Testing Dynamic Data Allocation Algorithms Within a GRID Simulator

Thursday, July 23rd, 2009

Student: William Boyd, Georgia Tech
Supervisor: Mario Lassnig, CERN PH-ADP-DDM
Student Colleague: Martin Barisits, University of Vienna

A Little Review From Last Time...

- Goals for the summer
  - Design and build a GRID simulator
- What's the point?
  - To model present load conditions and congestion within the GRID
  - To test dynamic data distribution algorithms for improved performance
  - All without disrupting ongoing activity within the GRID

MICHIGAN

CERN

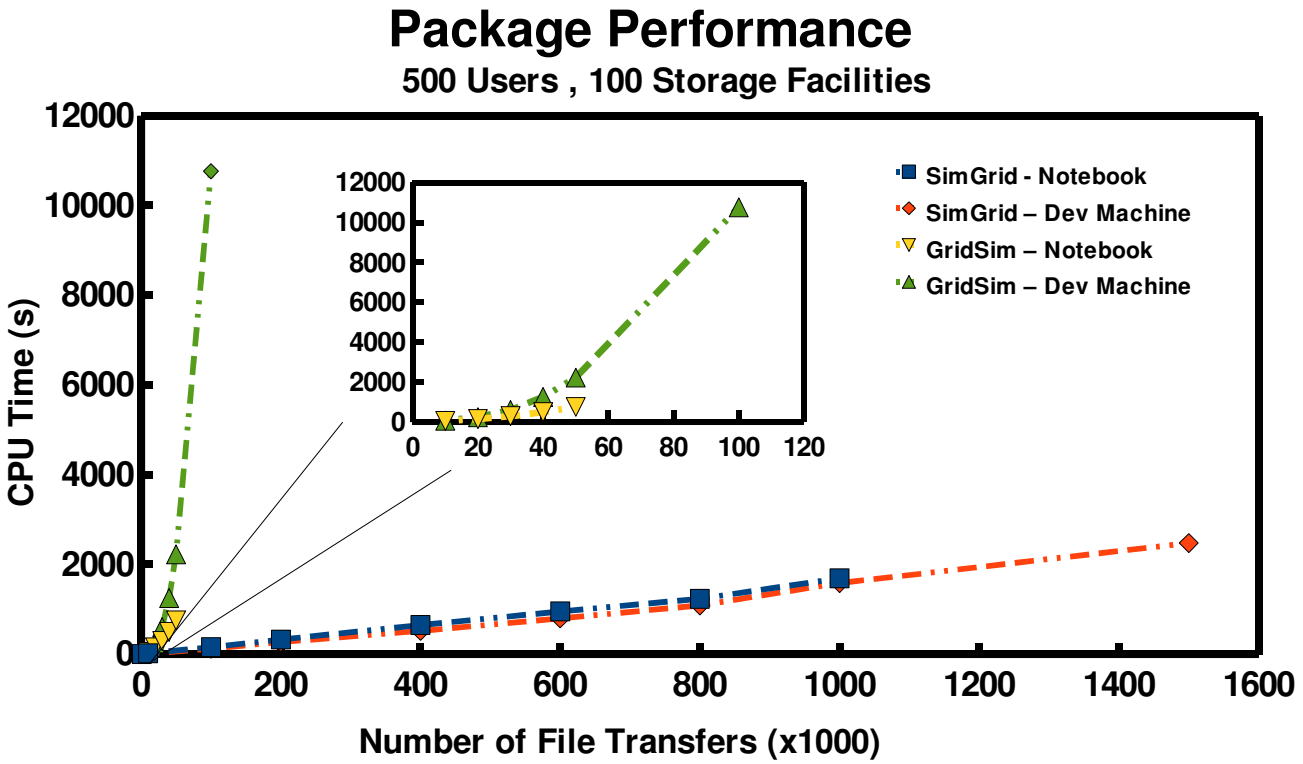
SimGrid vs. GridSim vs. SimPy ...

- Evaluation of GRID/Cloud Computing Simulation Packages
  - SimGrid
    - Written in C
    - Pros: Fast execution time; low memory consumption; scalable
    - Cons: Lacks some necessary functionality
  - GridSim
    - Written in Java
    - Pros: Highly developed; excellent internal logging of network traffic
    - Cons: Very slow execution time; memory consumption not scalable
  - SimPy
    - Written in Python
    - Pros: Fast execution time; Python easy to code in
    - Cons: Least developed of all three packages

MICHIGAN

CERN

And the Winner is....



- ...SimGrid
  - Attempted to simulate one day on GRID (~1.5 million file transfers)
  - GridSim: exponential scaling in CPU time with increasing transfers
  - SimGrid: linear scaling in CPU Time with increasing transfers

Ongoing Work

- TopologyGen.py - a topology generator
  - Python script to define links between nodes on the GRID
  - Parses “TiersOfATLASCache.py” for all Tier-1 and Tier-2 nodes within each GRID cloud
  - Queries DQ2 database to find used and free disk space at each node (among other things)
  - Compiles information into two XML files which define the GRID topology for the simulator

```
7438    <route src='CERN_52' dst='RAL-LCG2_MCDISK'>
7439        <link:ctn id='RAL-LCG2_MCDISK_InternalLink'>
7440        </link:ctn>
7441        <link:ctn id='RAL_OPNLinkInternal'>
7442        </link:ctn>
7443        <link:ctn id='CERN_52_InternalLink'>
7444        </link:ctn>
7445    </route>
7446    <route src='CERN_53' dst='RAL-LCG2_MCDISK'>
7447        <link:ctn id='CERN_53_InternalLink'>
7448        </link:ctn>
```

Platform.xml

```
4372    <process function='Tier1Storage' host='BNL-OSG2_DATADISK'>
4373        <argument value='1'>
4374        </argument>
4375        <argument value='980191641'>
4376        </argument>
4377        <argument value='796762983'>
4378        </argument>
4379        <argument value='0.03496'>
4380        </argument>
4381        <argument value='0.0'>
4382        </argument>
4383        <argument value='0.0'>
4384        </argument>
```

Deployment.xml

Ongoing Work (cont.)

- LoadGen.py - a load generator
  - Python script to simulate file transfer traffic on GRID
  - Generates CSV files for each active node
    - Unique file ID
    - Target node
    - File size
    - Inter-arrival time
- Plotting facility
  - Basic user interface in MatPlotLib Python library
  - Plot disk space evolution on each node following simulation
  - Plot file transfers vs. time on each link following simulation
  - Currently each link and node reports to its own log file – may implement database to log simulation history

MICHIGAN

CERN

Any Questions?

Dachau, Germany

Botanical Gardens, Zurich, Switzerland

Amsterdam, Netherlands