Scott Intro

Monday, October 19, 2015 9:08 AM

Exascale computing initiative

- 90 million / year for software
- Paul Messina from Argonne
- More operational rather than just research (big D small R)
- What prototypes can be developed in the first 2 years
- Piece of named software

Scidac world is about to implode
All computer science will be in one scidac
Data co-design will be coming up
ECI also is connected

First few years of making something real and being able to deploy it

Jay: 15 minute presentation on how the money is going to be distributed
Buckets of money to projects through an assignment process
Big unknown at the moment according to Scott

Scott: Apps and facilities are different money from software
He wants the Sandia + ORNL collaboration to produce some realistic software
10 labs may or may not be going for all the money
Jay doesn't think every lab can go for the pot because they will have capabilities

Software has to be facing to the apps
Doug Kothe leads the app

Ripped the program manager budgets
Some PM have 0 budget

Sirius stuff now

One call / month for the project

Everyone has to be delegated
University and lab

University funding doesn't come till december

Bi-weekly phone calls for the executive committee

More frequent calls for small demo groups

Demos as the software artifacts
- Not production quality
- Focus on papers
- Web page

Git repo central

Deliverables will have to stay cogent
1st year deliverables are in Scott's slides
Cross-cut areas and clarifications still needed

Memory/storage management/arrangement

For LCFs particle in cell codes are gobbling up the most computing time because they scale
Most of them are dealing with 5 or 6-D phase space
All physics can be described in 6D + time
PIC is popular

PICs can generate a lot of data
Like CS generating 10 TB/week

Major focus on this proposal is not checkpoint restart
Its something that can be dealt with
Its clear to everyone here
Not a major focus
Auditor based C/R

Scott gives explantion of Auditor C/R

Take a code
You generally have timesteps that you are taking
Occasionally you (with some frequency) will checkpoint restart
Global checkpoint restarts
C/R can be more local
If you had a secondary calculation that can approximate your answer (a local version of the compute)
Check the error threshold to checkpoint
Compute the checkpoint frequency

More details might come from Mark

We are going to focus one of the two examples in the PIC codes
Different approaches to doing this

Goals:

1. New set of apis
    a. Lifetime
    b. Utility
2. Usable big data
    a. More formal consturction of data management
    b. Prediction based (through models?)
3. Keep co-design like design process with the application users
4. Major revolution in the I/O system for HPC
    a. Most detached thing from the computation
    b. Even compilers don't optimize after I/O

Focus on the research questions

1. What abstractions can be used to describe data utility
2. How can they be executed in the ssio layer
3. Expedite time to solution
4. Accuracy performance tradeoffs
5. Storage size management

Would AMR take the wiggle room we have in the data away?
Scott doesn't think it matters

Scott doesn't think it matters

Take the data from an examplar (I thnk XGC)
Plot the data and it's a bunch of random squiggles
But if you sort it it makes a smoothish distribution

George: Non-parametric estimator of the density function is the graph

For most analysis you take the derivatives
If you reduce the number of significant digits (3 digits) and take the first derivatives
    Errors are pretty small
    1e-3
    Second derivatives same
    Having less digits can be precise enough

Manish: How much of this is application specific
Or how much can we make this application agnostic