

BDEC Workflow Breakout 1

Rick Stevens (ANL)
Satoshi Matsuoka (Tokyo Tech)

May 1, 2013
Charleston, SC, USA

“Taxonomy” of Workflows (1)

- Tightly-coupled (within the same machine or IDC) (vs. Loosely-coupled (a.k.a. Grid Workflow))
 - E.g., Data assimilation: weather, genome
 - Sometimes cyclic (loops), (pseudo) in-situ
 - High BW vs. low BW (also low-high latency)
 - Q: are there any loosely coupled BDEC workflows?
- Streaming data vs. non-streaming
 - SKA, LHC, Astron, EOS... – latency requirements, real time requirements
 - Cycle time – seconds vs. 24 hours

“Taxonomy” of Workflows (2)

- Automated vs. Dynamic, Human-in-the-loop
 - A.k.a. Batch vs. interactive, structured vs. non-structured ad-hoc, ...
 - Static workflow vs. Dynamic, dependent on nature of data, environment, person, ...
 - Complex, prefabricated WF vs. simple, throw-away queries, pseudo-deterministic vs. iterative, non-deterministic (in terms of convergence)
 - Differences in (over-) provisioning of resources
 - Our current Supercomputers are not really operated this way, so how do we achieve this?
 - Common solutions for BD and EC workflows?

“Taxonomy” of Workflows (3)

- Analysis Models: Rich or Shallow
- Goals of the workflow: Analytics, Exploration

Current Issues for BDEC workflow

- What are the impediments for a BDEC workflow on current machines?
 - File-based I/O (vs. persistent objects)
 - Resource scheduling? (malleability)
 - Data movement between resources
 - Data access contention (extreme parallel access)
 - Interoperability of internal data format
 - (note we have to think revolution vs. evolution)

Current Issues for BDEC workflow (2)

- In particular, why not the BD-WF concept taken off in the last 20 years (but rather superseded by say Map-Reduce?)
 - Does I/O matter? Pseudo in-situ/in-memory dynamic data structure (e.g. Data Spaces) not as convenient as it might seem?
 - Conversion of existing codes too cumbersome, and writing to a externalized canonical data format seems more convenient
 - Or is this the path forward, or we have not had a forcing background, mismatch in the requirements?
 - Can the scale of data be the forcing function (so BDEC this WILL happen?)
 - Web services? (everything becomes a JSON object? Key-value pairs) – but we need (fine-grained) parallel decomposition? Memcache a solution?
 - Being in-situ or not: matter of abstraction, or fundamental to BDEC?
 - Extreme data volume will mandate the change, traditional hierarchy still sustained or different?
 - Can this be absorbed with prog. langs, compilers, runtime, or ...?
 - What can we do in system space? Virtual memory?

Current Issues for BDEC workflow (3)

- Data models must be visible
 - How do we implemented, at scale, and efficiently
 - The Web world solution: REST API, etc.
 - It failed once with OODB... (in the '90s)
 - But practiced in multi-physics, couplers, in limited communities or settings
 - Also the PIG (map-reduce) system which is data parallel (and others...)
 - Key-value store, Region Templates

Current Issues for BDEC workflow (4)

- Do existing workflow systems solve this problem?
 - Does data layout and distribution description matter in this case? (with extremely large data and parallelism) Who does the transformation between components and how?
 - Similar to the coupler stuff in Climate... but not general enough, only worked for a small community
 - Optimize the transform (minimize frequency, optimal transformation among different components)
 - Do we transform in-situ or copy, overlay, ...?
 - For example, in genomic workflow, it would be better to construct an index in memory and reuse this instead of writing it to a file
 - Global namespace necessary
 - Multi-physics apps already employ these techniques on the compute side
 - But often cannot sustain persistent structure in-situ across jobs due to OS limitations, and more importantly, security

Current state-of-art

- SDSC Gordon SSDs, Mira RamDisk, PSC Sherlock
 - Are they being used in in-situ coupling?
- Commercial apps – common data model in a database in a workflow (EDA, etc.)
 - Will have scaling problem in parallelism
 - Copies, access patterns, are the problems (actually making everything properly work is non-trivial)
- Map-Reduce useful to break the synchronicity of coupling?
 - Off the shelf but wasteful solution? Or a good solution to decoupling for scalability, resiliency?

Why did Language-level interoperable data/object model fail?

- OODB, EJB, Corba, CCA...
 - Used in loosely coupled situation
 - Not “unsuccessful” but limited use in tightly coupled setting
- Will it be successful now in BDEC? With new requirements (application requirements with multiple data sets, extreme scale, usage scenarios, etc.), limitations (data movement expensive, massive contention), and technology (HPC capabilities, NVM, etc.)

Interoperability Issues

Topics included

Data Standards
Data provenance
Access to Models, Resources
Quality & Uncertainty
Discoverability & Annotation
Curating
Ontology, Semantics
Brokers, Identifiers
Workflow tools, exploitation of infrastructure, workflow metadata

1. Integration issues – between different communities (developed their own stds) - Observations and simulation more unclear

Data provenance – fit for purpose issues – fit to integrated – should you use it?
Compare workflows – can I trust the data – which simulation or observations, secondary data – generated by processing -
Metadata – how to work with this through a process – make a composite citation – pointing back to metadata – chain of custody – need data tracker
Reproducibility issues need to be able to be kept track of.
Future research improve – provenance information need to be part of workflow system – how to publish
We don't have well established standards for - how do we generate

2. There are open questions – Data provenance - automated processing – chain of custody and citation of who contributed - improvement with this –

Data and curation and annotation should include quality* – that can be important for decision makers - *fit for purpose

3. How do we design software as a service to go between models

Less developed areas – not at present combine different models not developed on same framework –
Automated workflow – metrics are needed on nodes in workflow – inputs in automated are precisely suitable for input
Signature questions – how to supply metrics –

Things need to be able to be scalable – and realistic – how to process data inputs between models

4. Developing methods and tools to understand the fit for purpose - **how do you build and information system that captures how the data is being used and which purpose – use data sets -**
5. **Ontology and Semantics – identify approaches that are pragmatic to help process along** – potential research project is to assess which pragmatic approaches work in different situations. There are things that exist - how to connect broker these together – between these groups. How align these - are formal descriptions really needed? The HIVE initiative: definition of a vocabulary for different science domains
6. Brokering approach is a useful way to or pragmatic approach - **how do we improve available and usability of the data through brokering** – how develop these collectively – plays a part in cloud computing - synergy pursued computational and data infrastructures– both use and there is not enough synergy between these - also brokering governance infrastructures. Granularity of metadata for brokers because not all data is at the same level of metadata
7. Identifiers – separate topics – brokering technologies to interoperate over identifiers - - build framework – and cross walks – how can we use identifiers in broker frameworks to – what is the major issues with – unique PID is a challenge and needs to be addressed by international approach - endorsed assigning identifiers to data – how will this be done?
8. Workflow tools - what are the major challenges – orchestrating different between infrastructures - how can we make models or data moveable between infrastructures? Problems of handling data exploiting models and data relationship – we need solutions for use different parts of the model. How to orchestrate the infrastructure – and how can we make it easier to move data – how can I use the data once it is moved?
Orchestrating workflows and data flows across e-Infrastructure