Large Data Visualization using Shared Distributed Resources

Jian Huang
Department of Computer Science
University of Tennessee
Knoxville, TN

Background

- To use large-scale shared resources for cutting edge computation jobs is a great idea
  - first coined under the term: “The Grid”
- To implement this vision for production use, several high-level services are needed. For example:
  - Authentication and security control
  - Resource discovery and management
  - Coordinated fail-over
  - Data transfer
  - QoS (reservation, monitoring, diagnostics …)

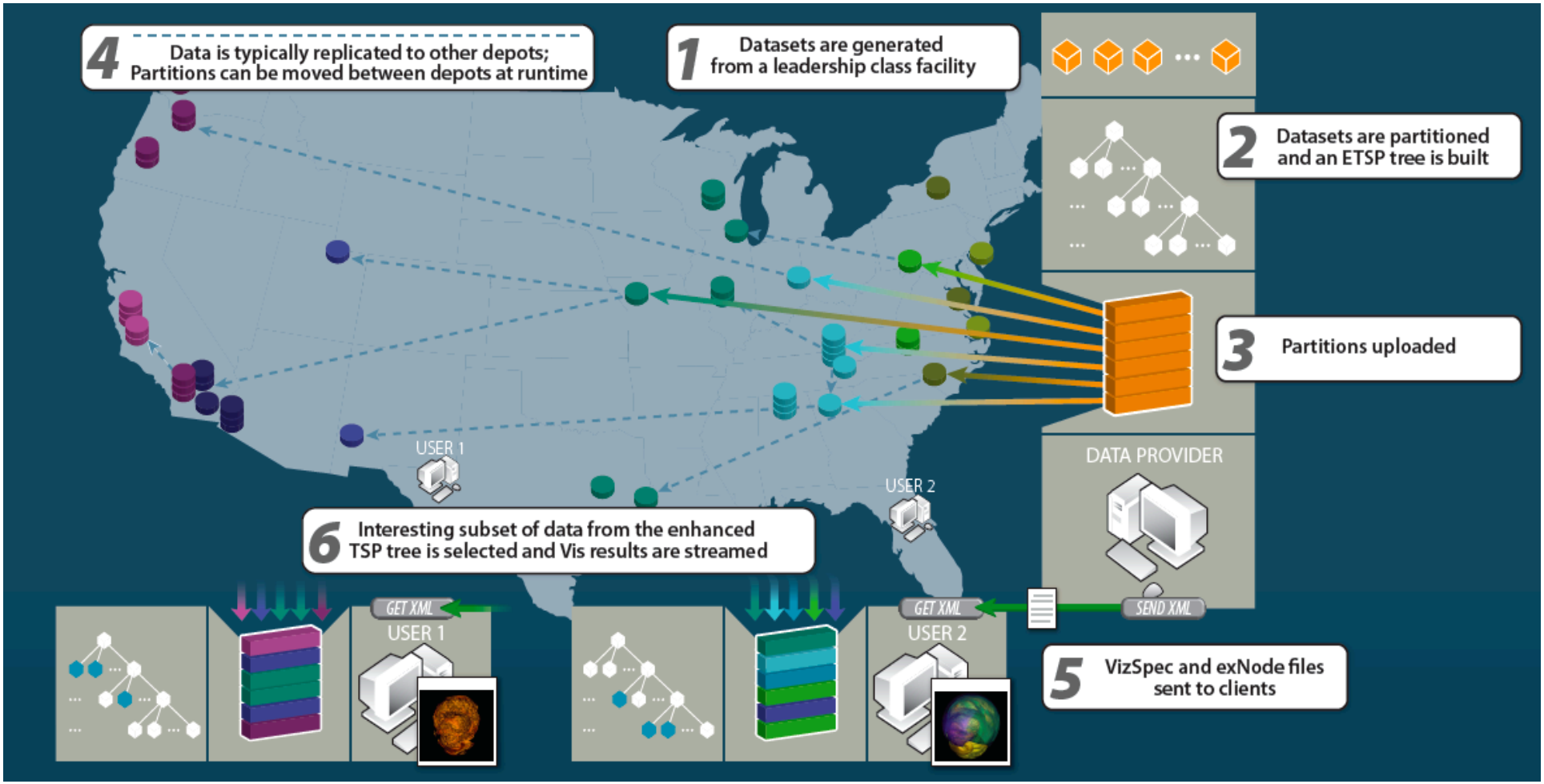
Background (cont.)

- A number of highly acclaimed experimental systems have been launched (with support from Globus, etc.)
- Good time to examine how to implement unique applications optimally
- We would like to focus on large data visualization:
  - Useful when available on-demand
  - Useful when can be shared in an executable form
  - Use as many processors as available (beyond clusters?)
  - Available in a widespread manner
  - Data intensive

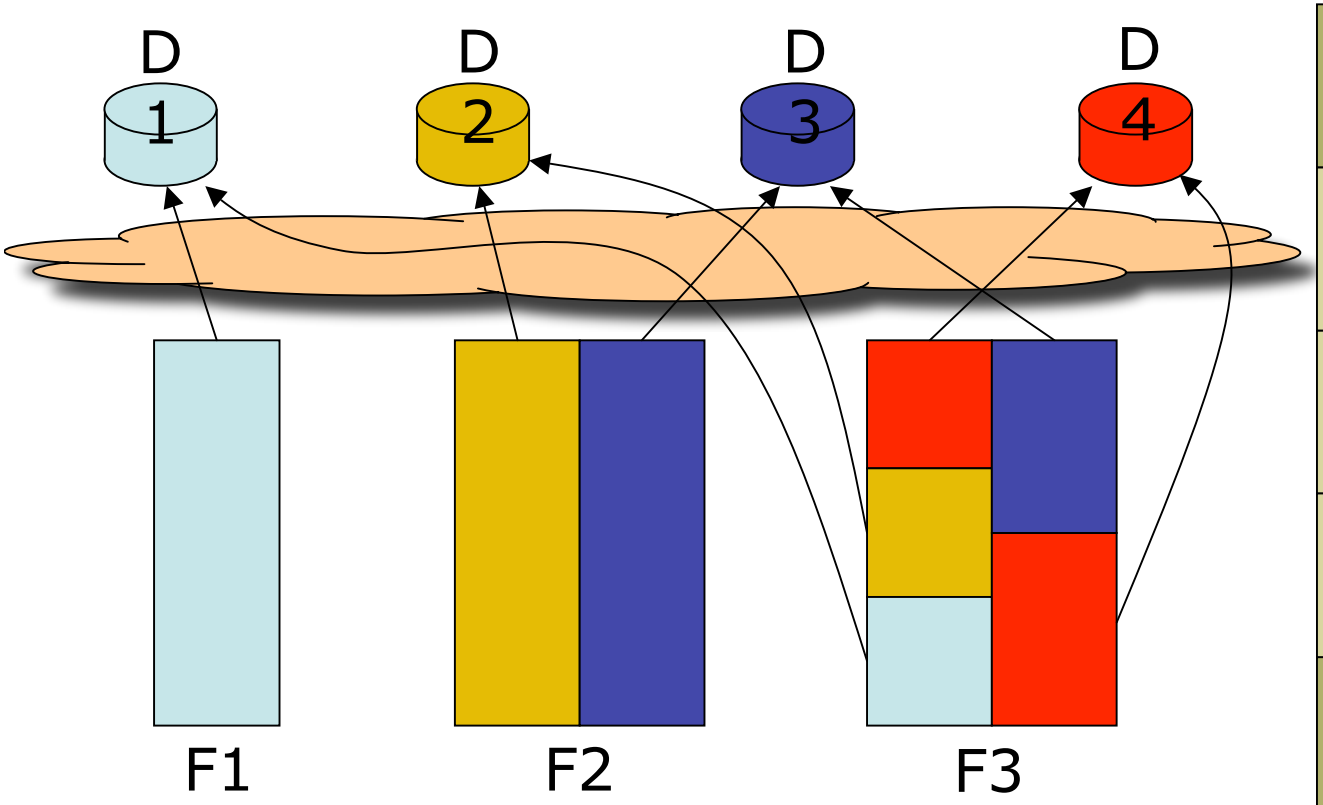
Distributed Visualization

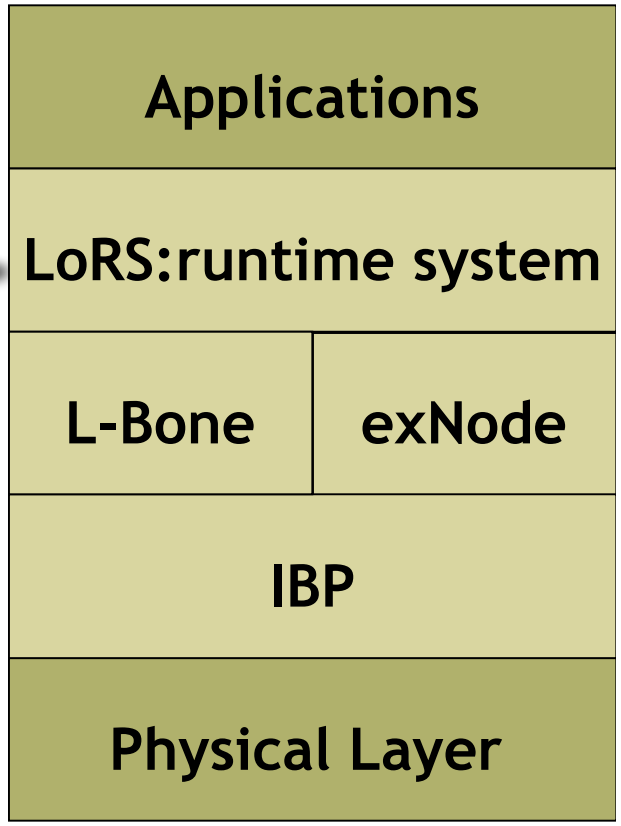
- Our use of this term extends its traditional meaning
  - Still aim to support geographically distributed users
  - The infrastructure does not need to be centralized as in “compute” centers
  - The comp/storage nodes can be independent Internet computers

Distributed Visualization



Data Replication: exNodes





Visualization Operations

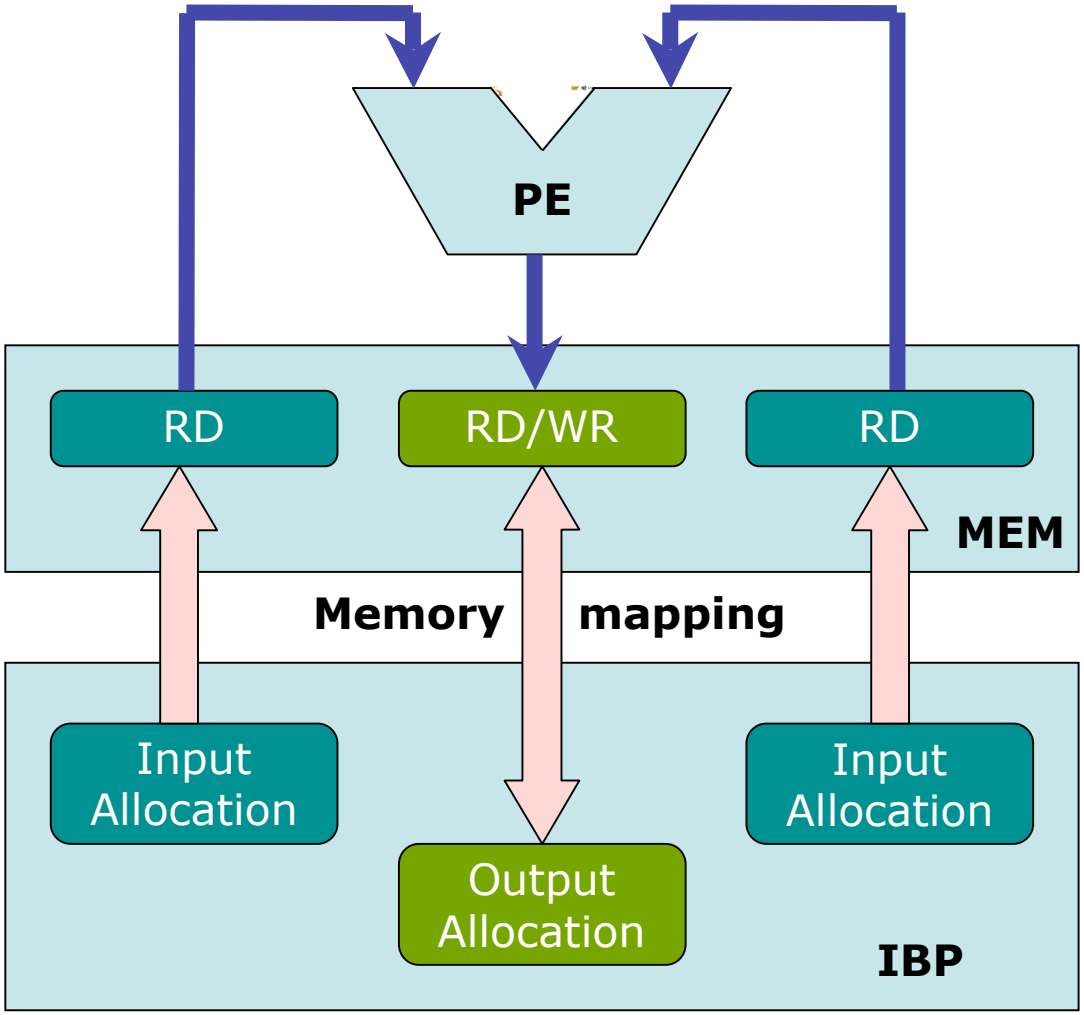
- We constructed a set of basic visualization operations as a highly portable library:
  - the Visualization Cookbook Library (vcblib)
  - includes major visualization algorithms like software volume rendering, iso-surfacing and flow visualization
  - builds and runs on Unix, Linux, Windows and Mac OS.

- vcblib provides a reliable and portable building block to deploy visualization operations to the wide area.

Executing vcblib ops on NFU

- NFU (Network Functional Unit) is a generic, best effort computation service
  - Maximum memory size
  - Limited duration of execution
  - Weak semantics
- Strong services must be constructed on top (I.e. the scheduler of the parallel visualization algorithm)

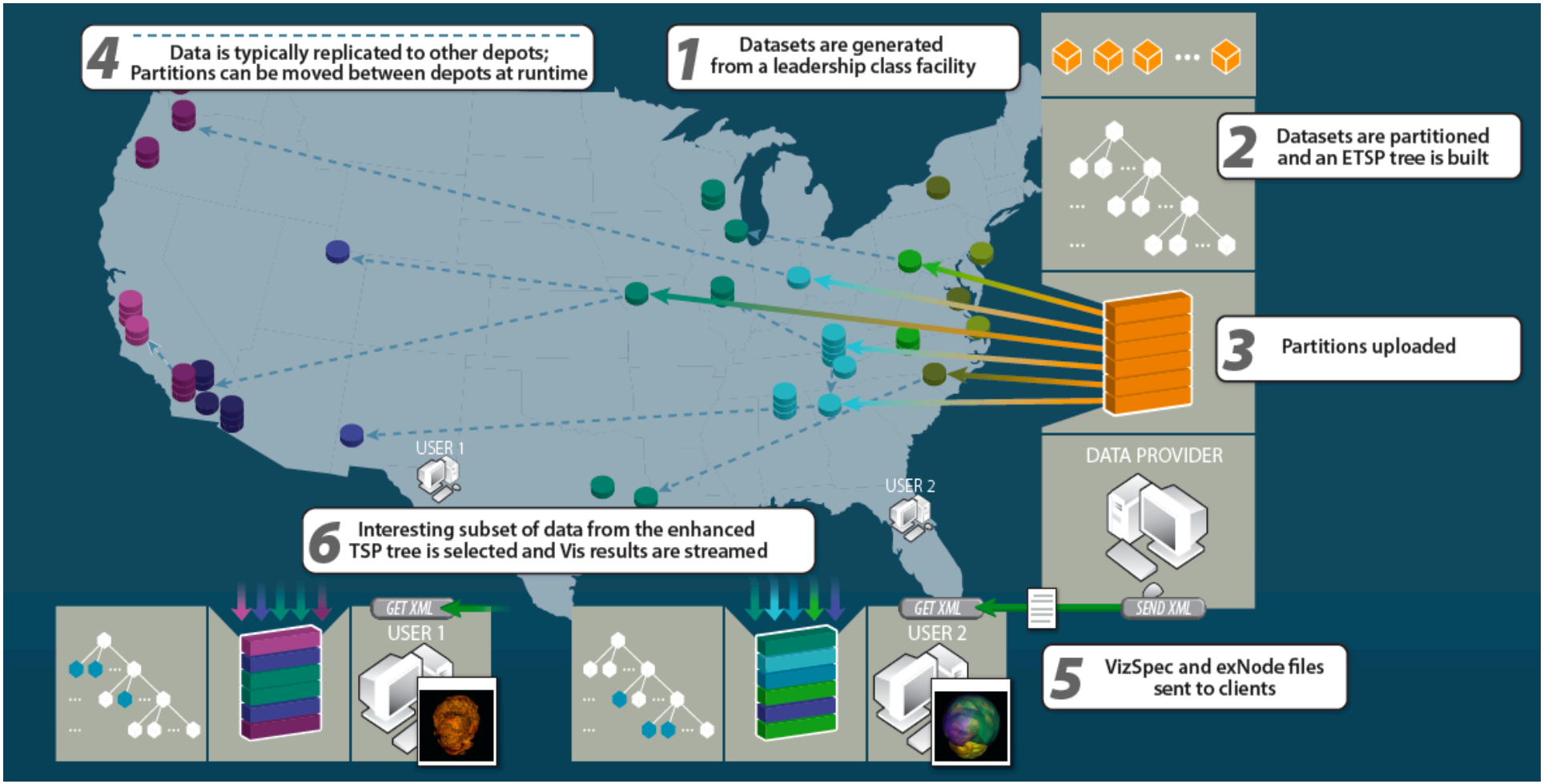
Network Functional Unit



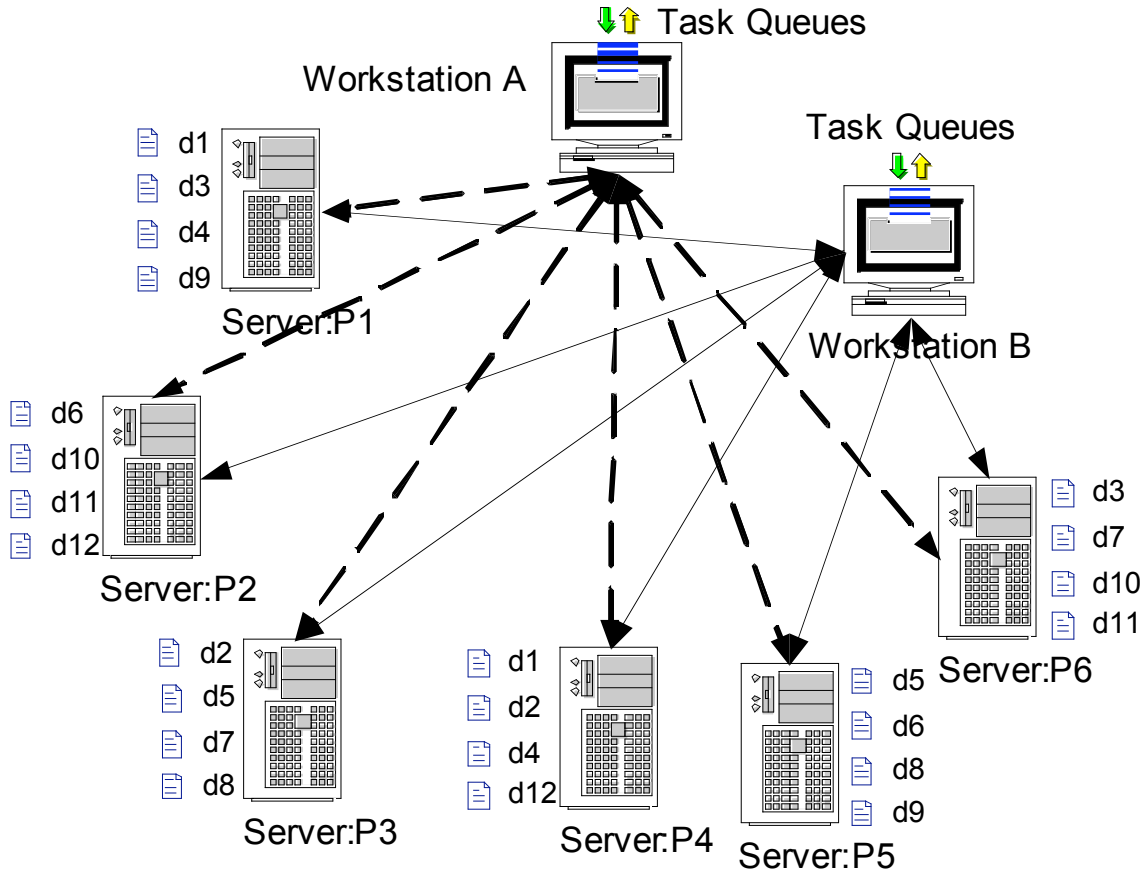
NFU is novel due to:

1. weakened semantic and
2. control of security-sensitive operations.

Distributed Visualization



Scheduling



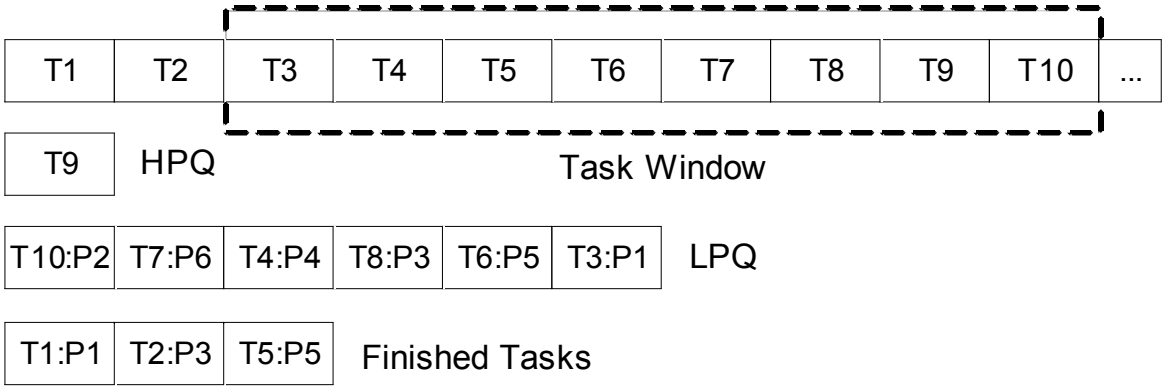
- Depots: {*P1*,*P2*,…,*Pm*}
  Pi described by bw *bi* & computational power *ci*
- Partitioned dataset {*d1*,*d2*,…, *dn*}, k-way replication
- Vis only need one copy of each *dj*
- (Optional) DM tasks
  Mij replicates *dj* on *Pi*

Key Challenge:

Resource performance varies over time !!!

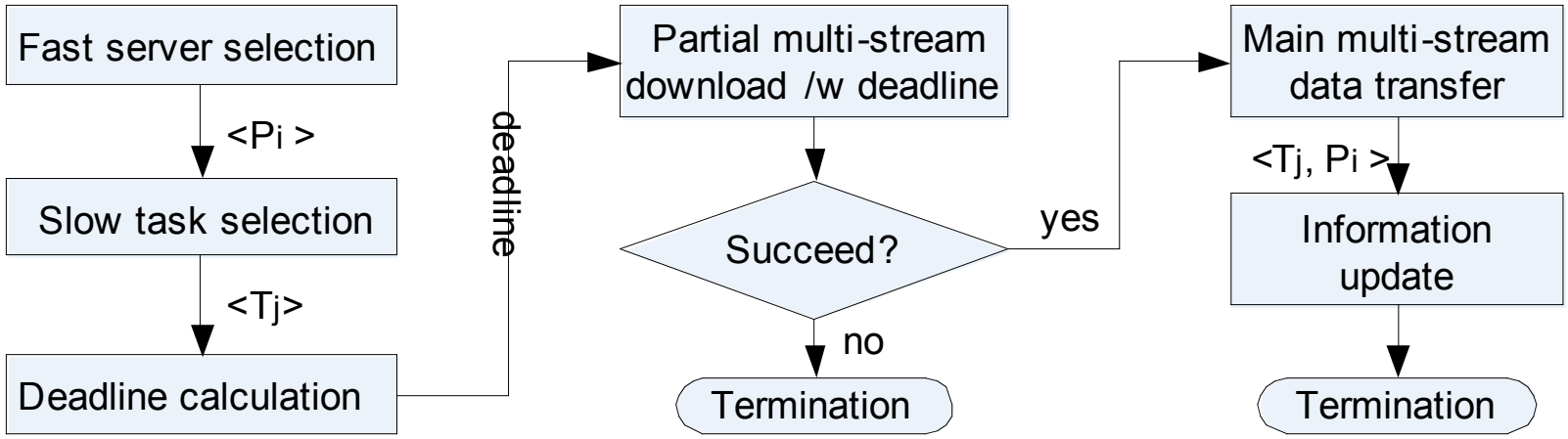
Scheduling

- Depots are ranked by number of volume partitions processed so far
- High vs. Low priority queues (HPQ vs. LPQ) of tasks
  - HPQ: tasks-to-be-assigned, keyed by shortest potential processing time
  - LPQ: tasks-already-assigned, keyed by longest potential wait time



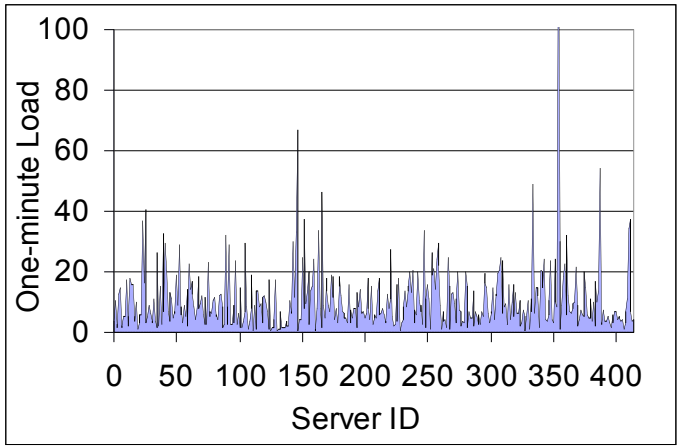
Dynamic Data Movement

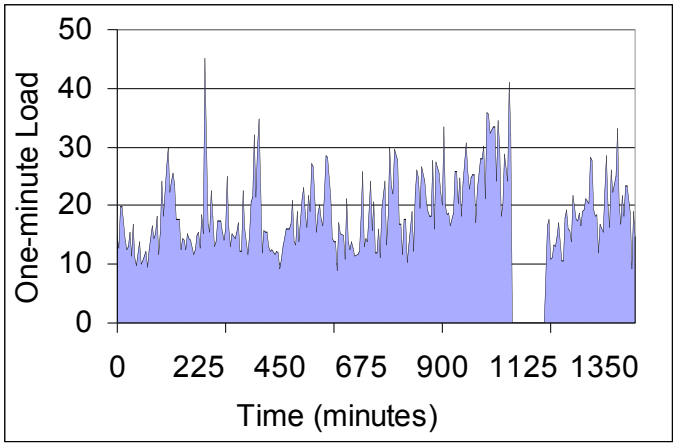
- Some data partitions are just “unlucky” to be on slow or heavily loaded servers
- After fast depots are done with local tasks, can dynamically “steal” some slow “partitions”



Results: the depots

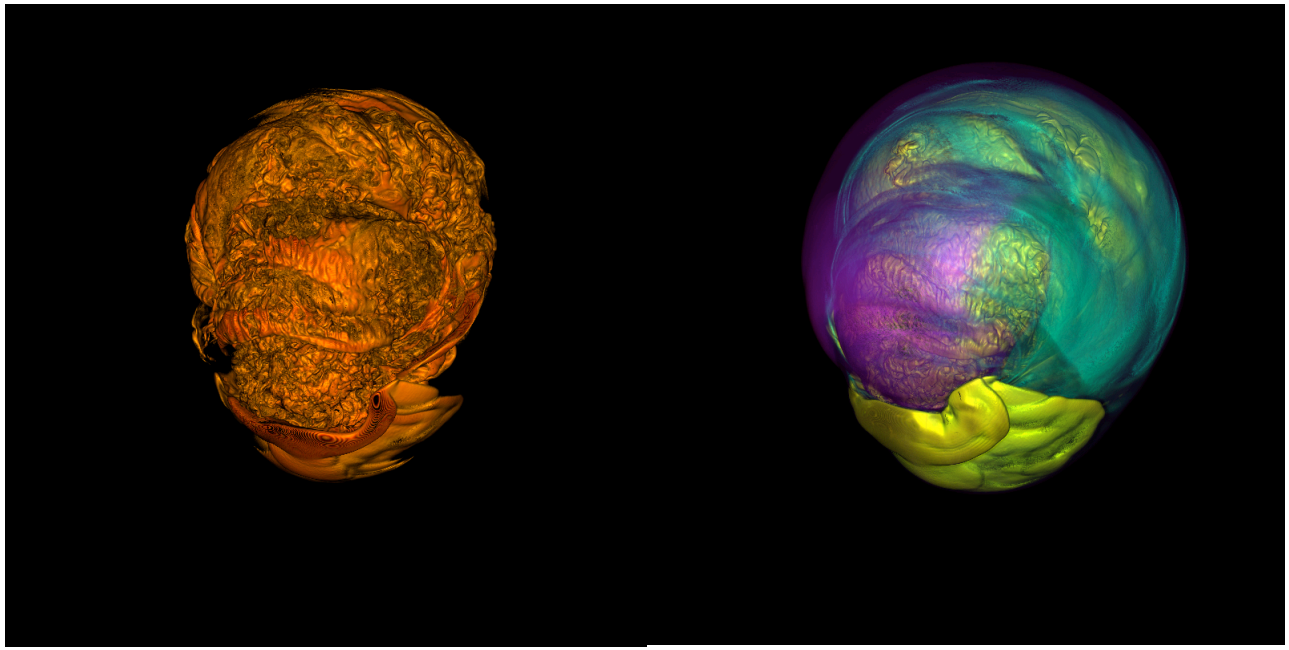
- Most of our depots are run by the Planet-Lab project
- The machines workload varies much from one to one
- The workload is also highly time varying





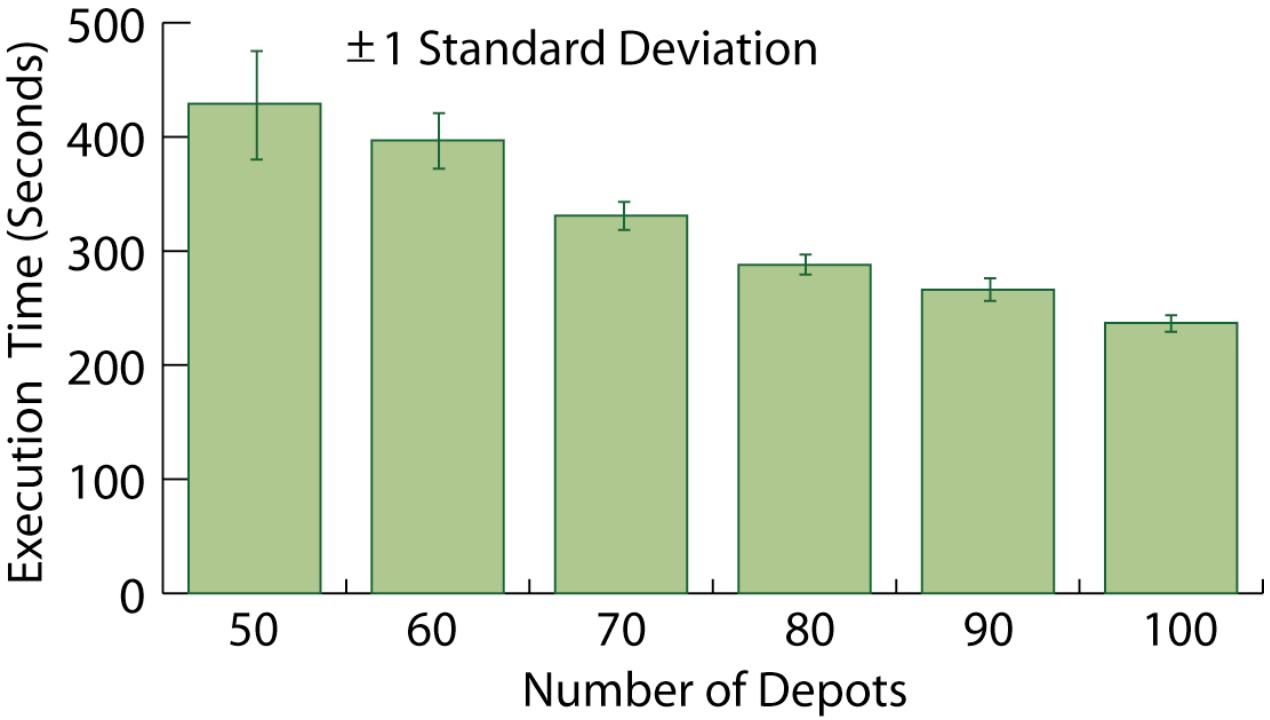
Results: the data

- Test data: 30 timestep of Tera-scale Supernova Initiative, 75GB in total
  - Provided by Tony Mezzacappa (ORNL) and John Blondin (ORNL) under the auspices of DOE SciDAC TSI project



Results: the performance

- 800x800 image resolution, 0.5 step size in ray-casting, per-fragment classification and Phong shading
- With 100 depots, the average rendering time: 237 sec



To the User

- You program your visualization by editing an XML file
  - ASCII file, 3KB in size
  - A template is provided

```
<viewing_parameters>
        <lighting num_lights=2> ... <lighting>
        <viewport_transform> ... <viewport_transform>
        <data_specific> ... <data_specific> ...
<viewing_parameters>
<raycasting>
        <scale> ... </scale>
        <color_scheme> ... </color_scheme>
        <interval_range> ... </interval_range>
</raycasting> ...
```

- Live Demo on SC06 Exhibit
  - 1-2 pm ORNL booth/Vanderbilt booth

Acknowledgment

- Seelab, CS/U. Tenn
  - Andrew Gaston
  - Jian Huang

  http://www.cs.utk.edu/~seelab

- ORNL
  - Jinzhu Gao

- Planet-Lab project provided access to many of the hardware

  http://www.planet-lab.org

- LoCI lab, CS/U. Tenn
  - Huadong Liu
  - Micah Beck
  - Terry Moore

  http://www.cs.utk.edu/~loci

- Funded by:
  - NSF CNS-0437508
  - DOE DE-FG02-04ER25610
  - DOE DE-FG02-06ER06-04