Testing Dynamic Data Allocation Algorithms Within a GRID Simulator

William Boyd

Georgia Institute of Technology

August 13, 2009
Introduction
The Problem

- Goal: Test data distribution strategies
- Need: GRID simulator
- Need: Ability to load the simulator with the current GRID topology
- Need: Inject the simulator with realistic workloads
- Process results
Approach
• **Two basic challenges**
  • Create a tool to get a snapshot of the whole GRID environment (network topology) and to generate Loads
  • Create a simulator which can operate on this input

• Different simulators for cloud computing are available in the research community

• For time reasons we decided to use a simulator package for a starting framework
Two basic challenges

- Create a tool to get a snapshot of the whole GRID environment (network topology) and to generate Loads
- Create a simulator which can operate on this input

Different simulators for cloud computing are available in the research community

For time reasons we decided to use a simulator package for a starting framework
• Two basic challenges
  • Create a tool to get a snapshot of the whole GRID environment (network topology) and to generate Loads
  • Create a simulator which can operate on this input
• Different simulators for cloud computing are available in the research community
• For time reasons we decided to use a simulator package for a starting framework
Package Performance

- Evaluated two cloud computing simulation packages
- Attempted to simulate one day on GRID (1.5 million file transfers)
- GridSim: exponential in CPU time with increasing transfers
- SimGrid: linear in CPU time with increasing transfers
Topology Generator
GRID Sites

GRID sites across the world
ATLAS Computing Model

- Hierarchical computing network
  - Tier-0
  - Tier-1
  - Tier-2
- Tier-0 (CERN) generates data
- Tier-1s store data
- Tier-2s process data

The Tier-0 and Tier-2 network configuration[1]
ATLAS Computing Model

- Hierarchical computing network
  - Tier-0
  - Tier-1
  - Tier-2
- Tier-0 (CERN) generates data
  - Tier-1s store data
  - Tier-2s process data

The Tier-0 and Tier-2 network configuration[1]
ATLAS Computing Model

- Hierarchical computing network
  - Tier-0
  - Tier-1
  - Tier-2
- Tier-0 (CERN) generates data
  - Tier-1s store data
  - Tier-2s process data

The Tier-0 and Tier-2 network configuration[1]
ATLAS Computing Model

- Hierarchical computing network
  - Tier-0
  - Tier-1
  - Tier-2
  - Tier-0 (CERN) generates data
  - Tier-1s store data
  - Tier-2s process data

The Tier-0 and Tier-2 network configuration[1]
The Topology Generator

- **TopologyGen.py**
  - Script to construct GRID topology
  - Parses TiersOfATLASCache.py
  - Finds and associates Tier-1s and Tier-2s
  - Queries the DQ2 database
  - Total disk space capacity
  - Used disk space
  - Topology is written to two XML files
The Topology Generator

- **TopologyGen.py**
  - Script to construct GRID topology
  - Parses TiersOfATLASCache.py
  - Finds and associates Tier-1s and Tier-2s
  - Queries the DQ2 database
  - Total disk space capacity
  - Used disk space
  - Topology is written to two XML files
The Topology Generator

- TopologyGen.py
  - Script to construct GRID topology
  - Parses TiersOfATLASCache.py
  - Finds and associates Tier-1s and Tier-2s
    - Queries the DQ2 database
    - Total disk space capacity
    - Used disk space
    - Topology is written to two XML files
The Topology Generator

- TopologyGen.py
  - Script to construct GRID topology
  - Parses TiersOfATLASCache.py
  - Finds and associates Tier-1s and Tier-2s
  - Queries the DQ2 database
    - Total disk space capacity
    - Used disk space
    - Topology is written to two XML files
The Topology Generator

- TopologyGen.py
  - Script to construct GRID topology
  - Parses TiersOfATLASCache.py
  - Finds and associates Tier-1s and Tier-2s
  - Queries the DQ2 database
  - Total disk space capacity
    - Used disk space
  - Topology is written to two XML files
The Topology Generator

- TopologyGen.py
  - Script to construct GRID topology
  - Parses TiersOfATLASCache.py
  - Finds and associates Tier-1s and Tier-2s
  - Queries the DQ2 database
  - Total disk space capacity
  - Used disk space
  - Topology is written to two XML files
The Topology Generator

- TopologyGen.py
  - Script to construct GRID topology
  - Parses TiersOfATLASCache.py
  - Finds and associates Tier-1s and Tier-2s
  - Queries the DQ2 database
  - Total disk space capacity
  - Used disk space
  - Topology is written to two XML files
Platform and Deployment Files

- **Platform file**
  - Node declarations
  - Link declarations
  - Route declarations

- **Deployment file**
  - Logfiles for each node
  - Total and used disk space
  - Used disk space by datatype
  - Tier-0 loadfiles
  - Associated Tier-1s and Tier-2s
Platform and Deployment Files

- **Platform file**
  - Node declarations
  - Link declarations
  - Route declarations

- **Deployment file**
  - Logfiles for each node
  - Total and used disk space
  - Used disk space by datatype
  - Tier-0 loadfiles
  - Associated Tier-1s and Tier-2s
Platform and Deployment Files

- **Platform file**
  - Node declarations
  - Link declarations
  - Route declarations

- **Deployment file**
  - Logfiles for each node
  - Total and used disk space
  - Used disk space by datatype
  - Tier-0 loadfiles
  - Associated Tier-1s and Tier-2s
The topology that is generated for simulation
Load Generator
Generating a Load

- Loadfile given to each Tier-0
- Loadfiles define dataset transfers
  - Unique dataset ID
  - Random (uniform) target Tier-1 storage node
  - Random (uniform) filesize (0.5-6GB)
  - Random (weekly distribution) inter-submission time
  - Dataset datatype (i.e., RAW)
Generating a Load

- Loadfile given to each Tier-0
- Loadfiles define dataset transfers
  - Unique dataset ID
  - Random (uniform) target Tier-1 storage node
  - Random (uniform) filesize (0.5-6GB)
  - Random (weekly distribution) inter-submission time
  - Dataset datatype (i.e., RAW)
Simulator
Features

• Builds network topology according to the injected topology File
• Simulates the shipment and processing of datasets
• Gives the user a framework to implement data distribution strategies
• Provides functions to write simulation output...
Features

- Builds network topology according to the injected topology File
- Simulates the shipment and processing of datasets
- Gives the user a framework to implement data distribution strategies
- Provides functions to write simulation output...
Features

- Builds network topology according to the injected topology File
- Simulates the shipment and processing of datasets
- Gives the user a framework to implement data distribution strategies
- Provides functions to write simulation output...
Features

- Builds network topology according to the injected topology File
- Simulates the shipment and processing of datasets
- Gives the user a framework to implement data distribution strategies
- Provides functions to write simulation output...
The way of a DataSet (1/4)
The way of a DataSet (2/4)
The way of a DataSet (3/4)
Simulation Results
Overloading the GRID

Tier-0 Dataset submission distribution

Disk space evolution with increasing daily dataset transfers
Data Storage by Datatype

Uniform dataset transfer distribution

Simulated dataset transfer distribution

Histogram of Dataset Transfers

Histogram of Dataset Transfers
Simulation Results

Scalability

Scalability of **MARTINWILLSIM**

- MartinWillSim run to simulate increasing number of days
- 250,000 tasks/day (800TB/day)
- Simulated one month of dataset transfers in 40 mins.
- CPU time linear with number of simulated tasks

![Graph showing MartinWillSim Performance](image)
Conclusion
Recap

- Evaluation of different Simulator Packages [2 Weeks]
- Design and Implementation of:
  - Topology & Load Generator [6 Weeks]
  - MartinWillSim Simulator [6 Weeks]
- Result Analysis
- Documentation
Recap

- Evaluation of different Simulator Packages [2 Weeks]
- Design and Implementation of:
  - Topology & Load Generator [6 Weeks]
  - MartinWillSim Simulator [6 Weeks]
- Result Analysis
- Documentation
Recap

- Evaluation of different Simulator Packages [2 Weeks]
- Design and Implementation of:
  - Topology & Load Generator [6 Weeks]
  - MartinWillSim Simulator [6 Weeks]
- Result Analysis
- Documentation
Recap

- Evaluation of different Simulator Packages [2 Weeks]
- Design and Implementation of:
  - Topology & Load Generator [6 Weeks]
  - MartinWillSim Simulator [6 Weeks]
- Result Analysis
- Documentation


Thank you!

- Mario Lassnig
- Vincent Garonne
- Dr. Homer Neal
- Dr. Jean Krisch
- Dr. Myron Campbell
- Jeremy Herr
- Jeremy Ticey
- Ingrid Schmid