The Case for Workflow Analysis

Breakout session

August 5, 2015

Premise - Everything is built on workflows

There are rules for all aspects of system behavior and interaction

- Applications and packages within
- Services
- Datapipes / storage systems
- Memory systems
- Work Schedulers
- Node capabilities
- Etc..

Why consider Workflow in Tools Development

- As an analysis approach
- Exascale
  - Move from commodity approach to integrated
  - More complex ecosystem
  - New runtimes - > I/O pipeline – Memory
  - The need to balance and fully utilize resources
  - What does performance mean?
    - Understand the end goals – what are you trying to measure?
    - The environment impact and possibilities
    - The constraints

As a communication tool

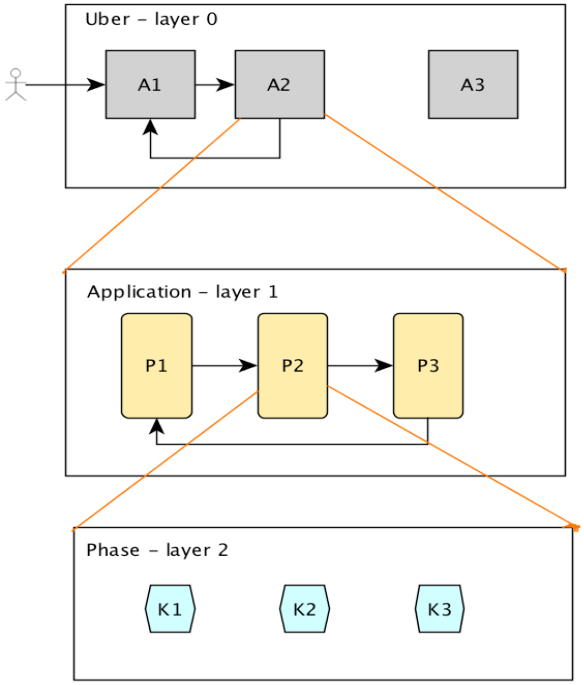
- Do we have an approach to dissect this larger picture to segments for analysis?
- Understanding the eb and flow of the layers of an application (in a given use case) and an environment provide a map to focus – this is the big picture or map
- Need a method to communicate across domain areas – a language
- Concept of workflows from an application perspective
  - carve out the layers
  - see the big picture
  - see the pieces in the context of the big picture
- workflows are a synthesis process for information and knowledge and are a way of breaking things into simpler chunks of information – both from an application and system environment perspective

As a drive toward productivity

- Help identify where to focus – use the map
- Help define interface points that need to be developed - bring stakeholders together
- How to interface with services to use more effectively
- Identifying common APIs across ecosystem – long term

Application Perspective

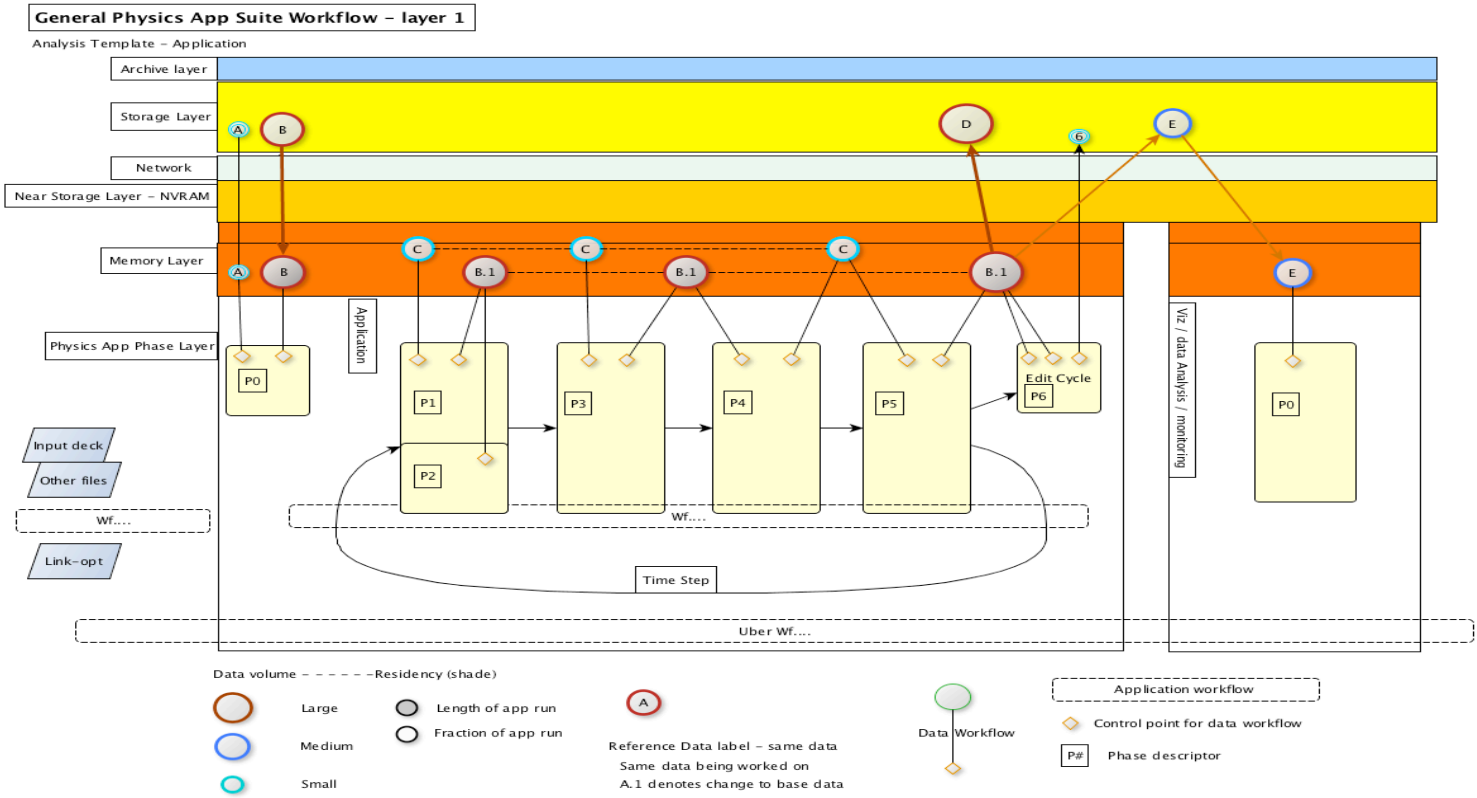
- Workflows are being requested for various use cases –
  - There is no defined approach in HPC – some in distributed computing
  - Look at Business process approach
- Develop approach to characterize workflows
- Needs from development teams / architecture teams / system teams
- Develop common taxonomy across domain areas
- start to generalize
  - Understand the details
  - Expand out to the use cases
- What are the metrics that you are interested in based on the use case? The layer?



Layer 1 – application characterization - example template

Characterize application – what does this mean to different domain specialists?

Layer 1
- physics
- algorithms
- interactions

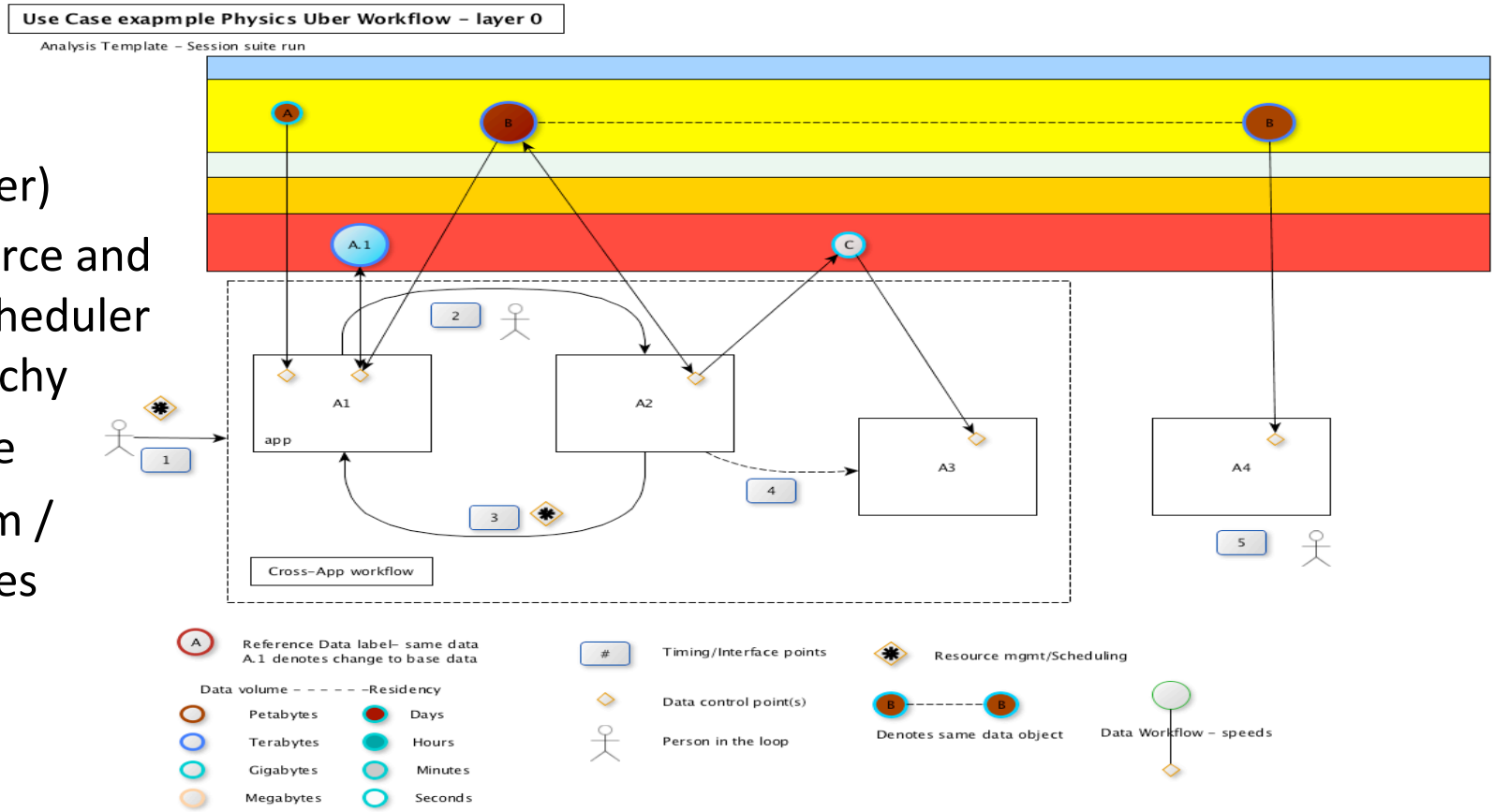


Layer 0 – Ensemble of applications – Use Case – example template

Layer 0

(Uber layer)

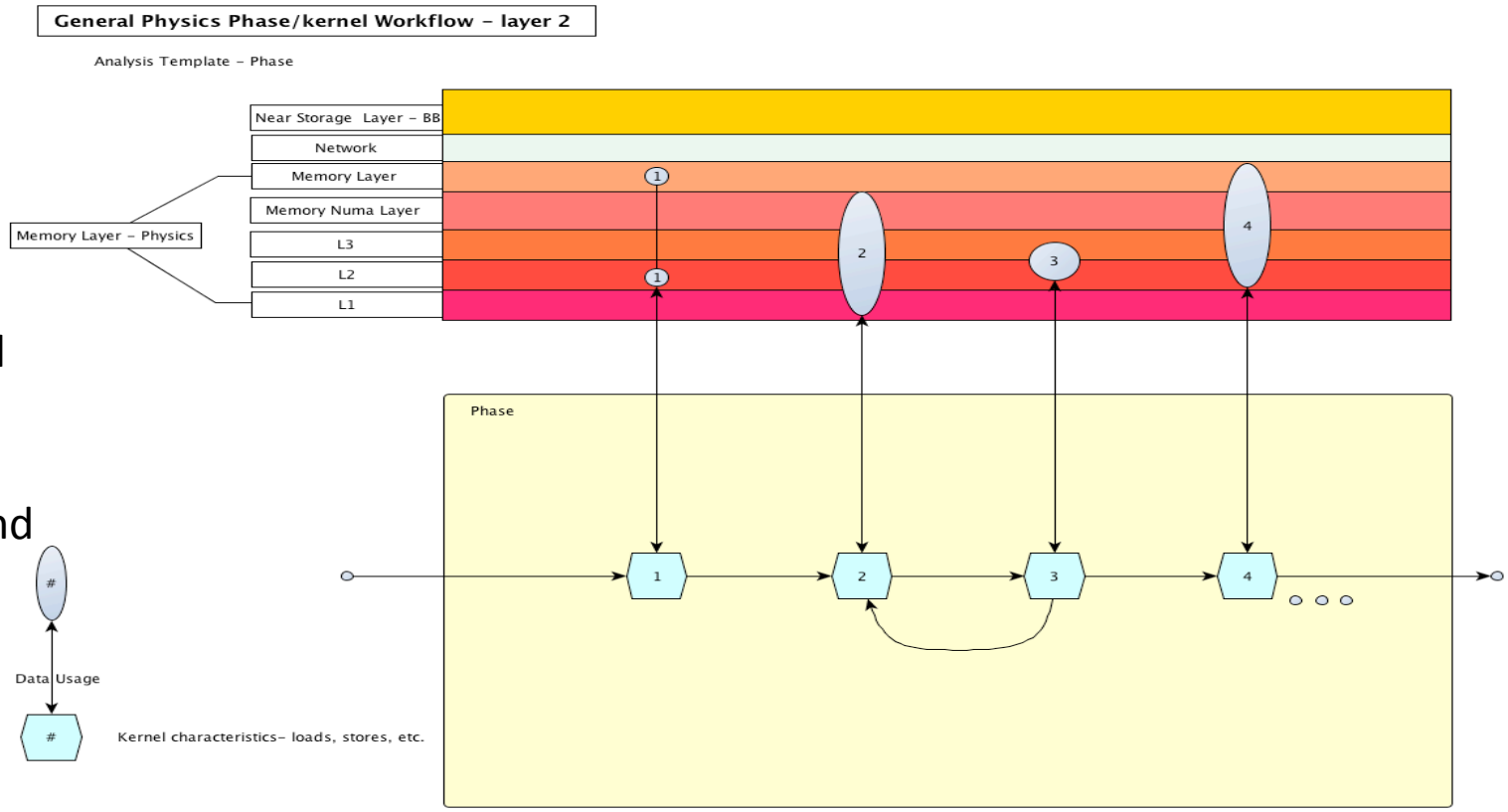
- Resource and job scheduler hierarchy
- People
- System / services



Layer 2 – kernel behavior within a phase – example template

Layer 2

- Lower kernels
- Libraries
- Data startup and shutdown
- Performance measurement and analysis



Random questions and comments

- in situ visualization – extension of an application WF
  - Viz community
- feedback loop to the application
- data mining capability within the simulation
- capture snapshot of the application at runtime
- how to do debugging or performance analysis in different scenarios
- production teams do not understand the workflows – can this help
- exascale is a distributed system in a box
- workflow is a productivity tool
- science per second
- data center operator perspective
- everyone is still in their own world on the production side
- do workflow tools give enough documentation
- there does not exist anything now

What's next?

- Awareness
- Consideration
- Assess Opportunities