The Virtual Grid Application Development Software (VGrADS) Project

Overview

Ken Kennedy
VGrADS Director
Rice University

http://vgrads.rice.edu/
The VGrADS Team

- VGrADS is an NSF-funded Information Technology Research project

Dan Reed

Keith Cooper
Ken Kennedy
Charles Koelbel
Richard Tapia
Linda Torczon

Fran Berman
Andrew Chien
Henri Casanova

Rich Wolski

Lennart Johnsson

- Plus many graduate students, postdocs, and technical staff!
Vision: Global Distributed Problem Solving

• Where We Want To Be
  o Transparent Grid computing
    - Submit job
    - Find & schedule resources
    - Execute efficiently

• Where We Are
  o Low-level hand programming
  o Programmer must manage:
    - Heterogeneous resources
    - Scheduling of computation and data movement
    - Fault tolerance and performance adaptation

• What Do We Propose as A Solution?
  o Separate application development from resource management
    - Through an abstraction called the Virtual Grid
  o Provide tools to bridge the gap between conventional and Grid computation
    - Scheduling, resource management, distributed launch, simple programming models, fault tolerance, grid economies
VGrADS Big Ideas

• Virtualization of Resources
  o Application specifies required resources in Virtual Grid Definition language (vgDL)
    - Give me a loose bag of 1000 processors, with 1 Gb memory per processor, with the fastest possible processors
    - Give me a tight bag of as many Opterons as possible
  o Virtual Grid Execution System (vgES) produces specific virtual grid matching specification
  o Avoids need for scheduling against the entire space of global resources

• Generic In-Advance Scheduling of Application Workflows
  o Application includes performance models for all workflow nodes
    - Performance models automatically constructed
  o Software schedules applications onto virtual Grid, minimizing total makespan
    - Including both computation and data movement times
Virtual Grids (VGs)

- A Virtual Grid (VG) takes
  - Shared heterogeneous resources
  - Scalable information service
- and provides
  - An hierarchy of application-defined aggregations (e.g. `ClusterOf`) with constraints (e.g. processor type) and rankings
- Virtual Grid Execution System (vgES) implements VG
  - VG Definition Language (vgDL)
  - VG Find And Bind (vgFAB)
  - VG Monitor (vgMON)
  - VG Application Launch (VgLAUNCH+DVCW)
  - VG Resource Info (vgAgent)
VGrADS Tool Research

• Scheduling of workflow computations
  o Off-line look-ahead scheduling dramatically improves in total time
  o Accurate performance models significantly affect quality of scheduling
  o Batch queue behavior can be predicted accurately enough for scheduling decisions

• Fault tolerance
  o Diskless checkpointing for linear algebra computations (application-specific)
  o Temporal reasoning for fault prediction
  o Optimal checkpoint frequency for iterative applications
VGrADS: What’s New

• **SC’04**
  - Scheduling EMAN application
    - Aware of performance models

• **SC’05**
  - Find and Bind (FAB) for resource selection
  - Scheduling EMAN application
    - Aware of batch queue predictions (and performance models)

• **SC’06**
  - Virtual Grid "slots" for resource availability
    - Start time + duration
    - Uses advance reservations where available
    - Uses batch queue prediction elsewhere
  - Scheduling LEAD application
    - Aware of reservations and batch queue predictions (and performance models)
The LEAD Vision: A Paradigm Shift

DYNAMIC OBSERVATIONS

Models and Algorithms Driving Sensors

Analysis/Assimilation
- Quality Control
- Retrieval of Unobserved Quantities
- Creation of Gridded Fields

Prediction/Detection
- PCs to Teraflop Systems

Product Generation, Display, Dissemination

End Users

The CS challenge: Build cyberinfrastructure services that provide adaptability, scalability, availability, useability, and real-time response.
LEAD Portal – Experiment Builder
VGrADS Application Collaboration

LEAD
Linked Environments for Atmospheric Discovery

Workflow Configuration Service
Create Services
App. Factory
Launch Services
Application Service (per task)
Run workflow one step at a time
Workflow and File Status
Event Broker
myLEAD (subscribes to messages from the broker and knows what magic to do with input/output files and talks to RLS/DRS)

Run job
Job Notification
Scheduler Mapper
Batch Queue Prediction
Virtual Grid Execution System
Scheduler
LEAD Resource Broker
Performance Model
Schedule toward a workflow deadline
Annotated DAG
DAG + Constraint

Portal
LEAD BPEL Workflow Engine

Linked Environments for Atmospheric Discovery
Schedule toward a workflow deadline

- DAG + Constraint
- Annotated DAG

**Resource Broker**

**Scheduler Mapper**

- Use performance model and map the tasks to the slots. If deadline can’t be met, return.
- Query the performance model for task’s resource requirements
- Find me two slots (vgFind)
- Bind Resources (vgBind)
- Return slots above threshold

**Performance Model**

- Here is the workflow and constraints + pointer to performance model. Give me a mapping
- Query the performance model for task’s resource requirements

**Virtual Grid Execution System**

**Batch Queue Prediction**

- Constantly collecting data over time
- Query Batch Queue prediction about probabilities of getting slots

**GT4 GRAM**

- If reserved submit PBS-glidin at slot start time else submit when BQP suggests

**PBS**

- Slot
- Slot

**Globus Gateway**

- Run Job

**Run job**

**Job Notification**

- If not reserved resource, ask - Is it time to submit?

- Constantly collecting data over time

**Virtual Grid Application Development Software Project**
Some Future Challenges

- Parallelism in the LEAD workflow manager
  - Parallel steps in different slots or within one slot

- Accurate Slot Requests Through Preliminary Scheduling
  - Minimization of wasted slot time
    - Accurate scheduling, better queue prediction
    - Dynamic adaptation of slot reservations
  - Requires some form of resource equivalence:
    - For step B, I need the equivalent of 200 Opterons, where 1 Opteron = 3 Itanium = 1.3 Power 5 (from perf models)

- Increased Schedule Robustness
  - Minimizing variation along the critical path

- Scheduling to Minimize Cost
  - In the presence of cycle exchange rates
  - Get the minimum-cost resources to solve the problem by the given deadline
VGrADS at SC’06

• Booth Talks and Demos
  o Tuesday, noon - GCAS booth (1825)
  o Tuesday, 2:30 - USC booth (2246) [Not live]
  o Wednesday, 1:00 - SDSC booth (1915)
  o Thursday, 10:30 - RENCI booth (1143)
  o What you’ll see
    - LEAD running on several clusters
    - Scheduler mapping LEAD components to slots
    - vgES managing slots via batch queue prediction

• Papers
  o “Improving Grid Resource Allocation via Integrated Selection and Binding” by Kee, et al. - Wednesday, 10:30
  o “Toward a Doctrine of Containtment: Grid Hosting with Adaptive Resource Control” by Ramakrishnan, et al. - Wednesday, 11:00
Launching from the LEAD Portal

• Work in Progress
Experiment Builder Portlet

Customize

- I have SPRUCE tokens and I would like to have the option of running SPRUCE workflows.
- Use the VGFADS Scheduler when running my workflows.
- Submit my workflows to the Workflow Configuration Service (WICS)

Submit

Copyright © 2008 LEAD Environment for Atmospheric Discovery. All rights reserved.
Experiment Builder Portlet

Experiment Wizard

User: VGRADS Demo  Project: SC-Testing
Name: VGRADS Workflow
Description:
Workflow: Case-Study NAM Initialized-WRF-Forecast

Select options for NAM data files required for this experiment

Description: Choose a set of NAM data files for generating interpolated boundary condition files. For a 6 hour forecast, please choose NAM forecast time steps ending with "180", "300", "420" and for a 12 hour forecast please choose NAM forecast time steps ending in "240", "360", "480" and "600". Note that all time step files should be from the same model run.

<table>
<thead>
<tr>
<th>Select</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Default option 0</td>
<td>The value of the default option is [ftp://grid-hg.ncsa.illinois.edu/ftp_scratch/ef/efac/20060905170000/0]</td>
</tr>
<tr>
<td></td>
<td>Default option 1</td>
<td>The value of the default option is [ftp://grid-hg.ncsa.illinois.edu/ftp_scratch/ef/efac/20060905170000/1]</td>
</tr>
<tr>
<td></td>
<td>Default option 2</td>
<td>The value of the default option is [ftp://grid-hg.ncsa.illinois.edu/ftp_scratch/ef/efac/20060905170000/2]</td>
</tr>
<tr>
<td></td>
<td>Default option 3</td>
<td>The value of the default option is [ftp://grid-hg.ncsa.illinois.edu/ftp_scratch/ef/efac/20060905170000/3]</td>
</tr>
<tr>
<td></td>
<td>Default option 4</td>
<td>The value of the default option is [ftp://grid-hg.ncsa.illinois.edu/ftp_scratch/ef/efac/20060905170000/4]</td>
</tr>
</tbody>
</table>

- **North American Model/CONUS 40 km**
  - **Form for ADAS (condut)**
    - **NCEP North American Model 40 km with forecast hours in separate files:** AWIPS 212 (R) Regional - CONUS - Double Resolution. Model runs are made at 00Z, 06Z, 12Z, and 18Z and have analysis and forecasts every 3 hours out to 84 hours. Horizontal = 185 by 129 points, resolution 40.63 km, LambertConformal projection: Vertical = surface, 1000 to 50 hPa pressure levels, layers, and depth.
    - **Form for ADAS (condut)**
    - **NCEP North American Model 40 km with forecast hours in separate files:** AWIPS 212 (R) Regional - CONUS - Double Resolution. Model runs are made at 00Z, 06Z, 12Z, and 18Z and have analysis and forecasts every 3 hours out to 84 hours. Horizontal = 185 by 129 points, resolution 40.63 km, LambertConformal projection: Vertical = surface, 1000 to 50 hPa pressure levels, layers, and depth.
  - **Form for ADAS (condut)**
    - **NCEP North American Model 40 km with forecast hours in separate files:** AWIPS 212 (R) Regional - CONUS - Double Resolution. Model runs are made at 00Z, 06Z, 12Z, and 18Z and have analysis and forecasts every 3 hours out to 84 hours. Horizontal = 185 by 129 points, resolution 40.63 km, LambertConformal projection: Vertical = surface, 1000 to 50 hPa pressure levels, layers, and depth.
    - **Form for ADAS (condut)**
    - **NCEP North American Model 40 km with forecast hours in separate files:** AWIPS 212 (R) Regional - CONUS - Double Resolution. Model runs are made at 00Z, 06Z, 12Z, and 18Z and have analysis and forecasts every 3 hours out to 84 hours. Horizontal = 185 by 129 points, resolution 40.63 km, LambertConformal projection: Vertical = surface, 1000 to 50 hPa pressure levels, layers, and depth.
  - **Form for ADAS (condut)**
    - **NCEP North American Model 40 km with forecast hours in separate files:** AWIPS 212 (R) Regional - CONUS - Double Resolution. Model runs are made at 00Z, 06Z, 12Z, and 18Z and have analysis and forecasts every 3 hours out to 84 hours. Horizontal = 185 by 129 points, resolution 40.63 km, LambertConformal projection: Vertical = surface, 1000 to 50 hPa pressure levels, layers, and depth.

**Timestamp:**
- 2006-11-07 11:00:22
- 2006-11-07 11:00:22
- 2006-11-07 11:00:22
- 2006-11-07 11:00:22
- 2006-11-07 11:00:22
- 2006-11-07 11:00:22
- 2006-11-07 11:00:22
- 2006-11-07 11:00:22
- 2006-11-07 11:00:22
- 2006-11-07 11:00:22
- 2006-11-07 11:00:22
- 2006-11-07 11:00:22
- 2006-11-07 11:00:22
- 2006-11-07 11:00:22
- 2006-11-07 11:00:22
- 2006-11-07 11:00:22
- 2006-11-07 11:00:22
- 2006-11-07 11:00:22
- 2006-11-07 11:00:22
- 2006-11-07 11:00:22
- 2006-11-07 11:00:22
- 2006-11-07 11:00:22
- 2006-11-07 11:00:22
- 2006-11-07 11:00:22
- 2006-11-07 11:00:22
- 2006-11-07 11:00:22
- 2006-11-07 11:00:22
- 2006-11-07 11:00:22
- 2006-11-07 11:00:22
- 2006-11-07 11:00:22
- 2006-11-07 11:00:22
- 2006-11-07 11:00:22
- 2006-11-07 11:00:22
- 2006-11-07 11:00:22
- 2006-11-07 11:00:22
Successfully created new experiment. You can monitor your experiment using the Workflow Composer.

Experiments

<table>
<thead>
<tr>
<th>Experiment Name</th>
<th>Description</th>
<th>Created On</th>
<th>Last Updated On</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>VGridADSWorkflow</td>
<td>No description</td>
<td>Wed Nov 08 00:35:29 EST 2005</td>
<td>Wed Nov 08 02:15:02 EST 2006</td>
<td>STARTED</td>
</tr>
<tr>
<td>MyFirstOne</td>
<td>No description</td>
<td>Wed Nov 08 01:20:04 EST 2005</td>
<td>Wed Nov 08 00:10:05 EST 2006</td>
<td>RUNNING</td>
</tr>
<tr>
<td>NAM Test</td>
<td>testing namin workflow</td>
<td>Wed Nov 08 01:32:52 EST 2005</td>
<td>Wed Nov 08 00:01:03 EST 2006</td>
<td>RUNNING</td>
</tr>
</tbody>
</table>
Scheduling with Batch Queues

• Last Year: VGrADS supported scheduling using estimated batch queue waiting times
  o Batch queue estimates are factored into communication time
    - E.g., the delay in moving from one resource to another is data movement time + estimated batch queue waiting time
  o Unfortunately, estimates can have large standard deviations

• This Year: limiting variability through two strategies:
  o Resource reservations: partially supported on the TeraGrid and other schedulers
  o In advance queue insertion: submit jobs before data arrives based on estimates
    - Can be used to simulate advance reservations

• Exploiting this requires a preliminary schedule indicating when the resources are needed
  o Problem: how to build an accurate schedule when exact resource types are unknown
Preliminary Scheduling Solution

• Use performance models to specify alternative resources
  o For step B, I need the equivalent of 200 Opterons, where 1 Opteron = 3 Itanium = 1.3 Power 5
    - Equivalence from performance model

• This permits an accurate preliminary schedule because the performance model standardizes the time for each step
  o Scheduling can then proceed with accurate estimates of when each resource collection will be needed
  o Makes advance reservations more accurate
    - Data will arrive neither too early or too late

• It may provide a mixture to meet the computational requirements, if the specification permits
  o Give me a loose bag of tight bags containing the equivalent of 200 Opterons, minimize the number of tight bags and the overall cost
    - Solution might be 150 Opterons in one cluster and 150 Itaniums in another