

ZFS Day

October 2, 2012

Matt Ahrens

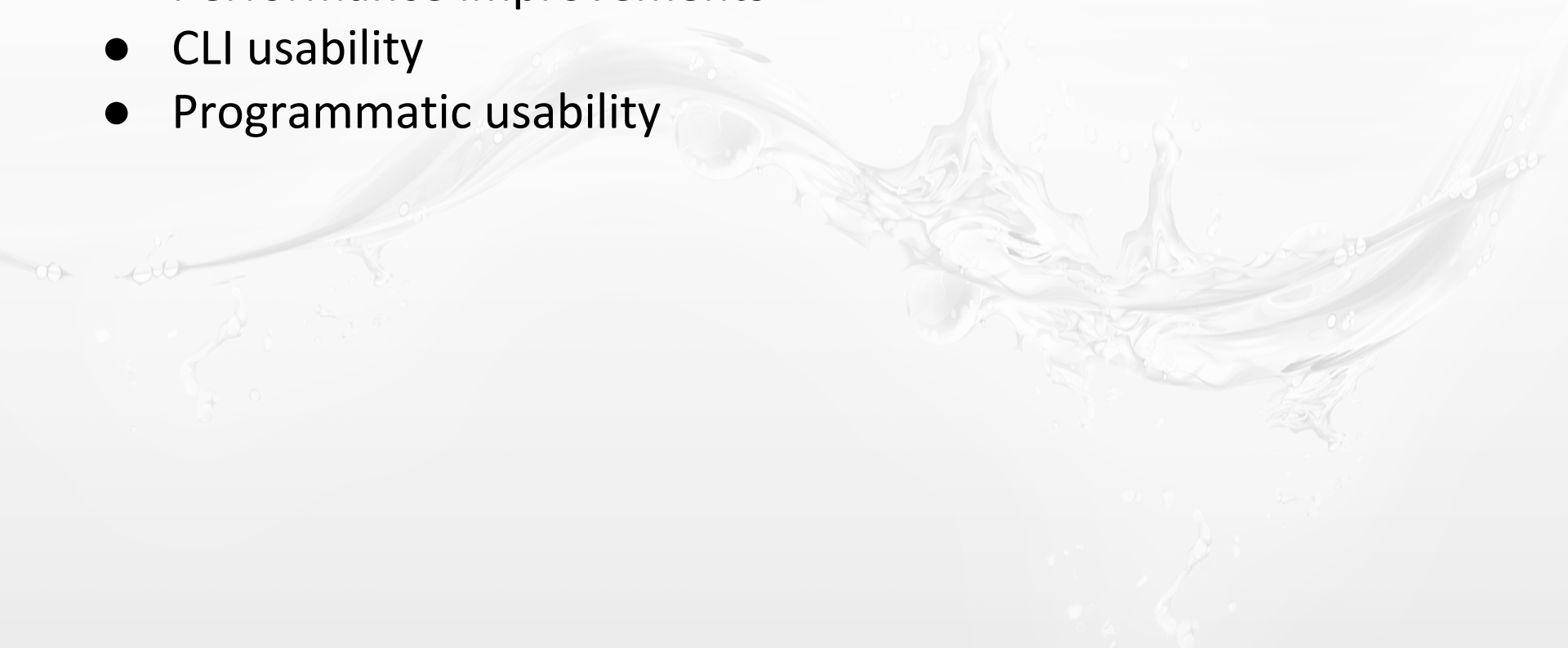
matt@delphix.com

[@mahrens1](https://twitter.com/mahrens1)

ZFS History

- 2001: development starts with 2 engineers
- 2005: ZFS source code released
- 2006: ZFS on FUSE for Linux started
- 2008: ZFS released in FreeBSD 7.0
- 2008: Sun's 7000 series ZFS Storage Appliance ships
- 2010: Oracle stops contributing to Open ZFS
- 2012: ZFS on (native) Linux release candidate
- Open Source has made it possible for all these platforms and businesses to base on ZFS
- And that's why we're here today for ZFS Day

Features unique to Open ZFS

- Space accounting
 - Debugging and testing improvements
 - Performance improvements
 - CLI usability
 - Programmatic usability
- 

Space accounting - only in Open ZFS

- refratio property
- written and written@... properties
- zfs **send stream size estimation**
- zfs send **progress reporting** (Bill Pijewski)
- zfs destroy -nv <snapspec> tells how much space is shared by list of snaps

Debugging & testing - only in Open ZFS

- zfs ioctl args in truss (Gordon Ross)
- ztest backwards compatibility option (Chris Siden)
- **ZFS test suite** returned to working order
 - many tests added
 - new, simpler test framework developed
 - (John Kennedy)
- ztest watchpoints
- more robust zfs mdb module
 - always find the right types
- more complete “zpool history” logging

Performance - only in Open ZFS

- **single-copy ARC** (George Wilson)
- imbalanced LUNs performance improvements (George Wilson)
- zfs destroy <snapspec> destroys many snaps at once
- empty_bpobj feature
- **background destroy** of filesystems, >100x faster destroy of clones (Chris Siden)
- improved performance for **partial-block writes** (in Spectra Logic)
- per-zone i/o throttling (in SmartOS)

CLI Usability - only in Open ZFS

- “zfs get -t <type>” (Andrew Stormont)
- “zpool iostat” separates out log devices (Mark Harsch)
- “zfs get” takes mountpoint (Sham Pavman)
- zpool “comment” property (Dan McDonald)
- per-vdev space usage & expandsz (George Wilson)
- zdb manpages written (Richard Lowe)

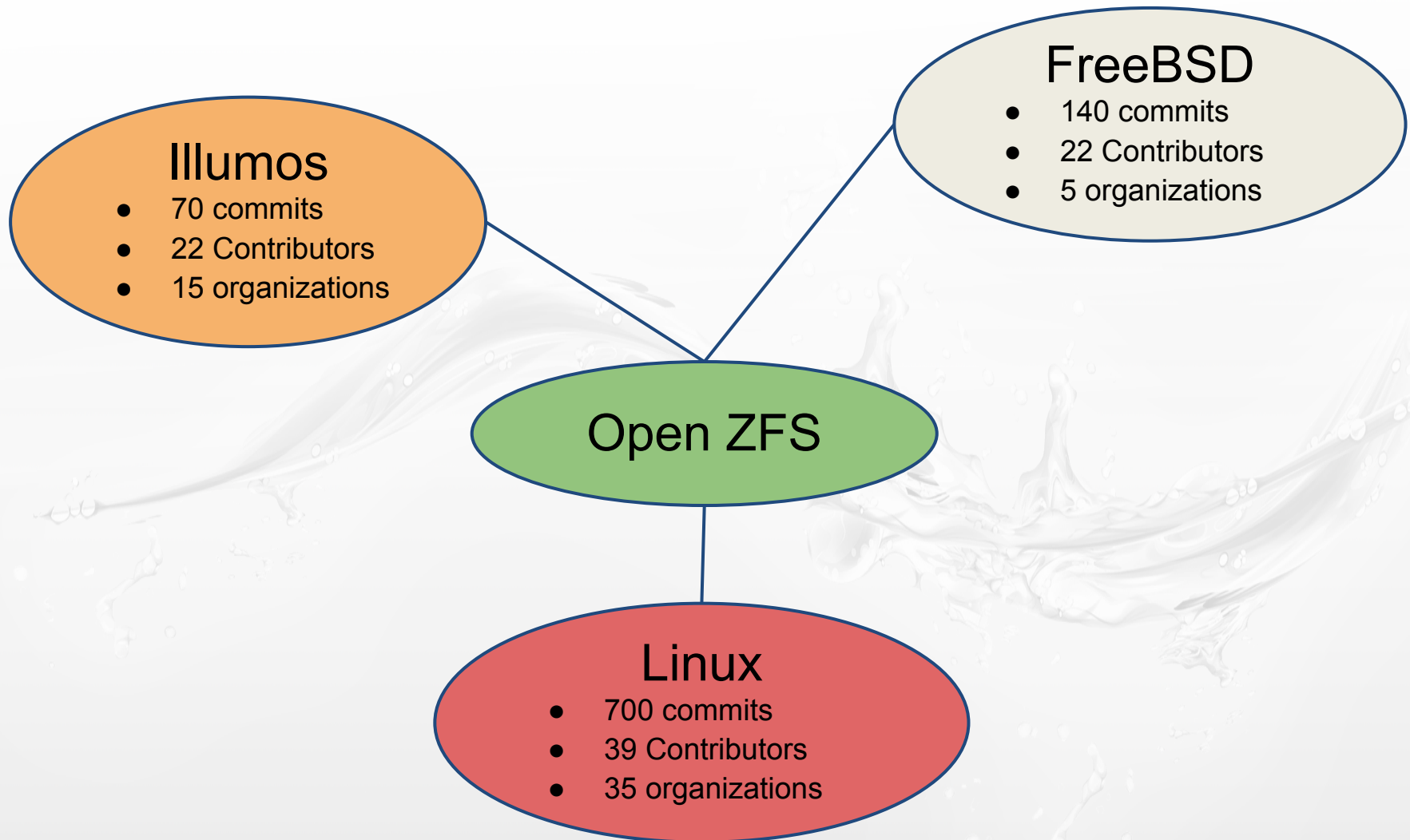
Programmatic Usability - only in Open ZFS

- clones property
- **libzfs_core**
- zfs snapshot <arbitrary snaps>
- zfs destroy <arbitrary snaps>
- background destroy of filesystems

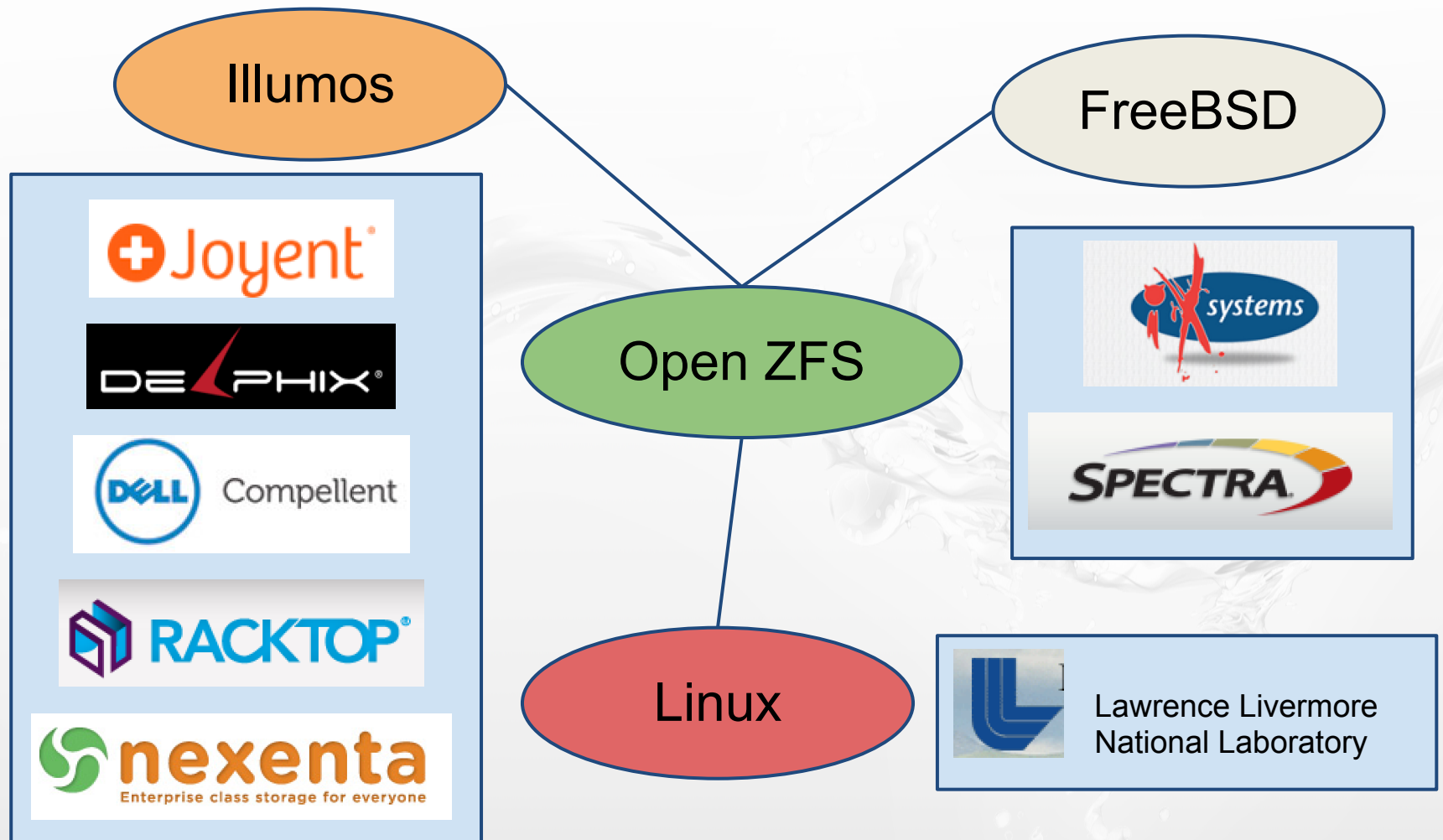
Features - only in Open ZFS

- feature flags (Chris Siden & Basil Crow)
- zpool reguid (Garrett D'Amore & George Wilson)
- dump to RAID-Z (in SmartOS)
- restore “aclmod” property (Albert Lee)
- SCSI UNMAP support (Dan McDonald)
- TRIM support (in Linux and FreeBSD)
- support for **4k sector size devices** (George Wilson)
- SPA i/o deadman (George Wilson)

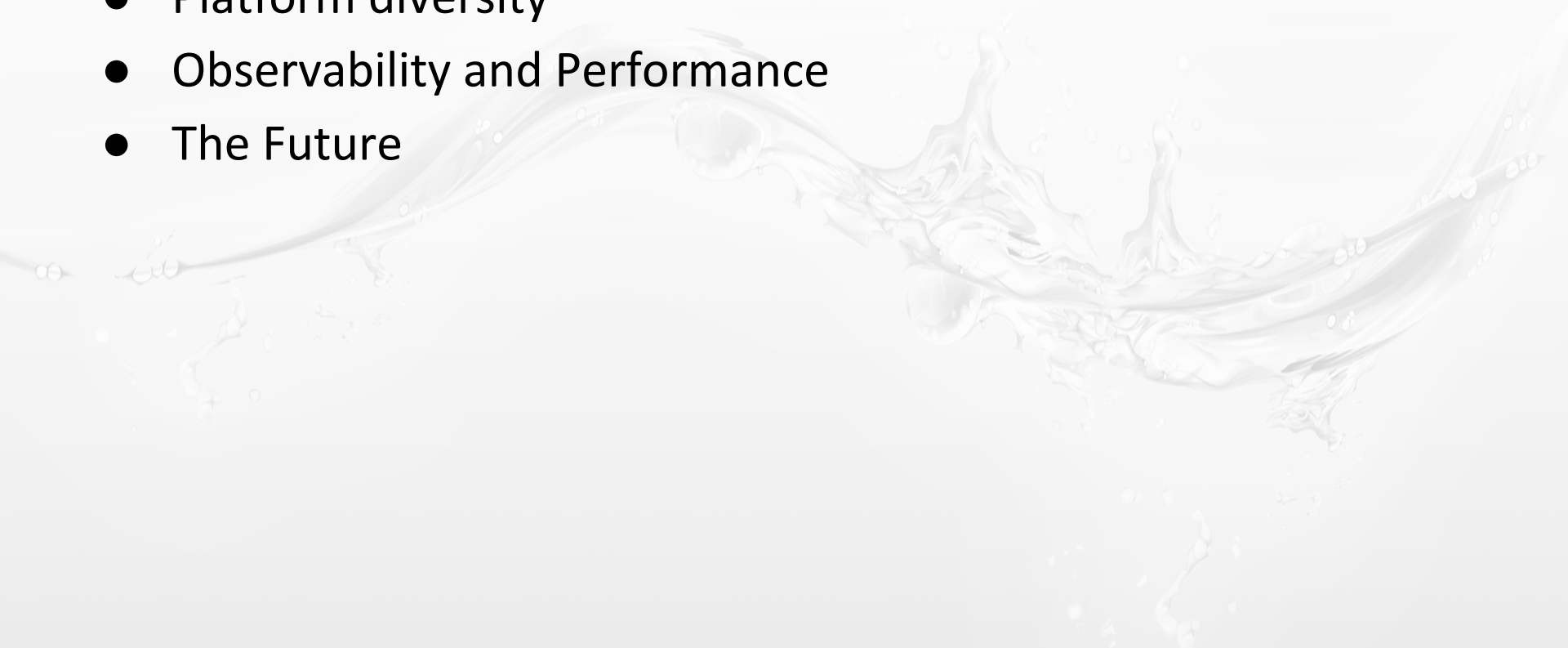
Platform Diversity on Open ZFS



Platform Diversity on Open ZFS



ZFS Day

- Companies have based their business on ZFS
 - How does ZFS make them successful?
 - Platform diversity
 - Observability and Performance
 - The Future
- 

ZFS Send and Receive

- NFS + low-bandwidth, high-latency link + rsync
 - result: thumb twiddling
- ZFS send + receive
 - latency-insensitive (unidirectional)
 - quickly finds & sends only the changed blocks / files
- Remote replication

ZFS Send progress monitoring

- 2005: incremental zfs send
 - just wait a while
- 2011: zfs send size estimation
- 2012: zfs send progress reporting
 - from Bill Pijewski of Joyent
- Today: extra fancyness



The future of Open ZFS

- lz4 compression (Saso Kiselkov)
- compressed lzarc (Saso Kiselkov)
- performance on fragmented pools (George Wilson)
- observability -- zfs dtrace provider
- wild, application-specific solutions
 - easily extensible architecture
 - modern, object oriented implementation
 - enables things like cross-platform provisioning

ZFS Day

October 2, 2012

Matt Ahrens

matt@delphix.com

[@mahrens1](https://twitter.com/mahrens1)