# Valmar: High-Bandwidth Real-Time Streaming Data Management

**David Bigelow**     **Scott Brandt**
Systems Research Lab, University of California, Santa Cruz
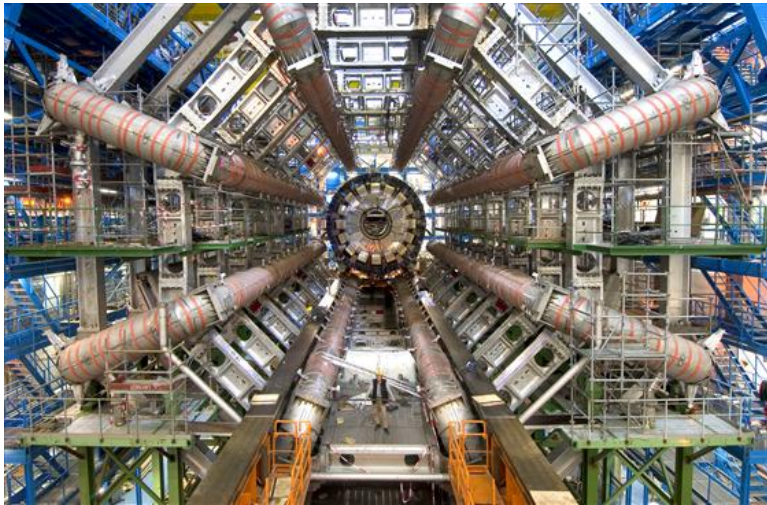
**John Bent**
EMC

**HB Chen**
Los Alamos National Laboratory

**MSST 2012**

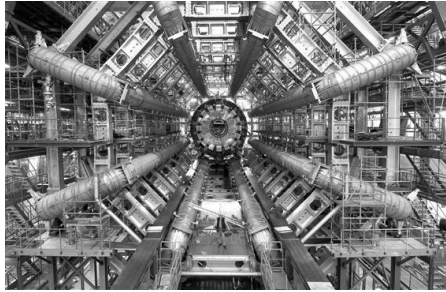# Ever-Increasing Data Rates



ATLAS Experiment, CERN
**300 MB/s**
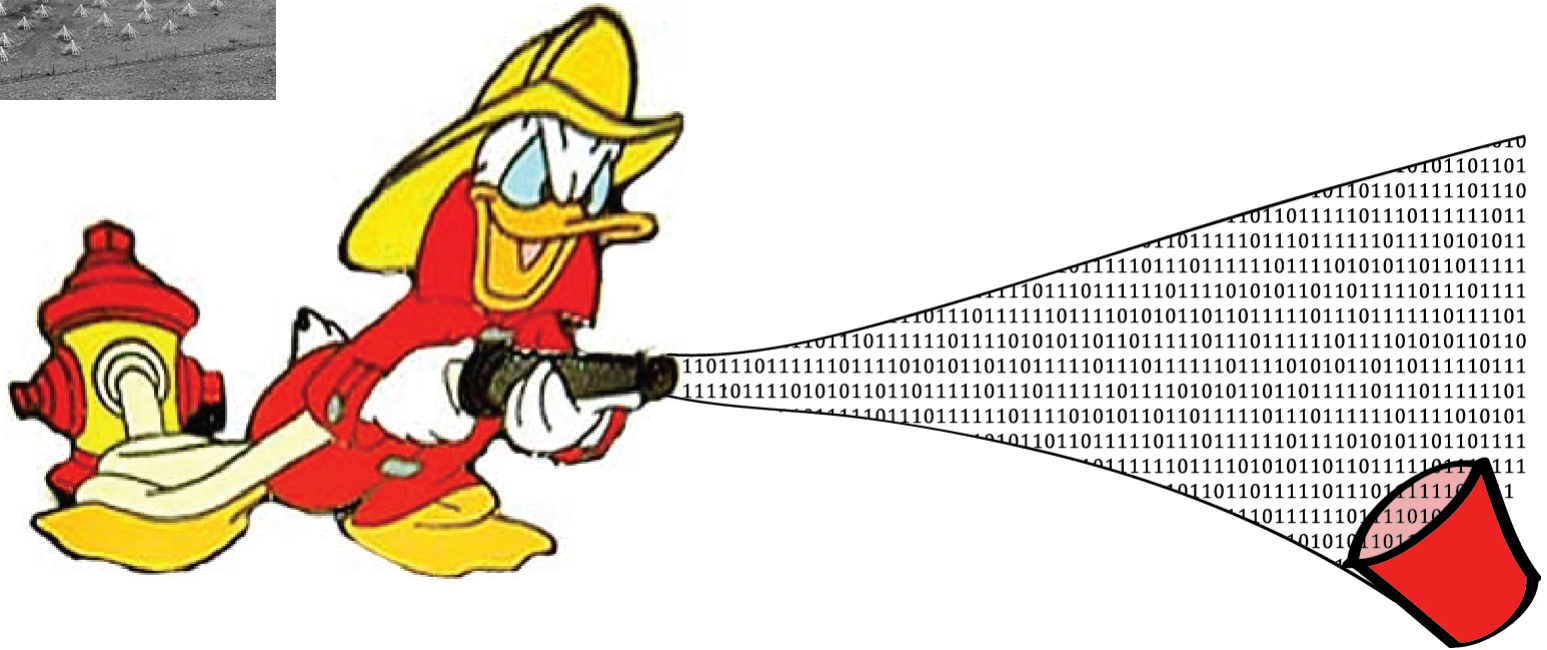


LWA Radio Telescope
**3.75 GB/s**



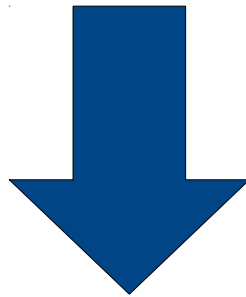Cisco CRS-1 Core Router
**Toward TB/s**

# A Firehose of Data

**How do we store it?**

**How do we search it?**

**How do we manage it?**

# Problem Area

- **Too much data to retain permanently**
- **"Interesting" data arrives unpredictably**
- **May take time to identify "interesting" data**
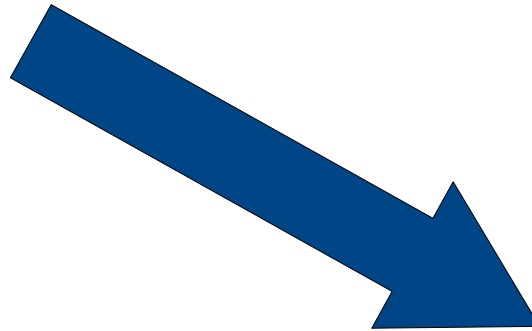- **Arrives in real time at high speed**

Thus,

We must fully capture, but only temporarily store, all incoming data.
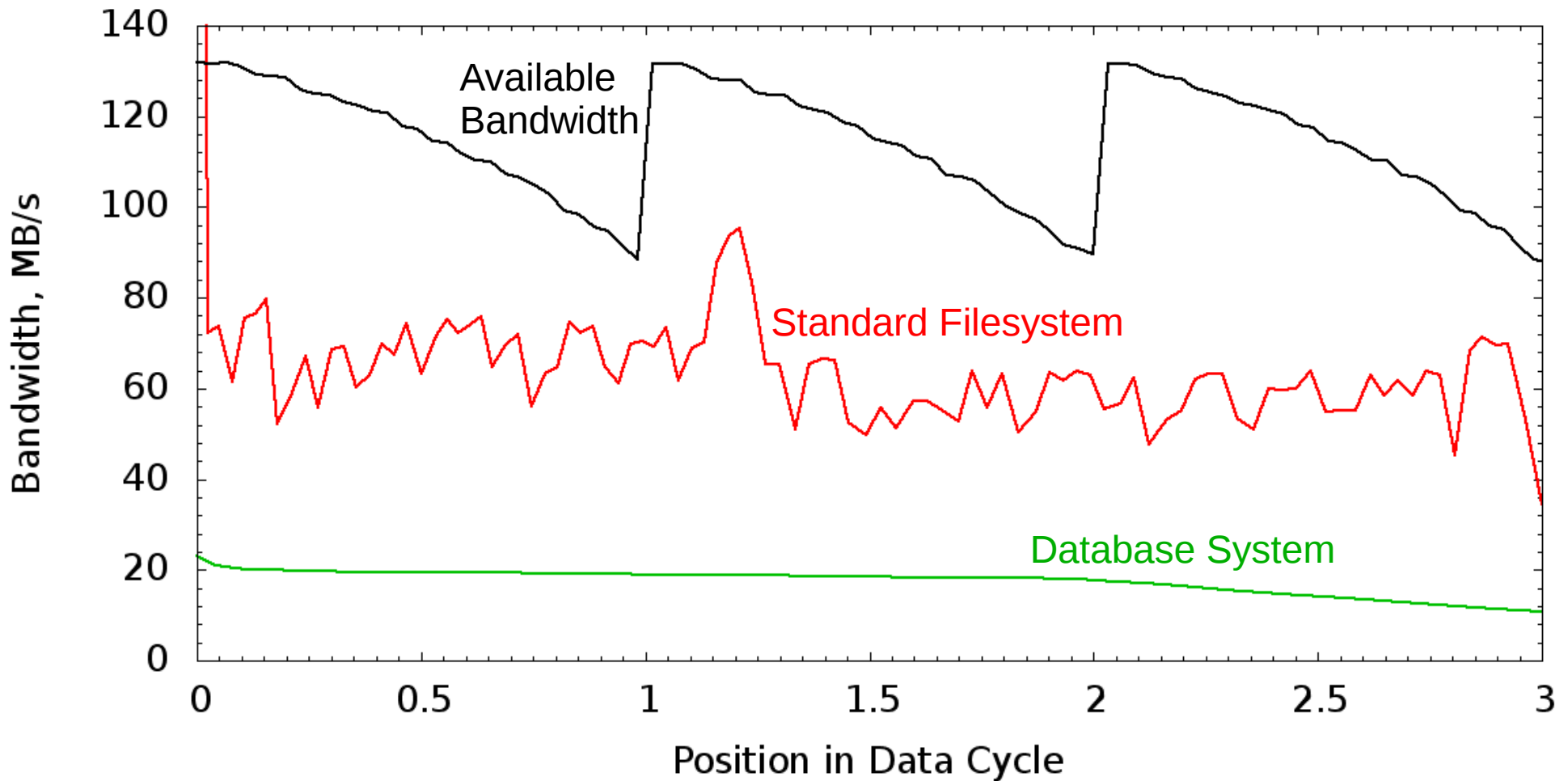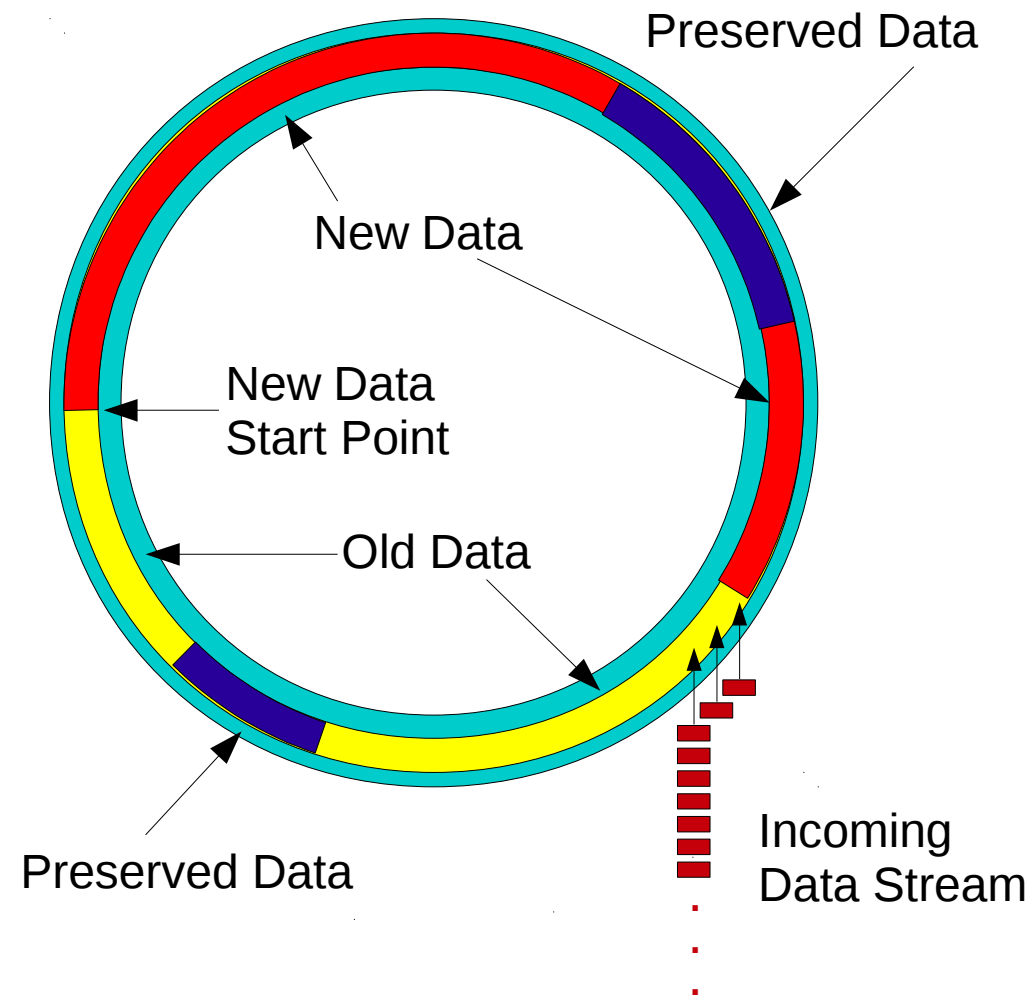
# TiVo on Big Data



Attach THIS...

...to THIS

# Existing Systems - Single Disk



Existing systems offer neither high performance nor reliable bandwidth

# Ring Buffer Model

- **Low data lifetime**
- **In-place preservation**
- **Data overwritten as first-in, first-out**
- **Incoming data gets priority**

Preserved Data

New Data

New Data
Start Point

Old Data

Preserved Data

Incoming
Data Stream

# Data Organization: Chunking

- **Data chunks consist of many data elements**
- **Size: in the tens of MB**
- **All I/O operations performed on chunks**
  - Hardware delay minimized (seeks, rotational delay)
  - Static disk layout
- **Disk performance becomes predictable**
  - Can make tight bandwidth guarantees
- **Overall bandwidth is higher**

# Effects of Chunking on I/O Times



Required time, 10 GB of Distributed I/O
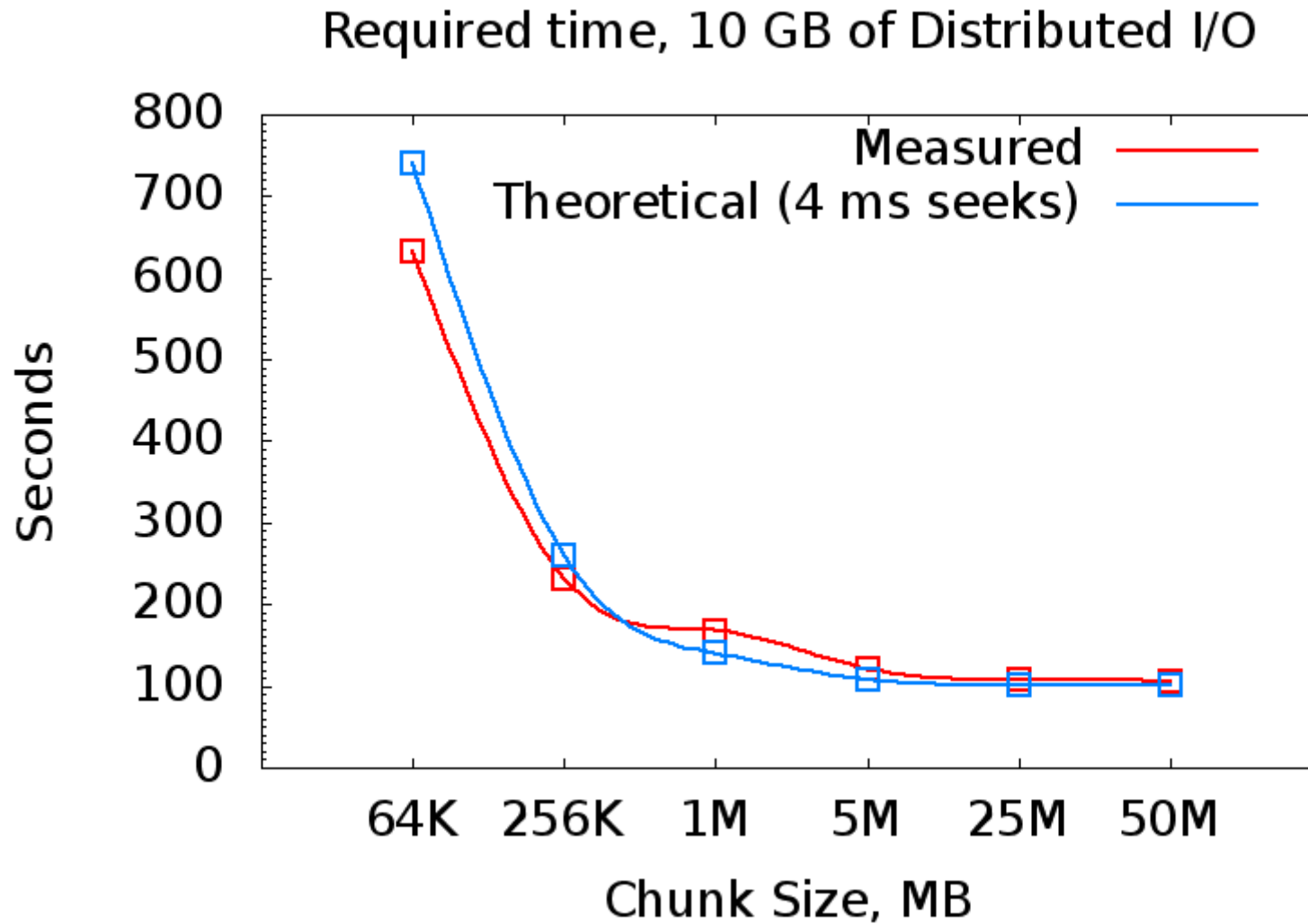
# Indexing

- **Same requirements and limitations as data**
  - Created and stored in real time
  - Expires along with the data
  - Usually not needed
- **Can have complex requirements**
  - Many indexing vectors
  - Index size can approach that of the data
- **Fundamentally a data categorization and movement problem**
  - Actual searching is fast once index is moved from disk to memory

# Indexing Example: IP Packets
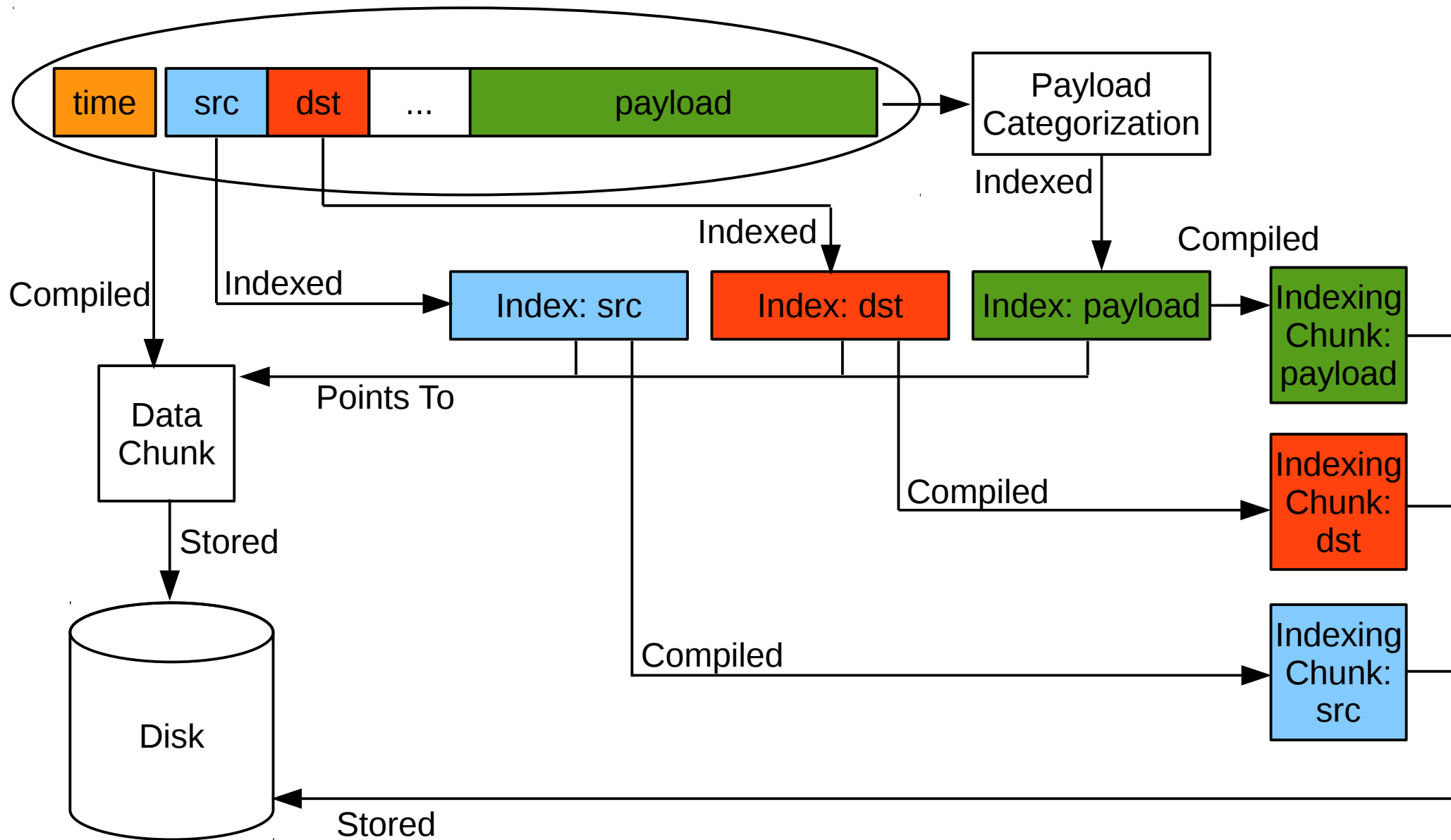
# Evaluation

- **Two comparison systems**
  - Flat filesystem (ext2)
  - Database (mysql)
- **Multiple systems and hard drives**
  - Results here are from one system/drive for comparison purposes (~80 MB/s sustainable write bandwidth)
- **Tests tweaked to suit base system ability**
  - Synchronizing constraints looser on comparison systems
  - Data per cycle was about 80% of the drive's maximum capacity

# Large Elements, Comparison
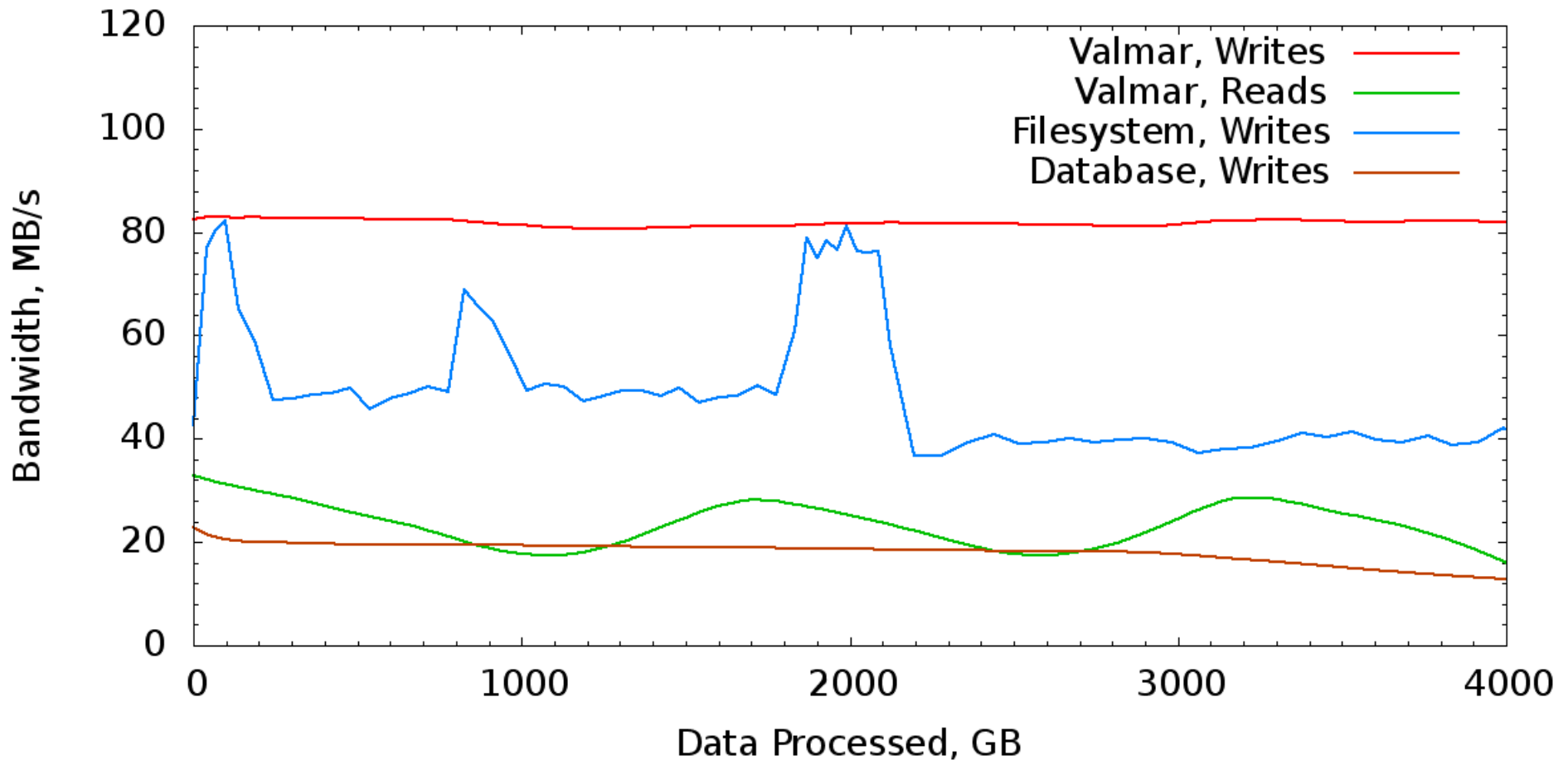
Requested Write Bandwidth: 80 MB/s
Requested Read Bandwidth: Best Possible
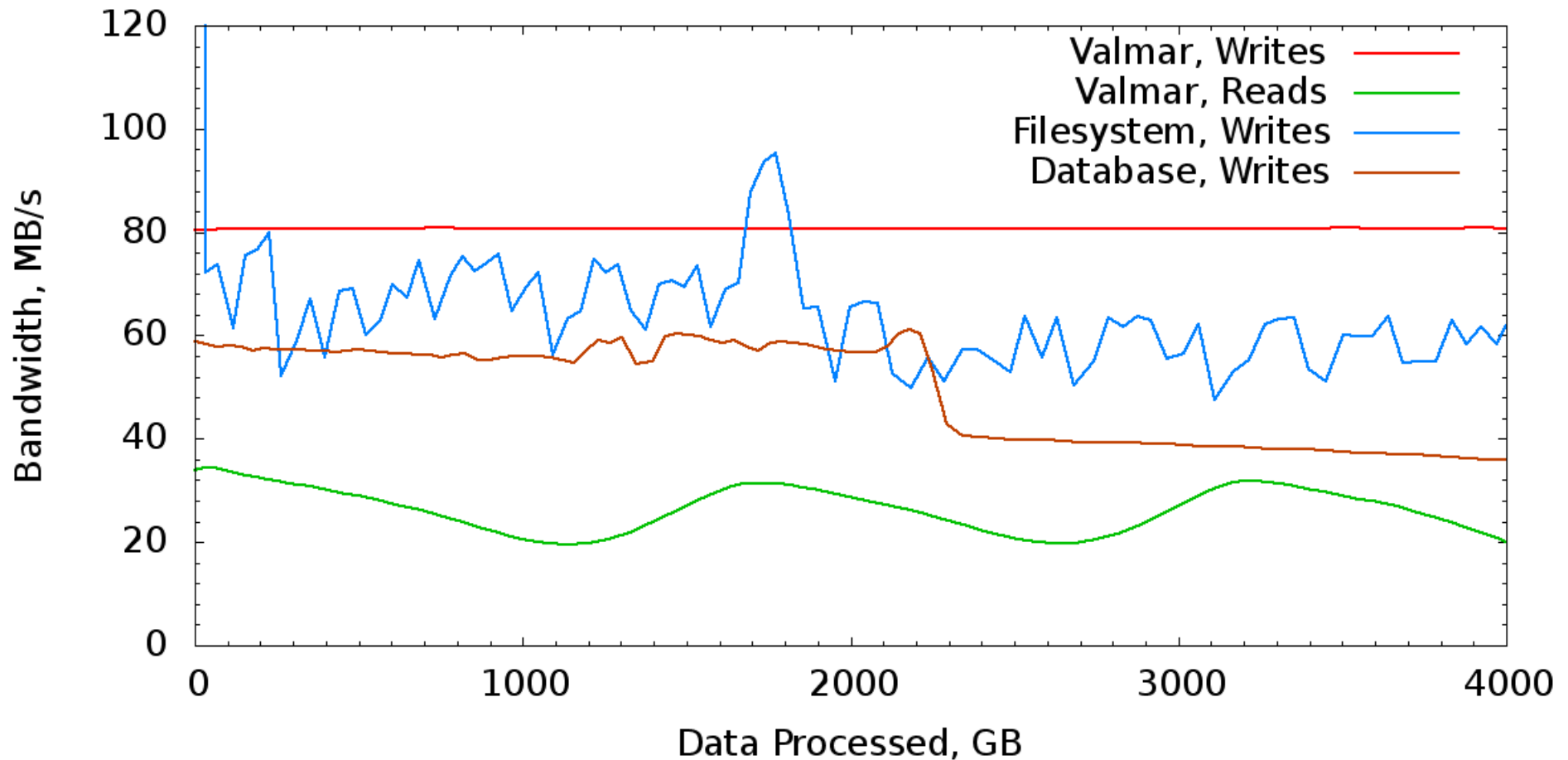
# Small Elements, Comparison

Requested Write Bandwidth: 80 MB/s
Requested Read Bandwidth: Best Possible

# Questions

- **Questions?**