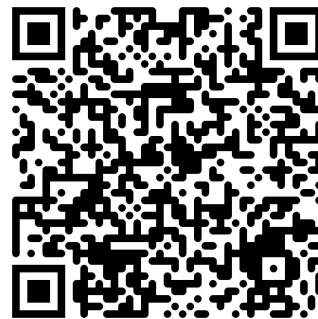
kubectl get volumegroupsnapshotclass

4. Default VolumeSnapshotClass exists (required by snapshot controller)

kubectl get volumesnapshotclass -o wide

5. Velero 1.17+ installed

velero version



Velero VGS Docs

The Journey: Design to Production



KubeCon



CloudNativeCon

Europe 2026

2025 Mar Design proposal
|
2025 May PVC grouping via ItemBlocks
|
2025 Jun Core VGS backup workflow
|
2025 Jul Documentation & user guide
|
2025 Nov Volume policy integration
|
2026 Feb Ceph RBD restore fix
|
2026 Mar CNCF TOC vote passed for Sandbox !

*Adopted VGS while still beta in
Kubernetes 1.32, early adoption
uncovered the real-world challenges
we shared today.*

What's Next



KubeCon



CloudNativeCon

Europe 2026

CNCF Sandbox: Velero CNCF Sandbox onboarding in progress !

VGS GA target in Kubernetes 1.36: Moving from beta to stable

Broader CSI driver testing : Validating across more storage backends

Application-consistent VGS: Combining pre/post hooks + VGS for full consistency

Get Involved:

- Velero: github.com/vmware-tanzu/velero
- KEP-3476 for the upstream API
- CSI driver-specific issues to driver maintainers

Try VGS in your environment. File issues. That's how every fix in this talk happened.

Key Takeaways



KubeCon



CloudNativeCon

Europe 2026

1. **Multi-volume apps need atomic snapshots:** Individual snapshots create inconsistent backups that fail on restore
2. **VolumeGroupSnapshot is the answer:** Crash-consistent, atomic snapshots via the Kubernetes API
3. **Velero makes it simple:** Just label your PVCs. Same CLI, same restore flow.
4. **VGS is transient by design:** Atomic snapshot during backup, standard Velero lifecycle after
5. **Test across CSI drivers:** The API spec is necessary but not sufficient



KubeCon



CloudNativeCon

Europe 2026

Thank You !

Shubham Pampattiwar: @shubham-pampattiwar

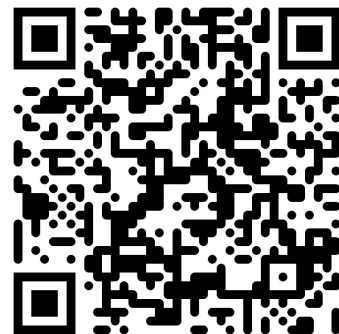
Scott Seago: @sseago

Velero: github.com/vmware-tanzu/velero

Come find us after the talk!



Session Feedback



Velero Repo