

# Specifying Workflows

Lance M Evans

Cray Inc,

2016-05-03

# Typical I/O Subsystem



COMPUTE

STORE

ANALYZE

# Customer Workflow Specifications

- **Every workflow is unique**
  - Each vertical market is similar within (but never identical)
  - Storage and I/O are called out when something is wrong
  - Devil's in the details
- **Customer knowledge varies**
  - May “think” they know how data flows through their systems
  - May not know about opportunities for improvement
  - Some consider their workflow a differentiator
  - HPC users run similar well-tuned workloads repeatedly
  - Analytics users are usually highly aware of workflow

# Use Cases

- **All-Read Query**

- Absorbs and preprocesses constant sensor data to a staging area
- Loads massive amounts of data into a quantity of SSD servers
- Perform parallel queries against massive servers
- Expunge data when it is stale, and repeat

- **GPU Load**

- Generate a video & photo data set with millions of images, 100s of GB
- Load identical data sets into hundreds of computers at once
- Iteratively process data through machine learning algorithms
- Synchronize many parallel activities and verify convergence

- **Checkpoint and More**

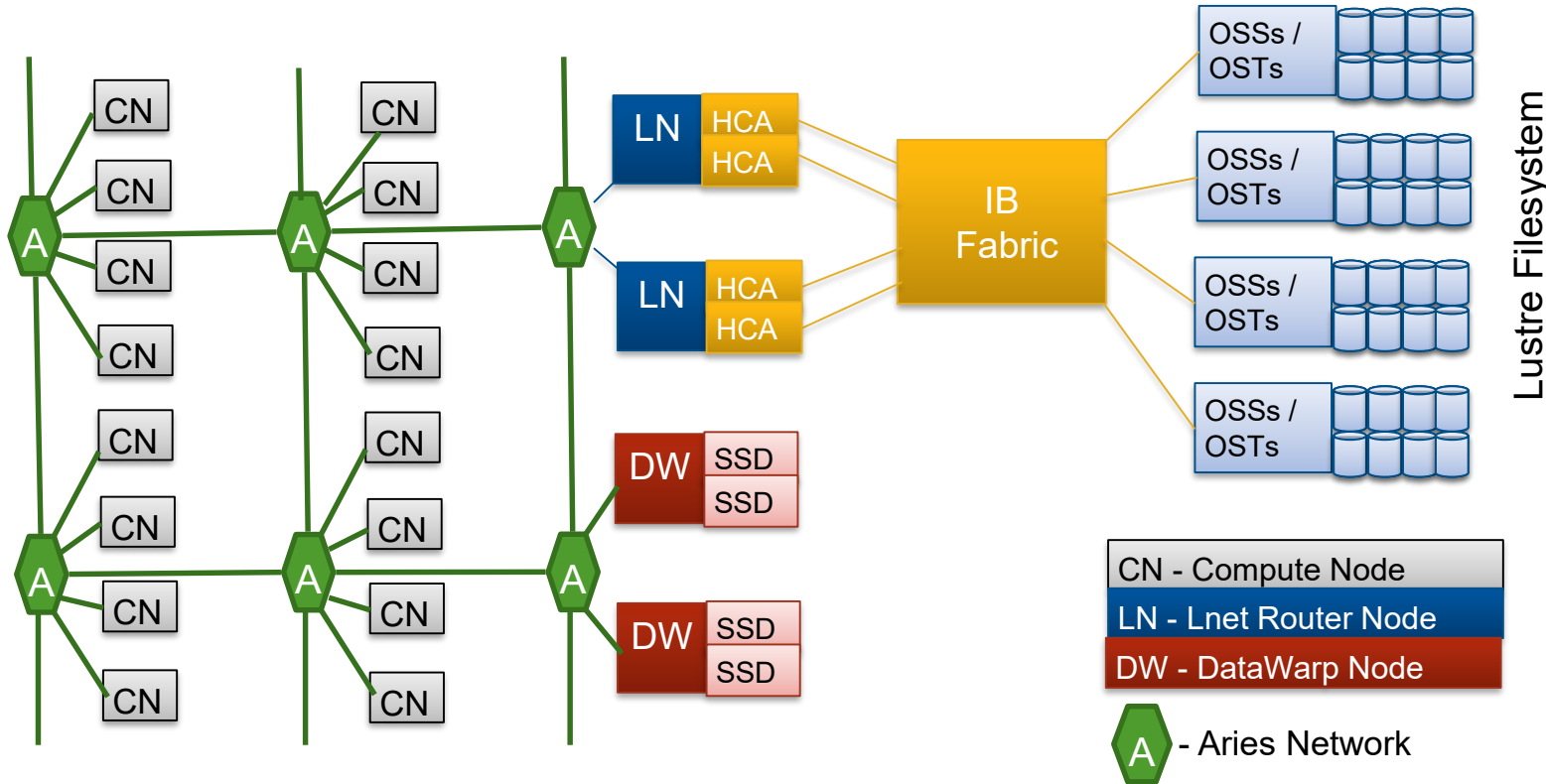
- Burst sequentially to a bandwidth optimized medium; destage to capacity tier
- Handle competing workloads that would otherwise thrash spinning disk
- Handle many nodes of a single job in parallel even if not tuned for huge I/Os

# Customer Workflow Specifications

## ● Implied Requirements

- *“launch an application at full system scale in less than 30 seconds...describe factors (such as executable size) that could potentially affect application launch time...describe how applications launch scales with the number of concurrent launch requests (pers second) and scale of each launch request ”*
  - **Translation: Open a bazillion files at once; open and read a single file a bazillion times concurrently**
- *“provide...consistent runtimes (i.e. wall clock time) that do not vary more than 3% from run to run in dedicated mode and 5% in production mode”*
  - **Translation: QoS controls on fabric, guaranteed I/O rates regardless of I/O pattern or size**

# DataWarp Summary



COMPUTE

STORE

ANALYZE

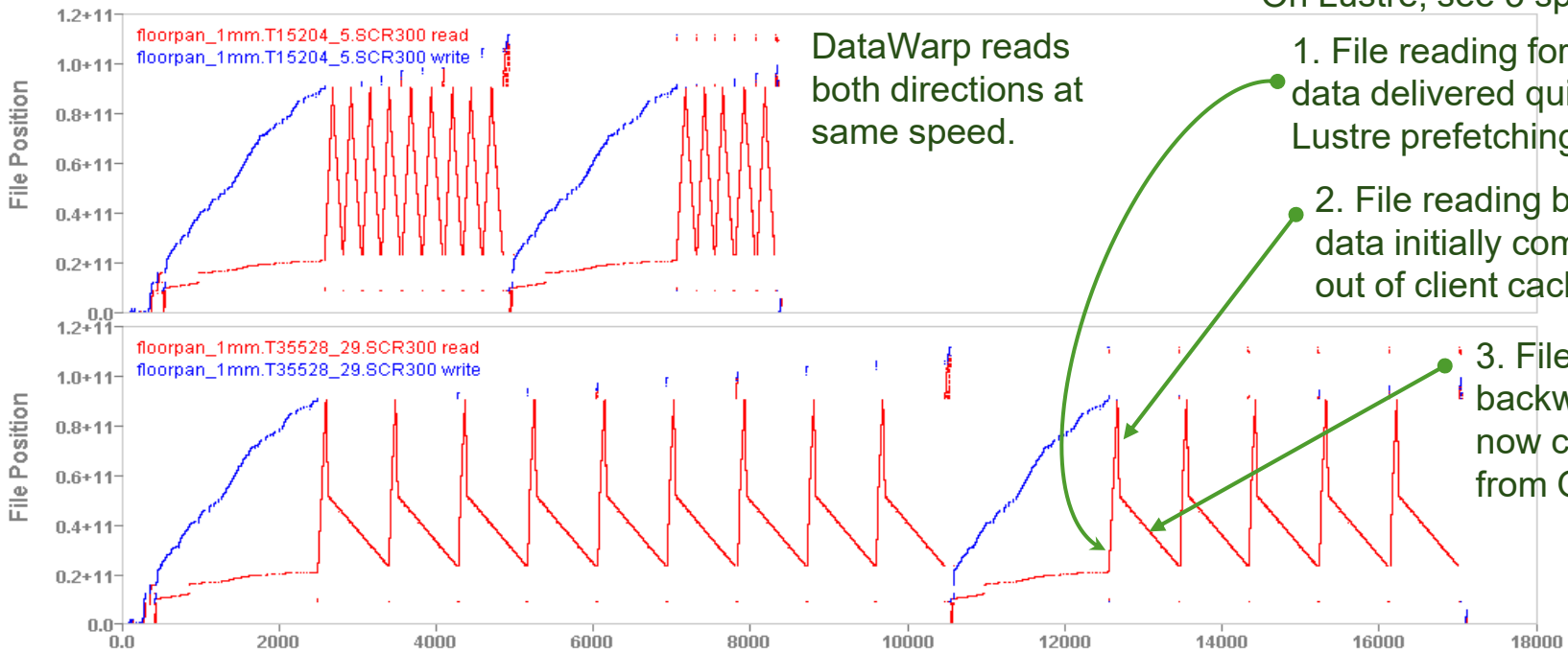
# Nastran Example – Forward/Backward Reads



File position (left) vs Time (bottom)

On DataWarp

On Lustre



On Lustre, see 3 speeds:

DataWarp reads both directions at same speed.

1. File reading forwards, data delivered quickly using Lustre prefetching

2. File reading backwards, data initially comes quickly out of client cache

3. File still reading backwards, data now comes slowly from OSTs

Lustre job takes twice as long.

*I/O activity in the SCR300 file, showing the forward and backward passes of reading the factored matrix.*

COMPUTE

Wall Time

STORE

ANALYZE

# Frequently Unanswered Questions

- **Project-Related**

- New or existing project?
- What is the current workflow?
- What are the drivers of change?
- What must remain the same?

- **Volume Variety Velocity Veracity**

- The “Guzintas and the Guzoutas”
- Where does data originate? Internally? Externally?
- At what point does it come into your control?
- With what frequency, format, data quantity, object quantity?
- When is data altered, reduced, multiplied?



# Frequently Unanswered Questions

- **Consumers**

- What applications and users access the data over its lifespan?
- What are the app interfaces' requirements?
- What is the concurrency and granularity of access?
- Profile moments when data altered, reduced, scaled, duplicated
- Does consumption and transformation yield a new source?

- **Data Husbandry**

- What are the security, provenance, fixity, validation requirements?
- How long must the data be retained? Are there legal holds?
- How is data expunged? Are there new / emergent requirements?