Concurrent Processing/ In Situ

Group A

What does it mean to be running in situ?

- A definition for in situ processing:
  - Process the data before it is written to the disk.
- Two technical directions meet this definition:
  - Co-located on a node (techniques that exploit data locality)
  - Concurrent processing (shipping data to dedicated vis/analysis nodes, possibly reducing first)
- We probably don't know what the winning direction is yet … flexibility will be key to keeping our options open in the future.
  - It may likely be a combination of the two
- This slide defines scope.

Motivation for in situ

- Reason #1: Want to make sure the calculation is running well
- Reason #2: We won't be able to write all this data … we must do it in situ
- And new opportunities:
  - we have a chance to do something we haven't before … we'll get to see all the data and that is a positive … we can capture transient events (are there examples?)
  - If we find something interesting, we can add extra attention to that event ("steering the analysis")
  - Steering the simulation (?)
    - C.S. Chang: provenance must be part of this process if we go that direction.
  - Debugging and performance optimization

Current state of the art

- In situ scaling results up to 15K cores
- Tightly coupled in situ analysis and visualization
- Efforts for “write once, use many” in production setting (VisIt); problems remain in terms of minimizing memory footprint.
- Some experience understanding what problems can be solved in situ and what can’t.
- Some efforts in identifying interesting data in situ.
- Feature detection (e.g. finding a hurricane in a climate code)
  - Finding a point in the data where something interesting happens
- Embedded as part of I/O pipeline (i.e. advanced concurrent processing)
- Many of existing success stories involve embarrassing parallel / easy parallel techniques with a few exceptions.

Issues With Current Approaches

- Some analysis requires looking at large time windows … how will we do this?
- You may not know what is interesting up front.
- New paradigms for skipping back in time.
- How to obtain interactive exploration in this setting?
- Some vis & analysis routines are fundamentally memory heavy (e.g. derived quantities) and some are fundamentally compute heavy. How does these fit?
- Not just solving one problem; we will be doing multiple analyses concurrently.
  - Further, aimed at single investigator approaches … how do we allow large groups of people to analyze the data in situ? (e.g. climate, policy decisions)

Identify needs and consider tasks, linkages, metrics for success, …

- Needs ("what do we need to tell the HW architects & system SW folks we need"):
  - Concurrent model: need fast data access to dedicated DAV resources
  - Memory bandwidth is a key issue in some processing models (e.g. the concurrent model where a dedicated node is processing a large amount of data)
  - System SW: co-scheduling of "DAV services" with the simulation based on data locality will be important
  - Programming model: effective model for dealing with memory hierarchy
- Tasks:
  - ?

Identify needs and consider tasks, linkages, metrics for success, …

- Linkages:
  - Networking
  - Programming model
  - System SW
  - I/O & Disk?
- Metrics for success:
  - Benchmarks for in situ DAV:
    - How much data reduction can we do?
    - How much power can save?
  - How much can we accelerate the rate of discovery?
  - Adoption of our techniques by simulation community
  - How generalizable are our solutions?

Identify places where need to do co-design and cross-cutting issues

- Cross-cutting: strong relationships between simulation scientists and DAV experts. We will lose this data, so we need to get it right as we are doing in situ processing
- Networking
- Programming model
- System SW
- I/O & Disk?
- Uncertainty quantification and analysis: applied math

Identify partnerships

- Folks facing exascale:
  - Exascale co-design centers
  - NNSA ASC, DoD DARPA UHPC, NSF PetaApps, NASA, international partners (e.g. PRACE), …?
  - ❖Simulation codes will also have programming model challenges
- Existing communities have had to make hard data reduction decisions (e.g. high energy physics, astronomy, remote sensing, etc.) … what can we learn from them?

Highlights and takeaway

- In situ is a way to more intelligently figure out which data to get off the machine … responsive to the exascale game changer.
- Simulations will need to accommodate hard analysis issues … paradigm shift to analysis-driven science / discovery-driven science?
- There will be a transition for what scientists can and cannot do in terms of analysis; how we will help/enable this transition?
  - Difficult upcoming conversations with communities that are leery of in situ. (e.g. climate)
- There are _many_ remaining research challenges with in situ.