SFD19 - Stellus
Embargo February 5th

*Jeff Treuhaft, CEO

Founded 2016
Data Path architecture 2017
Data path validations 2018
First customer deployments 2019
Commercial availability 2020

Rise in unstructured data (driven in large part by AI / ML)

Exponential increase in size of sources

Increases in performance requirements for unstructured data
- life sciences
- Media and entertainment
- IIoT

Need
- scalable, consistent performance
- common global namespaces
- variable file sizes
- high throughput
- no parallel access penalties
- easily manages data over time
- data system of record

Current competitors have built legacy file systems at the time when spinning disk and building private DCs were the focus
[file systems age - photo]

Stellus Data Platform: Scalable Performance and Capacity
- every client connection “sees” every key value store
- high performance, low latency KV fabric
- Large #s of application specific KV stores
- Massively parallel “data computer”
- scales nearly linearly for reads and writes
- Maintains consistency of performance under load

*Bala Ganeshan, CTO and VP of engineering
Stellus Data Platform

Design Goals
- parallelism
- scale
- throughput
- constant performance
- decoupling capacity and performance

- independently scale perfromance and capacity on commodity hardware
- distributed all, share everything KV based data model data path ready for new memories
- consistently high performance even as system scales

File System as Software [photo]
- stores unstructured data closest to native format: objects
- Data Services provided on Stellus objects
- Stateless - state in Key Value Stores
- user mode enables
-- on premises
-- cloud
-- hybrid
- independent from custom hardware and kernel

Don’t have deduplication capability built in

Algorithmic Data Locality and Data Services
- enables scale by algorithmically determining location - no cluster-wide maps
- built for resilience to multiple failure - pet vs. cattle
- understands topology of persistent stores
- architecture maintains versions - enables data services such as snapshots

Key-Value-over-NVMe Fabrics
- decoupled data services and persistence requires transport
- architecture maintains native data structure - objects
- NVMe-over-Fabric protocol enhanced to transport KV commands
- Transport independent
-- RDMA
-- TCP/IP

Native Key-Value Stores
- unstructured data is generally immutable
- updates result in new objects
- available in different sizes and performance characteristics
- we used application-specific KV stores, such as:
-- immutable data
-- short-lived updates
-- metadata

*Jeff

Stellus High-performance Unstructured Data Platform [photo]

Industry use cases
M & E Workflow [photo]
Life Sciences Workflow [photo]