The Virtual Grid Application Development Software (VGrADS) Project

Ken Kennedy
Center for High Performance Software
Rice University

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

- VGrADS is an NSF-funded Information Technology Research project

![UCSB Logo] Rich Wolski

![UCSD Logo] Fran Berman
Andrew Chien
Henri Casanova

![RICE Logo] Keith Cooper
Ken Kennedy
Charles Koelbel
Richard Tapia
Linda Torczon

![University of Tennessee Logo] Jack Dongarra

![ISI Logo] Carl Kesselman

![University of North Carolina Logo] Dan Reed

![University of Houston Logo] Lennart Johnsson

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

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

• Where We Are
  — Low-level hand programming
  — Programmer needs to manage
    - Heterogeneous resources
    - Computation and data movement scheduling
    - Fault tolerance and performance adaptation

• What Do We Need?
  — A more abstract view of the Grid
    - Each developer sees a scalable “virtual grid”
  — Simplified programming models built on the abstract view
    - Permit the application developer to focus on the problem
Abstraction: Virtual Grid Execution System (vgES)

• 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

vgDL Description

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 is studying a range of tools for grid programming tasks, including:

- **Scheduling of workflow computations**
  - Off-line look-ahead scheduling dramatically improves in makespan (total time)
  - Accurate performance models significantly affect quality of scheduling
  - Queue wait prediction allows scheduling into batch queues

- **Fault tolerance**
  - Diskless checkpointing for linear algebra computations (application-specific)
  - Temporal reasoning for fault prediction
  - Optimal checkpoint frequency for iterative applications
VGrADS Application Collaborations

EMAN
Electron Micrograph Analysis

GridSAT
Boolean Satisfiability

BPEL Workflow Engine
Dynamic Workflow

GridFTP Service
Start

LDM Service
Data arrives

Resource Broker

WRF Service
Ensemble Broker

Data Mining
Visualization Service

LEAD
Atmospheric Science

Montage Astronomy

Figure 1: Example of conflict analysis with learning and non-chronological backtracking

SAT problem:
Decision stack before conflict:
Level 5: \(V_5\)
Level 6: \(V_6\)
Level 7: \(V_7\)
Level 8: \(V_8\)
Level 9: \(V_9\)
Level 10: \(V_{10}\)
Level 11: \(V_{11}\)
Level 12: \(V_{12}\)

Learned clause: \(\neg V_7 \lor V_8 \lor V_{11} \lor V_{12}\)

New decision stack after backtracking:
Level 5: \(V_5\)
Level 6: \(V_6\)
Level 7: \(V_7\)
Level 8: \(V_8\)
Level 9: \(V_9\)
Level 10: \(V_{10}\)
Level 11: \(V_{11}\)
Level 12: \(V_{12}\)

Node labels: assignee|decision level, ancestor

Learning and non-chronological backtracking

Virtual Grid Application Development Software Project
VGrADS Demos at SC|05

- **vgES / vgMON (UCSD)**
  - Runs EMAN application under vgES
  - Track and visualize progress with vgMON

- **Batch queue scheduling**
  - Schedules EMAN onto resources fronted by batch queues
  - Allows running across clusters

- **GridSolve**
  - Submits linear algebra problems for solution on the grid ala NetSolve
  - Uses vgES for
    - Integrated performance information
    - Integrated monitoring
    - Fault prediction
    - Integrating the software and resource information repositories