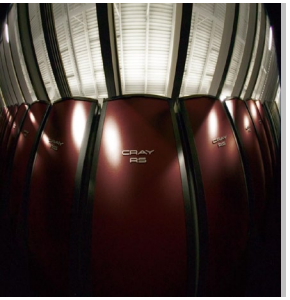

Data Pallets For Traceable Data

Jay Lofstead, Joshua Baker, Andrew Younge

PDSW-DISCS WIP

November 12, 2018

SAND2018-12555 C

Sandia National Laboratories

Exceptional service in the national interest

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

Containers circa early summer 2015

Sandia National Laboratories

- My initial contact. Key things noticed:
  - Portable
  - Multiple containers loaded to run an application to encompass and share libraries
  - Isolation
  - Encapsulation (File system in a file)
  - Unique hash code for each container

Which of these are the most important?

Containers circa early summer 2015

Sandia National Laboratories

- My initial contact. Key things noticed:
  - Portable
  - Multiple containers loaded to run an application to encompass and share libraries
  - Isolation
  - **Encapsulation (File system in a file)**
  - **Unique hash code for each container**

These two, when creatively used for storage, can link *ANYTHING* back to the creation context.

The challenge for 2.5 years: get funding to work on this. :-(

FFWD August 2018: Funding!

Sandia National Laboratories

- And an intern (Joshua Baker)!
- And Singularity is gaining traction
  - Key features: security and writeable container (if created before run)

Proof of Concept goals:

1. Zero application code changes
2. Automatic annotation with hash codes for context
3. Demonstrate in a workflow engine (Sandia Analysis Workbench)

Procedure

Sandia National Laboratories

1. Application changed, if necessary, to create a new directory for each output (0-2 LOC maximum needed)
2. Containerize the application
3. Containerize the input deck
4. Run the application specifying the input deck container as something to mount
5. Container system intercepts (using FUSE or similar) ‘mkdir’
    1. Create a new container for that name
    2. Annotate it with the hash ids for the running context
    3. Mount it at the new directory name
6. Repeat step 5 for each output
7. Profit! (i.e., whenever you want to know how data was created, check the annotations for a 100% guarantee of how)

Overheads

Sandia National Laboratories

- 700 KB for the container itself (ext3 for writeable)
- 1.1 MB for the annotation partition
  - Oddly large and one of the things we are investigating

- Runtimes 0.6 seconds (for gnuplot) total with 0.5 seconds being container load time. 0.02 seconds overhead for the container creation.

What's Left to do

Sandia National Laboratories

- More details in arXiv paper (https://arxiv.org/abs/1811.04740)

- TONS of issues to investigate related to containers and how to use them for a storage format. A few examples:
  - How to store all these containers efficiently
  - How to make them work so that they don't blow out node memory
  - N-1 files
- TONS more issues to investigate to further this as a reproducibility/traceability technique. A few examples:
  - linking with analysis outputs
  - what to do when raw data is not needed anymore
  - how to store all these containers

We are working on these any many other things I won't say :-)