HDF5 BOF @ SC08
November 19th, 2008
Quincey Koziol, Ruth Aydt

As people were settling in, asked about their demographics:

Academics: most
Labs: some
Commercial: a couple

US: most
Europe: a few
Asia, S America, Antarctica (none)

24 attendees counted mid-way through the presentation.

Questions during Quincey's presentation:

Question: What about ubiquity – people are hesitant to put their data in the format if they fear it will go away / not be supported.

HDF Group: We are committed to running on all major platforms but can't ever guarantee it. We are looking for ways to get funding to run and maintain everywhere. We do provide good stress test for compilers.

Audience: Issues are typically not compile time failures; usually problems with high-end performance on very large configurations; Sometimes at layers under HDF5 itself; Sometimes with applications not doing the "right thing".

Question: What makes something officially supported on a platform? What does it mean to be officially supported? *Returned to this after presentation complete. Some of response noted here, remainder later in "transcript".*

HDF Group: There are two main levels of support:

1. A major funder cares about a platform and has us do testing there; we generally call those "supported platforms" in that the library is tested and works. Includes regression testing for performance and correctness. And, historically we then make the platform binaries (or "certified source") available elsewhere.
2. For a given user, maintenance/support contracts where we build/test on their system.

Performance guarantees generally not part of "supported" because can vary widely depending on configurations. Can do one-on-one consulting to help with this.

We mentioned neither Cray XT4 nor IBM BG are supported platforms.

Audience: Concerns that one site may get HDF5 running on an unsupported platform, possibly with a small change, and another site do a slightly different change – splintering.

HDF Group: Noted that there is a difference in severity between format incompatibilities and library differences. Hdf-forum email list is one way to share information about changes made to get it working. Other online "venues" hopefully coming at some point. We would like to see vendor or agency provide some support to us so that we could make the important platforms "supported"

Audience: Issues typically not in compiling, but in performance for high-end platforms/apps.
HDF Group: Do "RFPs" have performance metrics in them?
Audience: Yes, but not always catching things – working together (across labs, for example) to get better. Typically don't explicitly require HDF, but do require codes that use HDF.

Question: Any plan for any sort of multi-writer chuck; arbitrator to handle locking issues for multi-writers. Handles from MPI Multi-writers. Can we turn on collective w/ chunks; multi-writers?

HDF Group: Yes, collective chunk I/O works in 1.8 provided the underlying MPI handles it and MPICH handles it fine.

Discussion after presentation. Mostly transcript of points, not particularly well organized.

Attendee from National Lab: HDF5 is what people should use. Some people expressed a lot of support for HDF5 during the BOF, but said that if users try it and have performance problems then they get a bad impression and tend to go elsewhere. Not clear where. User support group at Lab favorable toward HDF5.

Regarding the funding pie chart that was shown in slides, audience mentioned DOE funding slice was quite small. DOE funding used to be a larger. Now just a couple of special cases. Quincey talked about them some.

(Users) don't sign up for performance guarantees. They ask: Does it compile or not? Are simple expectations about performance being met? This is what users expect.

Ruth asked question about the number of APIs to see how attendees felt about that.
Quincey said: Strong feeling about how many APIs. Don't want to force you to do it just one way – give control and flexibility.
Most of the audience didn't seem concerned about the number of APIs. But, did say it could be confusing for new users.

Question about HL HDF5 library. What are the performance implications of using the HL library? Is it just wrapping other routine or does it do more caching, etc.

Quincey - Our HL APIs typically just wrap or handle details of frequent case (tables)

What about a parallel HL API? Several people felt a HL parallel I/O API would help reduce complexity *and*, perhaps more importantly, keep them from making "bad choices" with their applications.

What about netCDF4 as the HL API. Nice for earth science. Several NetCDF users in the audience and also some questions about it. Quincey explained that the Unidata netCDF team wanted to focus on the netCDF data model and not on the file format, and were officially backing netCDF-4 (on top of HDF5).

Talked about domain-level data models & APIs like HDF-EOS. This is the right way to go & HDF Group can work with people to help define the right HDF representation and also the community level representation.

Attendee with lots of HDF5 experience: Communities should be encouraged. "HDF5 is an API for developing domain-specific API. Encouragement for people to develop their own data model on top of HDF5; well-received by the audience.

One issue with current APIs --
Scientists think that every operation is equal cost. Not true. Corral the user or give feedback / guidance.

Is there a parallel performance VFD to give feedback there?
Quincey, NO, but it would be good.
User was looking for something like the "log" VFD applied to the MPI-IO/MPI-POSIX VFDs>

Performance regression testing. Users either really do get a lot worse performance –or- perceive they are getting worse performance. How to distinguish the 2 and if the first, why? Users run away when performance is bad.

Best Practice would be useful.

Plethora of interface options makes it difficult to figure out

Quincey said we need a recipe book or something like that, and audience responded favorably.

Attendees: Pretty good manuals – can usually find the information needed in online manuals.

Defect tracker – could it be public? <concerns about keeping some client info confidential were raised by audience and HDF Group concurred>

Helping new users get their feet wet

What are the performance implications of a particular choice? Issues are about performance at scale (1000+ cores).

(this goes back to more about performance on very large system – continuation of discussion that was started in Q/A above.)

Vendor's MPI does not work well… Vendor should be forced to address this. <from audience>

IOR was used to set acceptance standards. Makes it easy to choose power of 2, but sometimes that performs well while other things do not.

What are relevant benchmarks? Base 10 transfer sizes for IOR. File per processor IO and also collective IO.

One attendees biggest challenge: Reading data – how to do that fast? Doesn't know what will be in file in advance.

- More talk about scientists shouldn't be writing directly to HDF APIs.

Attendee: HDf5 is really a File format with object database model. That's want HDF5 is. Haven't trained the community on how to think on that view. ASCI DMF.

Application area: Astrophysics; visualization. Have to read the whole file to figure out what is needed.

<audience> What about samples of how to do image formats; provide templates.
Can HDF5 provide a recipe for building a community format?

Audience member suggested HDF Group host a collection of community- defined data models (HDF-EOS, Packet Tables, etc.) and have suggestions for how people should create their own.
<Many thought this was a really good idea!>

What about XML -> HDF5? HDF5 dump goes HDF5 > XML; need the other way.

XML schema to read at runtime and then leverage the HDF5 style sheets would be a big win & save lots of coding overhead.

This would play well with the community-defined data model idea – there could be a schema that validated an HDF5 file conformed to the data model, or maybe also a tool that generated skeleton code to create a file that conformed to the schema.

Seemed to be a lot of support for this in audience – way to 'bootstrap' program reading HDF5 file without having to get file layout from the file itself

HDF5 situation on bluegene. –running but not supported. Only minor tweaks needed

What about netCDF & HDF5? More discussion here.

CDAC : an atmospheric [visualization?] tool from LLNL that doesn't currently support HDF5, but suggested that it would be a good payoff if HDF Group worked with them to get HDF5 supported.

In general there were audience comments that support for plug-ins that allowed tools to read HDF5 would be good. IDL was mentioned (by Ruth) as case where this is happening.

Wrapping up

At end of session mentioned that traditionally NASA-HDF workshop in the fall; HDF Group may sponsor next year. What would you like to have? Let us know!

Also at end of session said we're still figuring out our business model and how to "market" to those who aren't the scientists. How to sell the value of the technology in terms of time / $ saved. Need to develop case studies, testimonials, etc. Have considered foundation model – at this point nothing ruled out.

Also talked about hopes for better user "interaction" forums – blogs / wikis / code repositories/ etc. Also, interest in OPeNDAP work and other ways of accessing data remotely.