Enabling the petabyte-sized mailboxes
7 fellow co-founders
8 years in the field
50 million users
2015 launch
2006 idea & first concept
2007 design dev starts
2009 1st massive production
2012 the project is open sourced
2014 10+ PB managed
2015 OpenIO fork
3 avenue Antoine Pinay
Parc d'Activité des 4 Vents
59510 Hem

USA
180 Sansom Street, Fl 4
San Francisco, 94104 CA
Summary

- **Cyrus scalability issues**
  - Storage infrastructure challenge
- **OpenIO technology**
  - Open source object storage
- **Cyrus IMAP 3.0**
  - OpenIO for a scale-out Cyrus
Cyrus scalability issues

- Migrate mailboxes to new clusters as local storage becomes full
- Storage cost with SAS drives
- Ends up with unused CPU/RAM resources on Cyrus servers

Share-nothing clusters
Be smart with storage
Email distribution by size (based on actual data)
Problem: more storage you need, more servers you waste

Over the years

Storage

CPU/RAM usage

Over the years
Solution: abstraction between **Cyrus servers and storage**
Object storage
Object storage magic

The simplest and best way from 1 hard drive to 1,000s
How to keep track of trillions of objects?
Existing technologies

**Single name node**

- Good for few large files (large web indexes, nosql tables, …)
- Bad for numerous small objects (emails,…)

**Distributed Hash Tables Consistent Hashing**

- Good for trillion of objects
- Bad because of rebalancing of part of the data when adding capacity
OpenIO is different
Trillions of objects? Let’s do the math

$1,000,000,000,000 = 100,000,000 \times 10,000$

Mailboxes  Emails per Mailbox

Do not track objects, track containers!
OpenIO technology highlights
Directory design with indirections  

Grid & Conscience
“All problems in computer science can be solved by another level of indirection.”

David Wheeler
Indirections
Pragmatic design

Containers store locations of objects, not objects
Grid & Conscience
On-the-fly best matchmaking

Grid of nodes

- Massively distributed
- Each node takes part in directory & storage services
- no SPOF - Resilient to node failures

Conscience

- Collects metrics of each node
- Computes node scores
- Real time asynchronous process
- Advanced load balancing: select the best nodes at a particular time route requests to them
#NeverRebalance
Real life use-case
deploy as you grow

ISP 30+ M mailboxes
Cyrus 3.0 + OpenIO
Cyrus 3.0 archive pool
Cyrus 3.0 + OpenIO

Steps

#1 Scale out archive storage

Available

#2 Bring Conscience to Cyrus Murder

Sept 2015

To provide the best Cyrus IMAP node when a mailbox is created or migrated

#3 Scale-out backend for calendars & contacts

End of 2015

To provide per-user tiny databases spread across the grid of nodes
Grid for Apps
scale-out back-ends for your apps
Get started
atlas.hashicorp.com/openio/boxes/sds-cyrus/

Get the Vagrant box

$ vagrant up