



VOPipe on VOSpace

Tamás Budavári, Dmitry Mishin,
Alex Szalay, Richard Wilton
The Johns Hopkins University





VOSpace on Swift

The OpenStack Object Storage System



What is Swift?

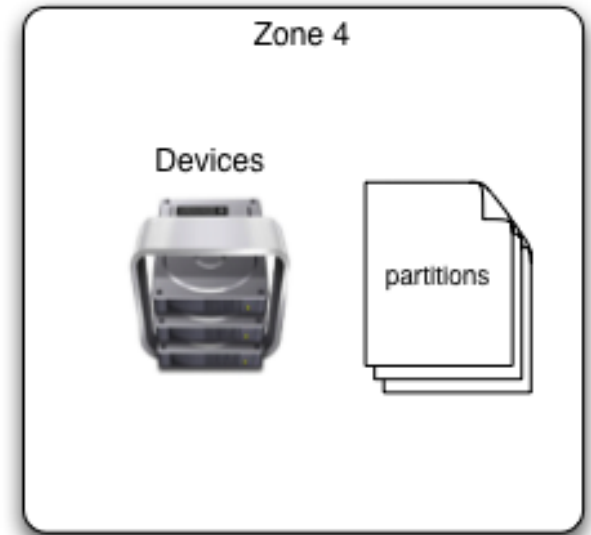
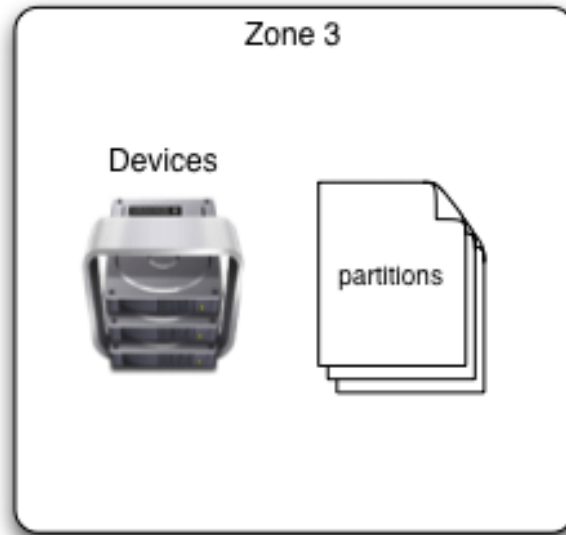
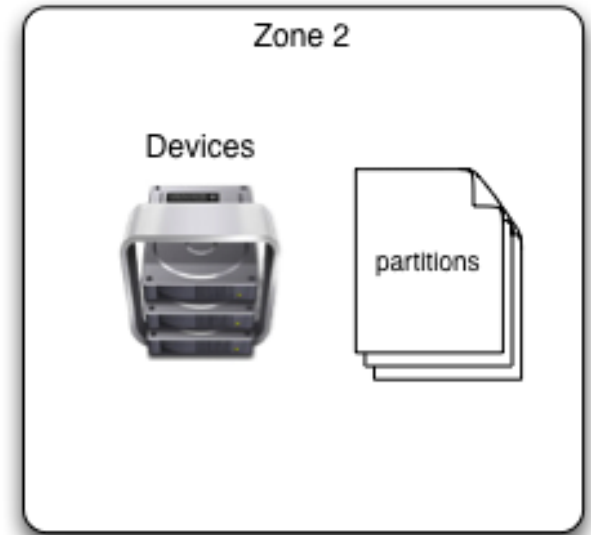
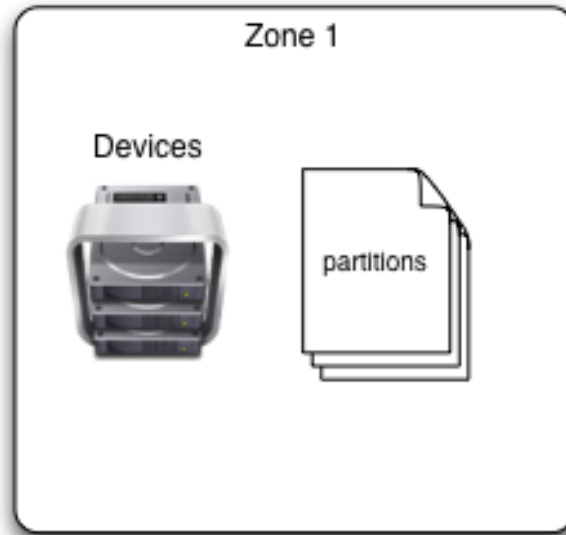
OpenStack Object Storage is software for creating redundant, scalable object storage using clusters of commodity servers to store terabytes or even petabytes of accessible data

<http://swift.openstack.org/>

<https://launchpad.net/swift>

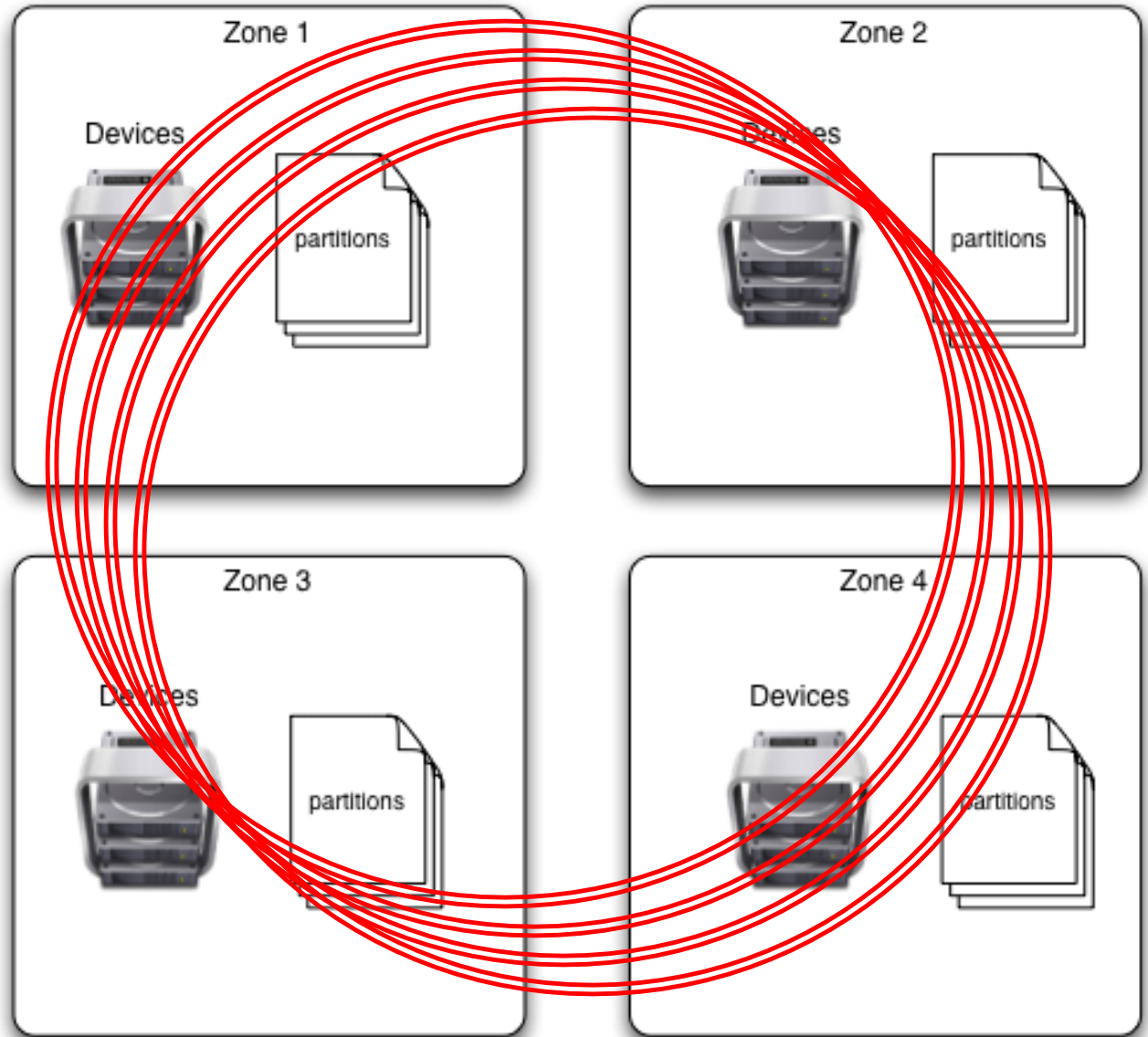


Swift Architecture



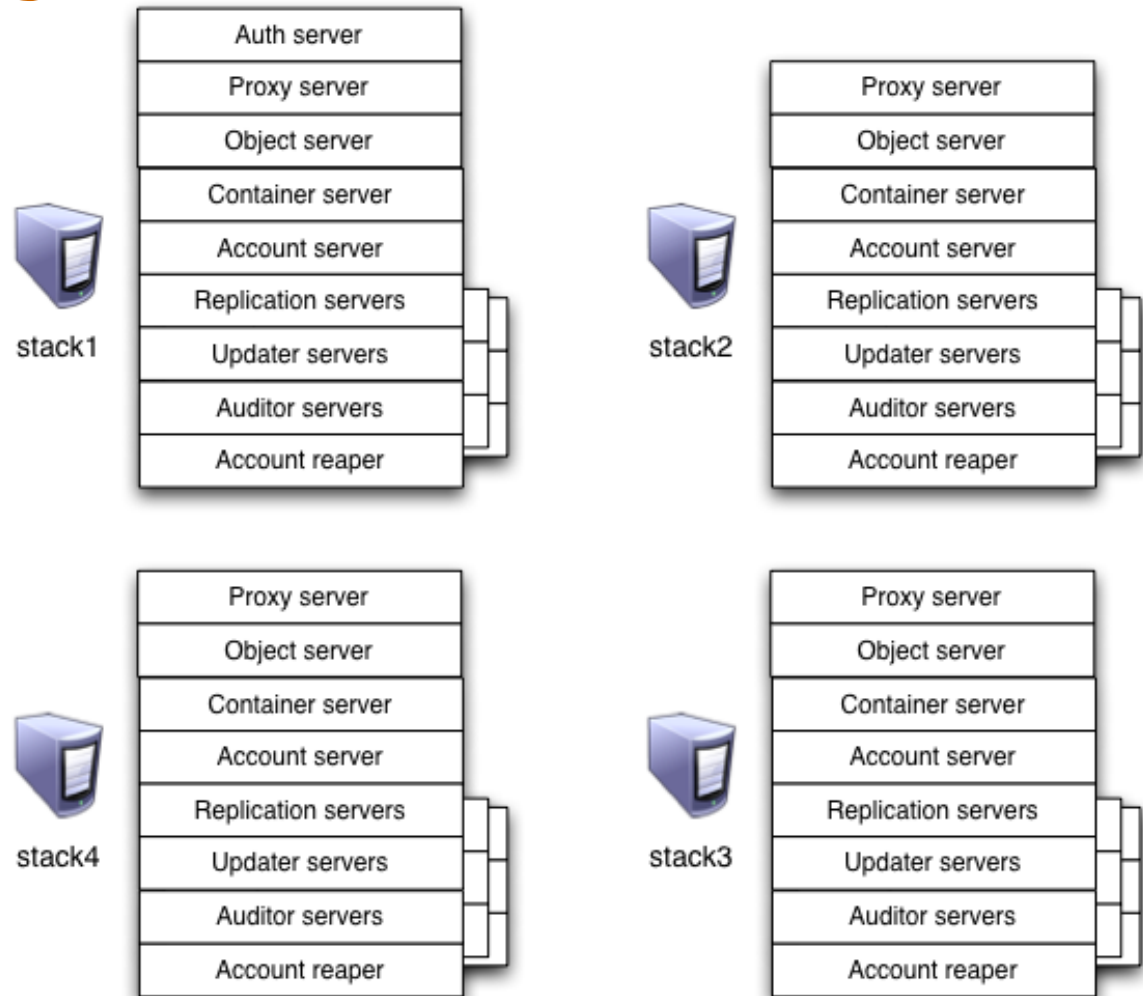


Swift Architecture





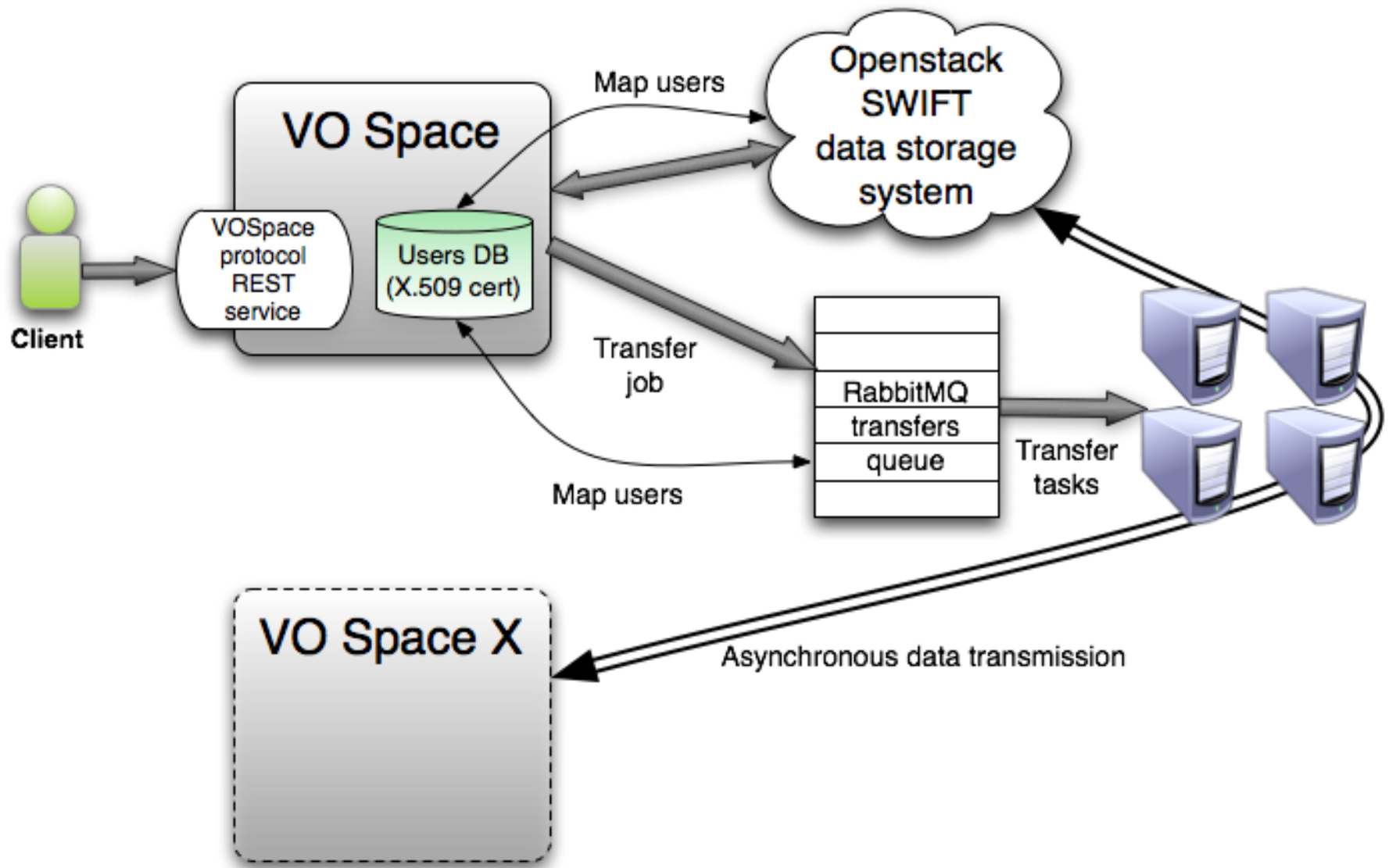
Swift Servers





VOSpace 2.0

By Dmitry Mishin





Test Widgets

pha.jhu.edu tasks queue

id	state	start time	end time
33f17bfc-7d9f-4c8d-bcd9-1df87c0bf68c	COMPLETED	2010-12-03 20:31:36 UTC	2010-12-03 20:31:36 UTC
PULLFROMVOSPACE	vos://pha.jhu/cont1/cloudfiles.properties		

pha.jhu.edu swift cluster

- pha.jhu.edu
 - 12345
 - 333333333
 - cont1
 - cont2
 - 109
 - 1234
 - Zoom
 - .DS_Store
 - 1
 - frame-000109-1-301-0011-z00.jp2
 - frame-000109-1-301-0011-z12.jp2
 - frame-000109-1-301-0011-z25.jp2
 - frame-000109-1-301-0011-z50.jp2
 - frame-000109-1-301-0012-z00.jp2
 - frame-000109-1-301-0012-z12.jp2
 - frame-000109-1-301-0012-z25.jp2
 - frame-000109-1-301-0012-z50.jp2
 - frame-000109-1-301-0013-z00.jp2
 - frame-000109-1-301-0013-z12.jp2
 - frame-000109-1-301-0013-z25.jp2
 - frame-000109-1-301-0013-z50.jp2



Editing metadata

4c8d-bcd9-	COMPLETED	2010-12-03 20:31:36 UTC	2010-12-03 20:31:36 UTC
ACE		vos://pha.jhulcont1/cloudfiles.properties	

- 1
- 123
- 2
- IMG_0862.CR2
- cloudfiles.properties

```
cont1/cloudfiles.properties
<node xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="vos:UnstructuredDataNode" uri="{url}">
  <properties />
  <accepts>
    <view uri="ivo://ivoa.net/vospace/core#anyview" />
  </accepts>
  <provides />
  <capabilities />
</node>
```

Save Cancel



VOPipe

Asynchronous dataflow

Service orchestration



Scientific Motivation

- ✚ Stream data efficiently using VOSpaces
 - ✚ Like async messaging but with performance
 - ✚ E.g., transport between SkyQuery engines

- ✚ Chain VOSpaces into data-flows
 - ✚ Easy dynamic setup for specific tasks
 - ✚ E.g., solving a crossmatch problem



VOPipe for Data Flows

✚ Simple interface

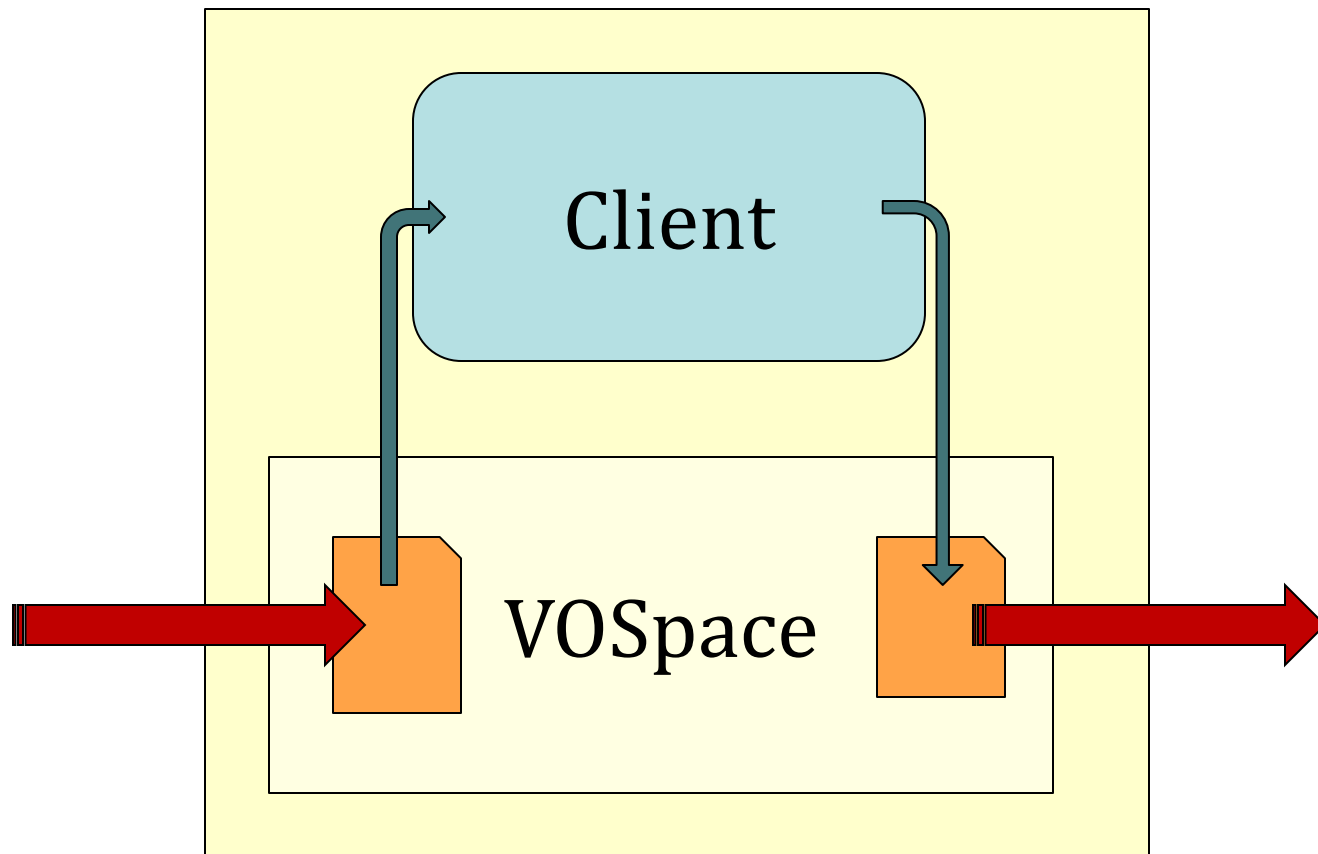
- ✚ CreatePipe (src,dest)
- ✚ DestroyPipe (pipe)
- ✚ WriteChunk (pipe,name)
- ✚ ReadChunks (pipe,timeout,state)

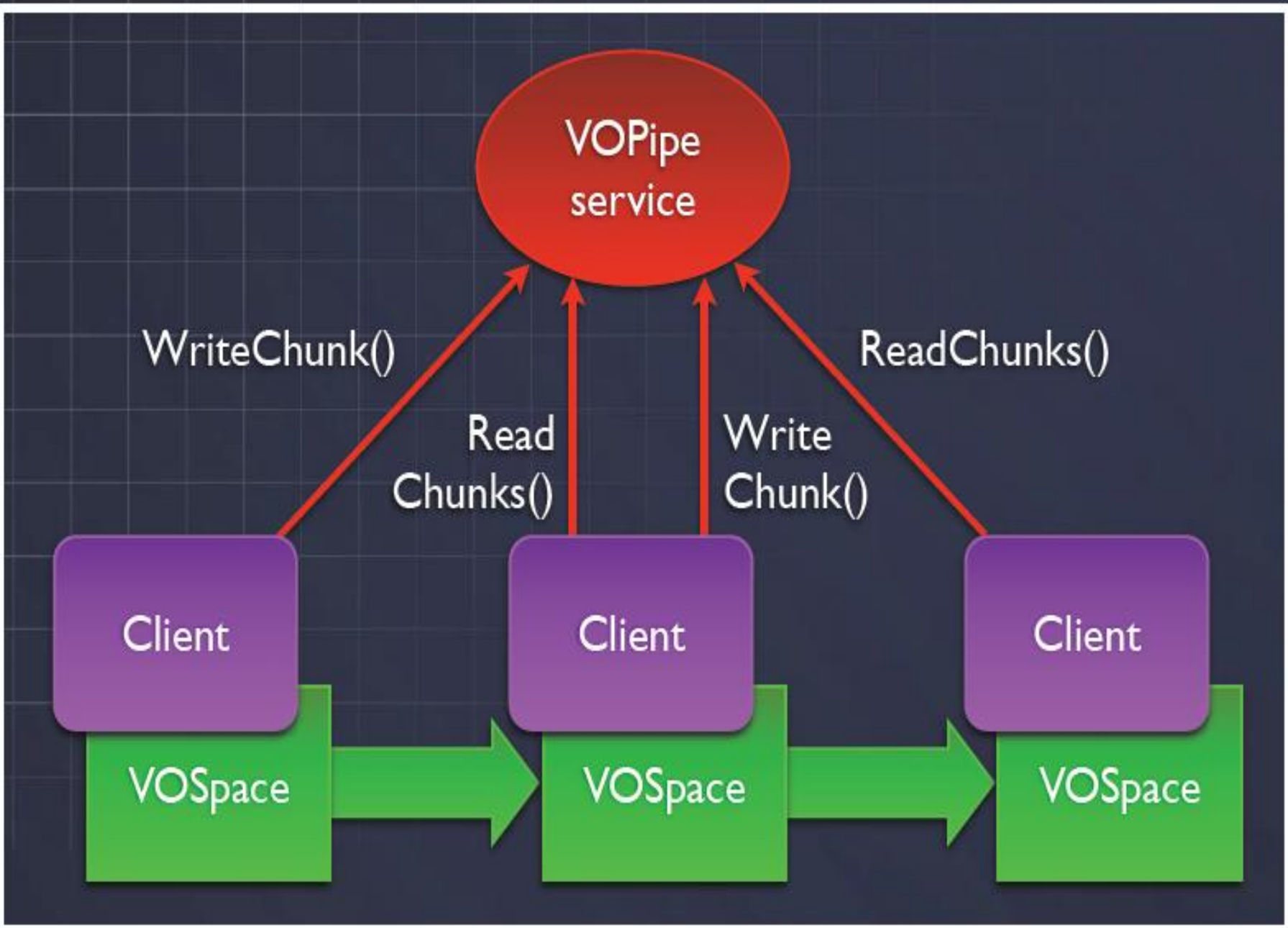
✚ Implementation

- ✚ VOSpace 2.0 is the highest priority now!
- ✚ Three (almost) fully functional codes



In Action







Summary

- ✚ VOSpace 2.0 is here!
 - ✚ Need to test interop: CADC, Caltech, JHU, anyone?
- ✚ VOPIpe? The time is ripe!
 - ✚ Reliable async dataflow between VOSpace nodes
 - ✚ Architecting services: work on “local” data
- ✚ Essential for large-scale data science!