An Overview of Sirocco

Matthew L. Curry
Center for Computing Research
Sandia National Laboratories
Albuquerque, NM, USA
mlcurry@sandia.gov

Also: Lee Ward (SNL), Geoff Danielson (SNL), Anthony Skjellum (Auburn University), Jay Lofstead (SNL)

Parallel Data Storage Workshop (PDSW15)

16 November 2015
EXASCALE IS COMING
Why We Need a Revolutionary Design

- Static organization is not optimal for performance
  - Striping can cause hotspots, coupling
  - Can only optimize placement coarsely, if at all
- POSIX semantics hurt performance
  - Global shared memory
    - False sharing
    - Consistency semantics
  - Attributes
- Need richer I/O modes for more varied applications
Our Answer – A Clean Sheet Redesign

- A two-part system:
  - The Sirocco Object Store (SOS) – A low-level, hierarchical, fixed-depth object storage system
    - Superset of ASG API, developed jointly with ANL
    - Fine-grained transaction support
  - Smart clients that expose user APIs – E.g., POSIX, HDF, S3, etc.
Our Answer – A Clean Sheet Redesign

- LWFS-inspired philosophy
  - Clients bring/opt-in to services they require
- Peer-to-peer inspired design
  - Data and location(s) are decoupled
  - Greedy optimization of QoS (network, storage, reliability)
  - Popularity drives copy creation
Data Moves to Ensure Safety

- Data is written immediately to fast, close stores.
- Alternate stores can be selected for immediate safety, or if close stores are overloaded.
- Servers collaborate with neighbors to ensure data safety.
- As storage fills in fast tiers, data is ejected into safer servers.
Conclusion

- Sirocco is a significant departure from traditional PFS design
  - LWFS- and P2P-inspired
- Designed for write performance first, read performance a distant second, and almost not at all for legacy concerns
  - Necessary evils, etc.