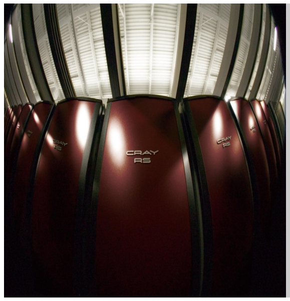


Sandia
National
Laboratories

Exceptional service in the national interest

Delta: Data Reduction for Integrated Application Workflows and Data Storage

**Jay Lofstead, Greg Jean-Baptiste,
Ron Oldfield**

**Scalable System Software
Sandia National Laboratories
Albuquerque, NM, USA
gflofst@sandia.gov**

HPC-IODC

June 23, 2016

SAND2014-17090 C

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

The Problem

- Simulation output continues to grow—desired simulation output far larger.

- IO bandwidth to parallel file system not sufficient
  - Burst buffer helps, but can still overwhelm capacity because of cost keeping capacity down

- Lossy techniques may not be appropriate

- Lossless can be computationally expensive

- Passive system level checkpointing is too coarse

Motivation

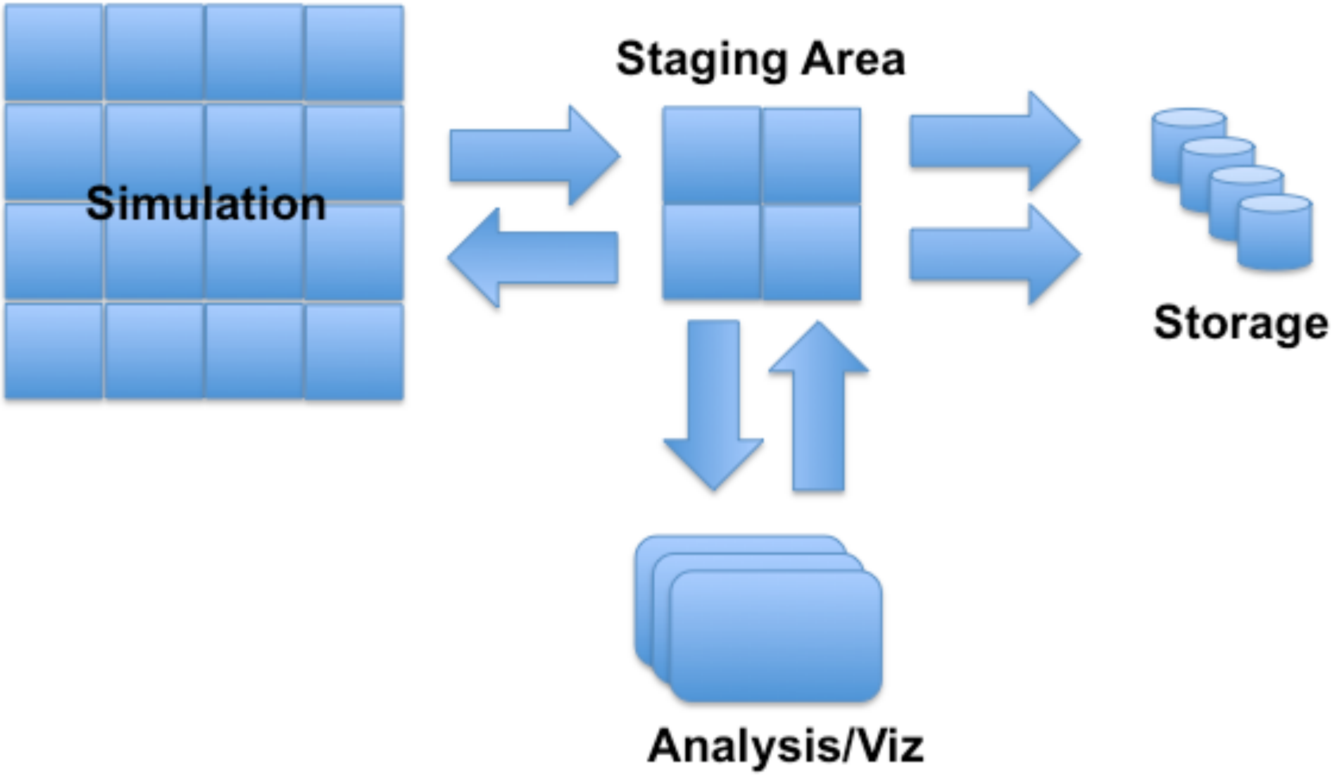
- LAMMPS tested
  - “crack” example roughly 40-60% of data changed per output (leaving 40-60% UNCHANGED)
  - “melt” example roughly 60-75% of data changed per output (leaving 25-40% UNCHANGED)

- Deal.II tested (Finite Element library)
  - Similar potential savings to “crack” example

25-40%, minimum (60% maximum), data unchanged per output step. Can we use de-duplication techniques to reduce this?

Target architecture

- Use in compute area storage to stage data



The Challenges

1. Reduce data volume losslessly

2. Keep computation overhead as low as possible
   - O(n)

3. Work across different numerical methods

4. Keep space overhead reasonable/small
   - O(n)

Major Related Work

- Network-level compression not appropriate
  - Extra software required
  - Potentially extensive extra storage space

- ConCORD works on VMs and does not focus on moving data off node

- AI-Ckpt works at the page level leaving opportunity on the table—particularly if huge pages are used

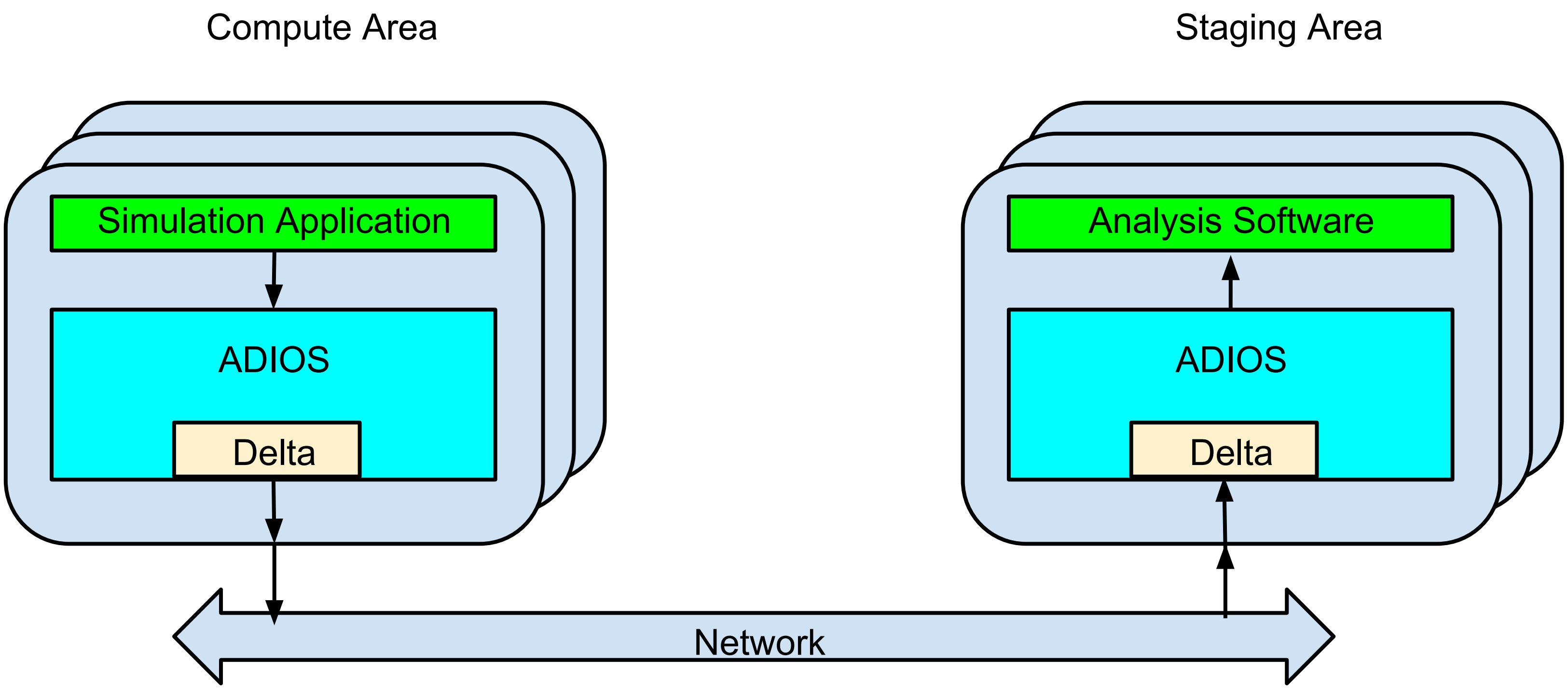
- Isabella does compression, but must sort first

Solution Approach

- KISS in action! :-)
- Simple diff between vars
- Use bitmap of full array size to indicate changed (included) elements
- Maintain the last output to diff against for next
- Performance O(n) (linear scan of data buffer)
- Data Overhead O(n) (save last output)

Target Design

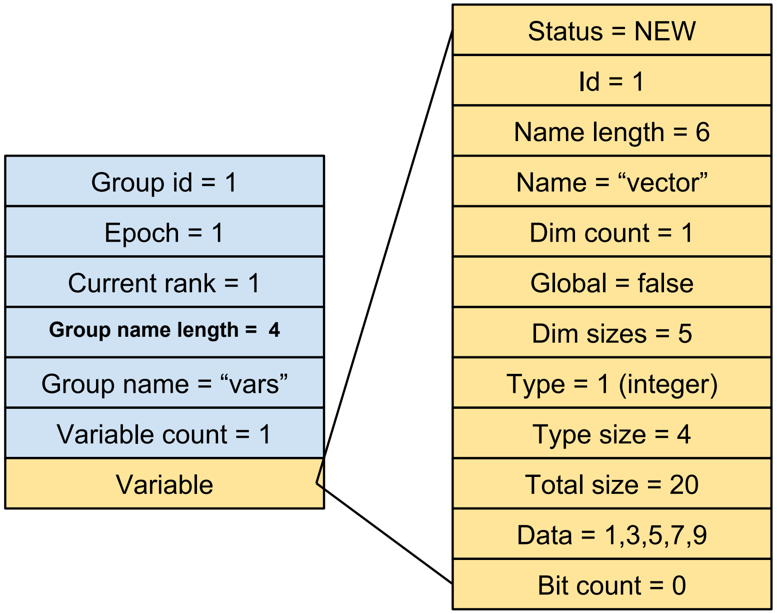
- Trivial changes for clients
- Invisibly operates
- Potential to keep reduced data longer



BP Format Adaptation

- Largely ADIOS/BP still, but some changes to encode differences

Full Data Set



Group id = 1

Epoch = 1

Current rank = 1

Group name length = 4

Group name = “vars”

Variable count = 1

Variable

Status = NEW

Id = 1

Name length = 6

Name = “vector”

Dim count = 1

Global = false

Dim sizes = 5

Type = 1 (integer)

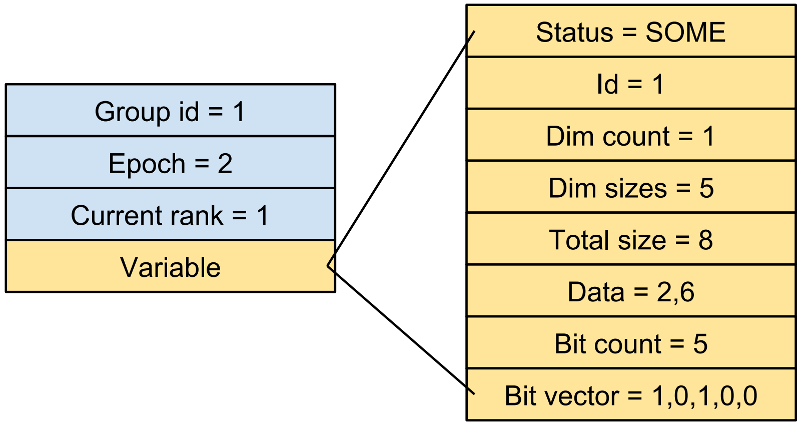
Type size = 4

Total size = 20

Data = 1,3,5,7,9

Bit count = 0

Reduced Data Set



Group id = 1

Epoch = 2

Current rank = 1

Variable

Status = SOME

Id = 1

Dim count = 1

Dim sizes = 5

Total size = 8

Data = 2,6

Bit count = 5

Bit vector = 1,0,1,0,0

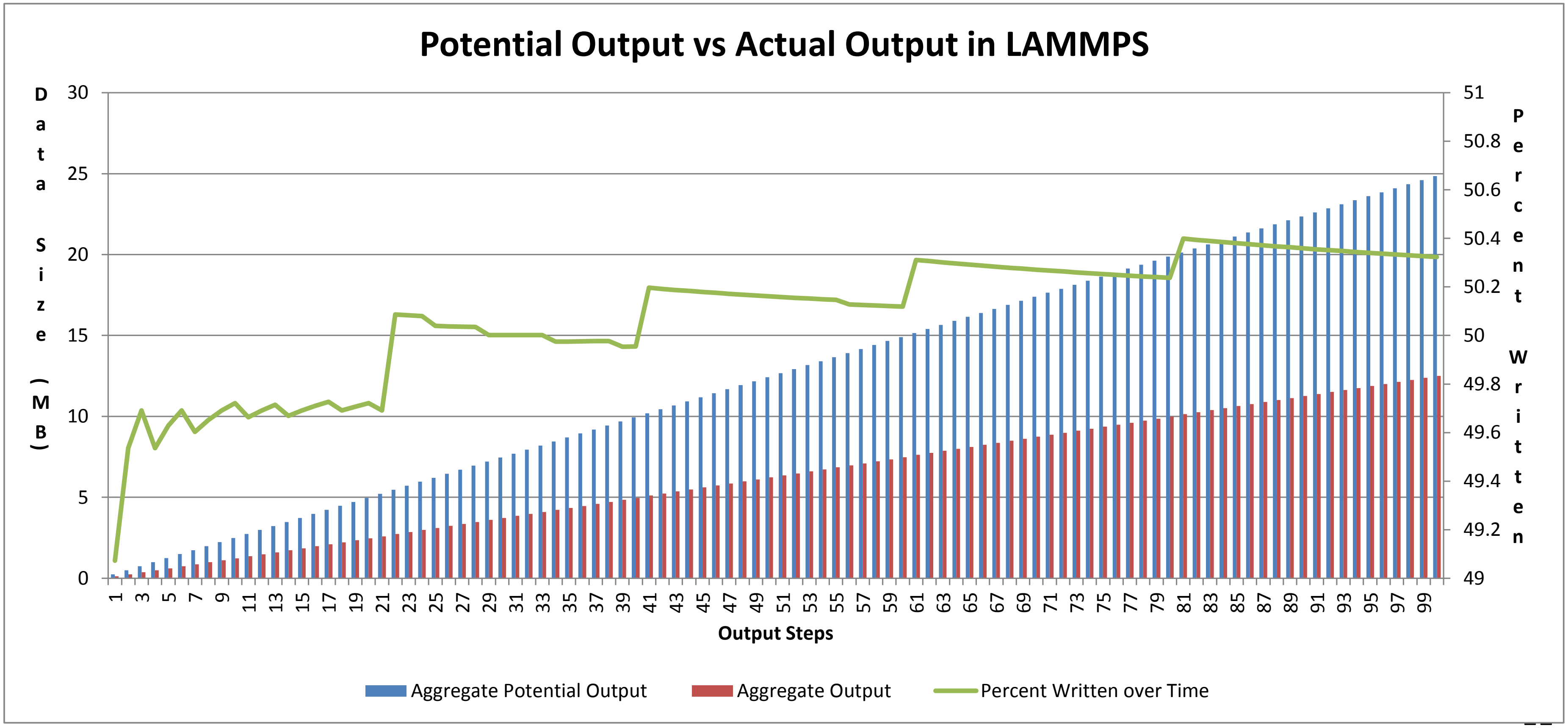
Evaluation Setup

- Chama machine at Sandia
  - 1232 nodes; 16 cores, 32 GB/node; 4x QDR InfiniBand; RHEL 6

- Molecular Dynamics (LAMMPS)
  - “crack” detection simulation

Data Volume Differences

- Blue is potential total output size, red is actual
- Green shows percentage of full output



Conclusion and Futures

- Simple, low complexity, low overhead approach can be very effective for multiple numerical methods, but not all
  - PIC codes are probably poor candidates, for example
- Applications just need to use the "delta" transport to gain advantage and no visible impact
  - Example does an expansion at the receiver automatically for test version
- Could expand to storage as a new BP format version saving space and time
- Currently only arrays considered—scalars are always sent
- Diffs represented in evaluation are against initial data set—savings could be much better if the last output always saved instead

Questions?