Breakout session

Tuesday, December 06, 2011

11:26 AM

Prioritize

Names of everyone in the room:

* 1. Scott Klasky (ORNL)
  2. Hasan Abbasi (ORNL)
  3. James Osborn (ANL)
  4. Steve Gottlieb (Indiana)
  5. Richard Bower (BU)
  6. David Bader (LLNL)
  7. Ian Foster (ANL)
  8. Just observing ()
  9. Jackie Chen (SNL)
  10. Anshu Dubey (UC)
  11. Robert Whitten (ORNL)
  12. Bronson Messer (ORNL)
  13. Bill Tang (PPPL)

What do you want from collaboration when you do simulation science? What are the major points

* + Automated metadata capture

Bill: When DOE finds the money it wil lhappen

What are the impact of new technologies. What advances in technologies will enable new sciences

Scott wants to list all the pain points

David Bader: We are starting at the bottom of the learning curve and if we just do vision/problems we won't move up the learning curve

Integration beyond anyhting we have done because the technologies are new. Advance the science by redifining the paradigm

We need to keep up with the technology curve so that when we develop new techniques they don't make our existing efforts stale

Bill Tang: We have to show impact so that we can convince the directors to support us and we have to sell this to experimentalists. What are some of the low hanging pieces of fruits that we can say "this si making a difference so if you invest in this, it will have this sort of impact. This is stuff we couldn't do 5 years ago but now we can due to these new developments".

Need real crisp statements of what we are going to go after.

Its not true that large organization don't know how to collaborate but that the tools are too specific to domains and they don't generalize. We need to look at existing successes and see how they did it in (say) 10 years and replicate the effort to new domains in 2 years (or less). Eg. Streamlinign information/code sharing in combustion (Jackie)

Scott: What are the gaps? ASCR wants a research program so we need to know whats missing. ASCR wants our input (we are KINGS!)

Jackie: We aren't going to figure out how to do this (solve the problem ) in a day. In combustion area it took a year of a multi-lab effort. This is a multi-faceted problem from tech to social culture and each community will deal with collaboration differently. We can't distill this into a report in the course of a break out session.

Scott: Can you summarize some of the things that you saw in your collaboration efforts. What were the issues? If we can understand what are the issues we can look at what are the gaps and when we get a CFP we can see what people should go after? Both low hanging fruits and what are the fundamental problems.

Richard: If the main barriers are sociological (or so it seems) and that obfuscates what the technological barrier. So we should look what the ideal (not socially constrained) world what are the tech problems?

There is a controversy over if the impediments are actually sociological.

Anecdote: Sociology is that your data never sees the light of the day unless it is leading to a publishable work. Can we encourage the sharing of non-publishable data.

Perhaps the technology might bring out the socilogy with it.

It goes hand in hand - we have the solution for you, you are stupid if you cant' use it /s

Why are people afraid of collaboration? They are feareful of not getting credit. Generic problem of reward mechanisms.

Motivations are different for different size of teams (slide 26 I think).

Lets take each of these exampels at the different sizes and see if we can find a common thread.

Anshu: Different prespective. FLASH had an original purpose but people started to use it for different things (different domains) and they all want to contribute to the original. Core team is trying to get all this coordinated but version control is a crude way of dealing with this.

Bronson: We roll our own tools because we want to have users/developers instead of users and developers. That works for some tools but won't work for FLASH.

Scott: We can't solve all the problems (not 100% solution). We can't roll our own either. Bronson says that it was a necessitiy an we want to set up a research program that solves the issue.

We need something that cuts across the domains, some low level middleware. We need to look at if we can find a commonality between say combustion/fusion.

Find where we have overlap and what are the differences.

Scott: What are the pains and workflows.

Revision tracking/change tracking in pdfs for example.

Ian: We should look at organizing studies that actually look at what the work processes it. More than anecdotal examples. If we have a 10 year process we should look at the fundamental issues. For example: Google must have some sophesticated processes.

Computing is now product is a paper. Basic research drives paper production but not product production.

Successful companies have basic research and rapid R&D step that creates new products.

Look at commonalities of workflows between fusion/combustion climate science.

Collaboration involves a workflow between collaborators - beyond just running a simulation.

How can we find gaps if we don't identify the targets?

Where are the gaps in the tools? WE need to come up with tools that have a big impact in this area.

The biggest gap is that almost everything we need exists but doesn't exist in an integrated way that we can use. Sort of like a lab of equipment but unconnected.

Our roll your own products are the glue that creates the integration .

Level of effort for a small collaboration is 1 technical staff member 1 year.

Until recently people only cared about data for short periods of times, but now because simulations are so expensive that we need to keep it around, track the provenance. Data curation is now important.

Scott says glue isn't out there.

What is it that we need in a collaborative environment.

Is it glue? Is it what are the pieces we need to glue things together.

If we can come up with a report that says what are the common things.

Ian: Glue trivilizes the objects but its really end to end processes.

Many times we want to do the same thing twice. Streamlining time to solution (oh snap!) because we repeat so many tasks in a workflow.

Scott wants to create a document that describes these repetable steps.

Many workflows across communities.

Need to document it.

No one is smart enough to come up with the ultimate solution. We should take the ingridients that are out there for successful efforts and evolve the solutions. There is no top down solution. It’s a grassroots problem.

We want to make the tools more dependable, faster, robuts and easy to use.

Linux for example developed an open software approach. They made a framework that made it grassroots.

Multiphysics code that required interoperability… We need to create components that work together with each other.

Hasan: grassroots are inefficient, they make progress by being like grass.

ESGF workflow described by David (slide).

Workflow is scale independent.

Hard part is how do we switch components/analysis? ESGF has a lot of diversity of data so they have studied that in great detail.

Slide has a high level view of the science and the interaction between theory observation and simulation.

Bronson wants to cut the boxes out and have a single development collaboration.

3 levels of literay [ developers/user, user, <last one I didn't catch but at the lowest level>]

Use the workflow diagrams as the context as to whats there, whats missing and what needs to be connected.

Instead of day in the life o fa scientest you need to do day in the life of community so we don't tailor to individuals but to groups

In ESGF there is a lot of expert judgement and the codes are evolving.

What are the pain points?

* + Biggest is smart use of the provenance capture and go through the workflow to capture the provenance and then automating it for the next guy.

Provenance publishing is really hard because so many sources.

Ontology, provenance and publishing.

Fast querying? That’s also needed.

Astro has auto capture of data.

Our focus is on large simulations so why are we talking about experiments/sensors.

WE need to work with experimentalists because when we go to exascale we will have issues with data integrity.

Remote visualization is very important because the data sets are way too large and the render power required is needed.

We don't need a lot more interaction between people. (Controversial)

We can't live without remote visualization. AND we need to move to reduced data sets.

We are all about big simulations. WE need to advise ASCR how LCFs will be involved in this.

Computer science solutions are solved but they often miss the difficulties of dealing with LCFs.

With extreme scale we can actually lose access to data on short resources. Where is a DOE data center?

Sort of like TerraGrid/SDSC. NERSC is the closest thing.

We need to move work to data because we can't really move the data.

Default right now is to get every collaborator access to where the data is instead of moving data to where collaborators are.

Security needs to be addressed as a first class issue.

Federated data systems. Keep the data as close to the source as possible and don't move it if not needed. Requires tools that do search/queries/indexing. Move the data only when absolutely required.

Scott thinks there is a lot of further development required.

Ian says that we have pieces of technologies but we don't have them integrated.

Why can't we have DOE labs have a common login? Bronson says it would literally take an act of congress. (ouch)

Scott says we need to share data differently than actually running simulations.