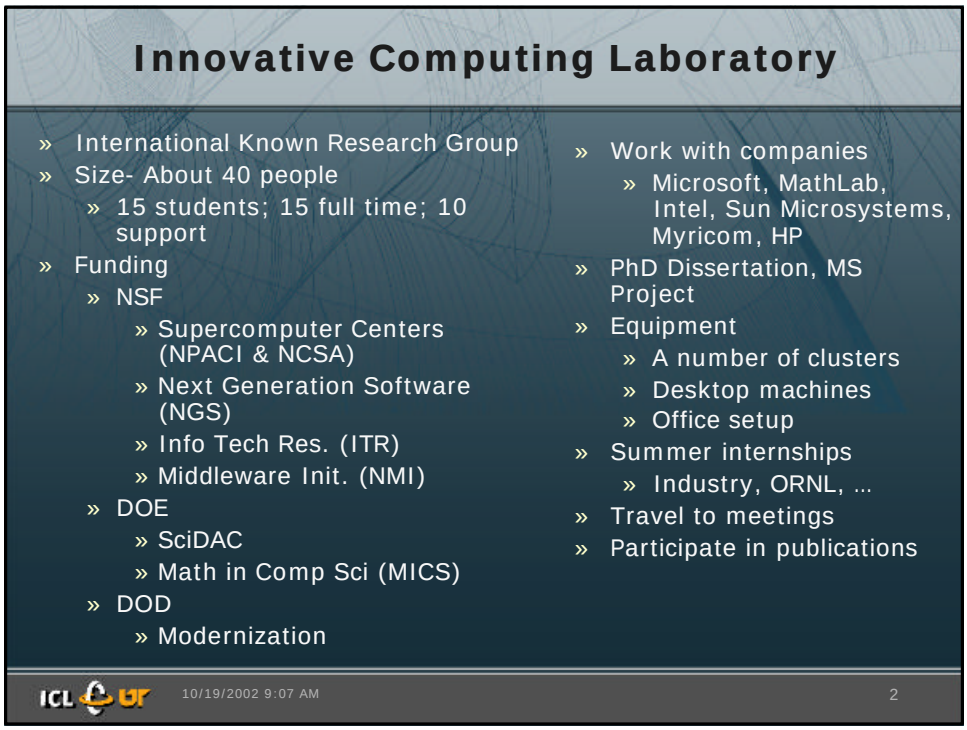

INNOVATIVE
COMPUTING
LABORATORY
Jack Dongarra
University of Tennessee
http://www.cs.utk.edu/~dongarra/
http://icl.cs.utk.edu/
ICL UT
Innovative Computing Laboratory
COMPUTER SCIENCE DEPARTMENT
UNIVERSITY OF TENNESSEE


Innovative Computing Laboratory
» International Known Research Group
» Size- About 40 people
» 15 students; 15 full time; 10 support
» Funding
» NSF
» Supercomputer Centers (NPACI & NCSA)
» Next Generation Software (NGS)
» Info Tech Res. (ITR)
» Middleware Init. (NMI)
» DOE
» SciDAC
» Math in Comp Sci (MICS)
» DOD
» Modernization
» Work with companies
» Microsoft, MathLab, Intel, Sun Microsystems, Myricom, HP
» PhD Dissertation, MS Project
» Equipment
» A number of clusters
» Desktop machines
» Office setup
» Summer internships
» Industry, ORNL, ...
» Travel to meetings
» Participate in publications
ICL UT
10/19/2002 9:07 AM
2

Four Thrust Research Areas

- Numerical Linear Algebra Algorithms and Software
  - EISPACK, LINPACK, BLAS, LAPACK, ScaLAPACK, PBLAS, Templates, ATLAS
  - Self Adapting Numerical Algorithms (SANS) Effort
    - LAPACK For Clusters
    - SALSA
- Heterogeneous Network Computing
  - PVM, MPI
  - FT-MPI, NetSolve
- Software Repositories
  - Netlib, NA-Digest
  - NHSE, RIB, NSDL
- Performance Evaluation
  - Linpack Benchmark, Top500, PAPI

ICL UT 10/19/2002 9:07 AM 3

Collaboration

- CS Department here at UTK
- Oak Ridge National Laboratory
- UC Berkeley/UC Davis
- UC Santa Barbara/UC San Diego
- Globus/ANL/ISI
- Salk Institute
- Danish Technical University/UNIC
- Monash University, Melbourne Australia
- Ecole Normal Superior, Lyon France
- ETHZ, Zurich Switzerland
- ETL, Tsukuba Japan
- Kasetsart U, Bangkok, Thailand

ICL UT 10/19/2002 9:07 AM 4

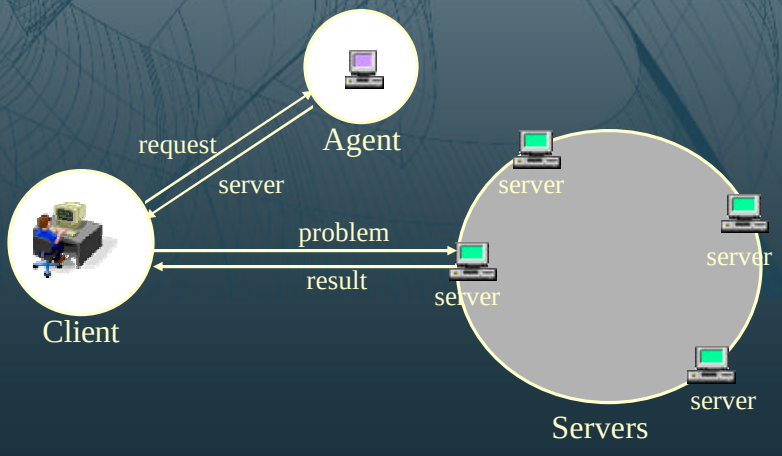

What Next?
» Jack -- Welcome
» Sudesh Agrawal-- NetSolve
» Kevin London -- PAPI
» Graham Fagg -- Harness/FT-MPI
» Asim YarKhan -- GrADS
» Victor Eijkhout-- SANS
ICL UT
10/19/2002 9:07 AM
5


NetSolve
Sudesh Agrawal
ICL UT
Innovative Computing Laboratory
COMPUTER SCIENCE DEPARTMENT
UNIVERSITY OF TENNESSEE

Introduction

- What is NetSolve
  - Is a research project started almost 6yrs back.
  - NetSolve is a client-server system that enables users to solve complex scientific problems over the net.
  - It allows users to access both hardware and software computational resources distributed across the net.

How Does NetSolve Work?



Usability

- Easy access to software
  - Access standard and/or custom libraries.
  - No need to know internal details about the implementation.
  - Simple interface or API to access these libraries and software
- Easy access to hardware
  - Access to machines registered with NetSolve system.
  - User's laptop can now access the power of super computers.
  - No need to worry about crashing user machine.
- User friendly interface to access the resources
  - C, Fortran interface
  - Matlab
  - Octave
  - Mathematica
  - Web

ICL UT 10/19/2002 9:07 AM 9

Features of NetSolve

- Asynchronous and Synchronous requests
- Sequencing
- Task Farming
- Fault Tolerance
- Dynamic addition and deletion of resources
- Pluggability with Condor-G
- Pluggability with NWS
- Pluggability with Globus
- Interface with IBP

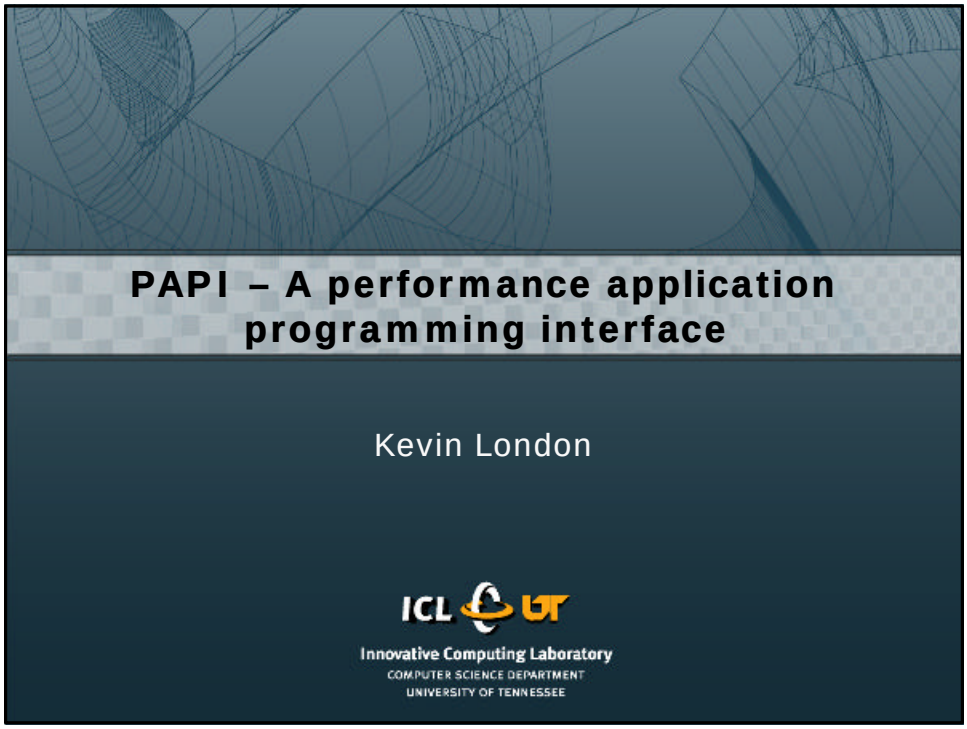
ICL UT 10/19/2002 9:07 AM 10

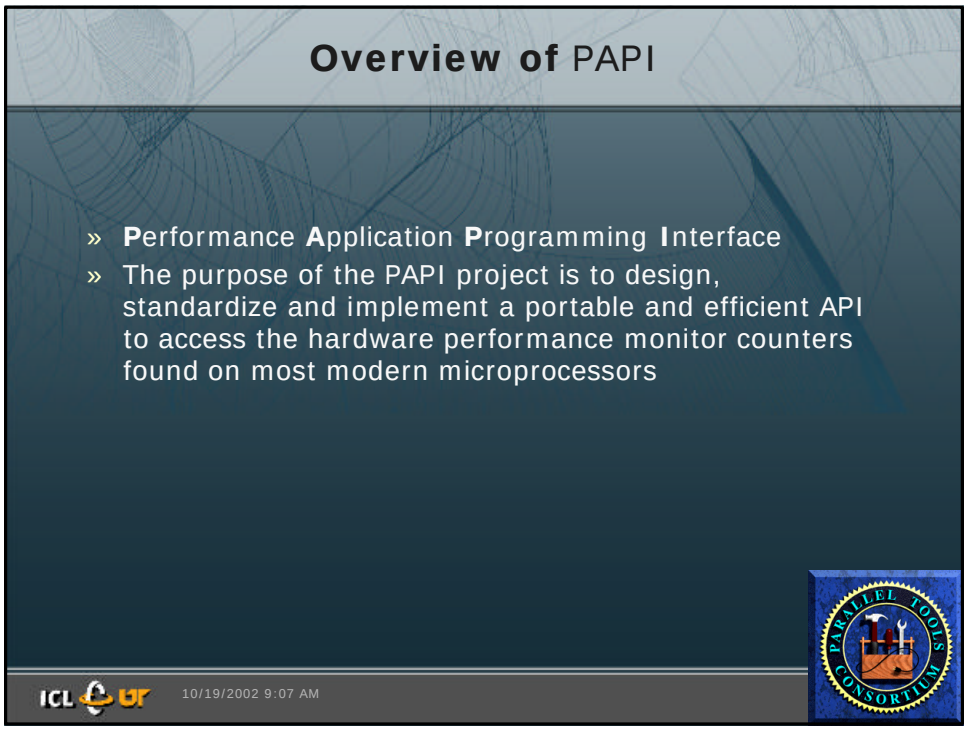
Future plans

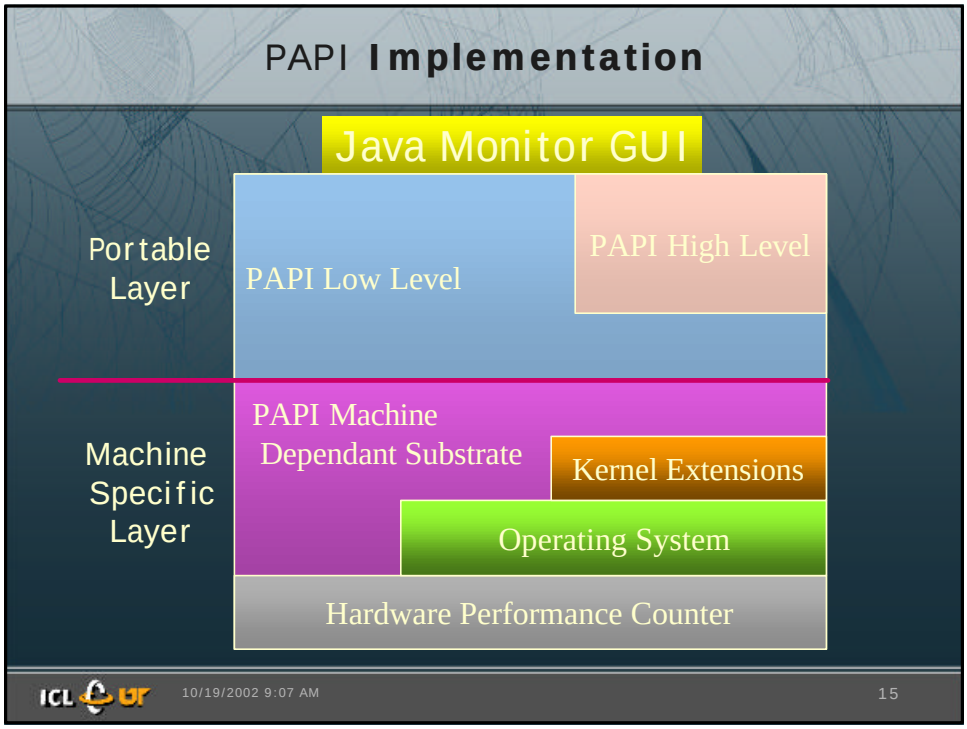
- NetSolve-E, which would be a revolutionary evolution of NetSolve.
- Client and Server can sit behind NATs and be able to talk to each other
- We would be able to incorporate different types of resources
- More dynamics would be added, to allow plug and play capability into the system.
- Resources would be able to come and go on the fly
- Many more.....
- In short, a revolution is going to happen in a year or two ☺
- For more information contact us at NetSolve@cs.utk.edu

Final Note

Thanks


PAPI – A performance application programming interface
Kevin London
ICL UT
Innovative Computing Laboratory
COMPUTER SCIENCE DEPARTMENT
UNIVERSITY OF TENNESSEE


Overview of PAPI
» Performance Application Programming Interface
» The purpose of the PAPI project is to design, standardize and implement a portable and efficient API to access the hardware performance monitor counters found on most modern microprocessors
ICL UT
10/19/2002 9:07 AM
PARALLEL TOOLS CONSORTIUM


PAPI Implementation
Java Monitor GUI
Portable Layer
PAPI Low Level
PAPI High Level
Machine Specific Layer
PAPI Machine Dependant Substrate
Kernel Extensions
Operating System
Hardware Performance Counter
ICL UT
10/19/2002 9:07 AM
15

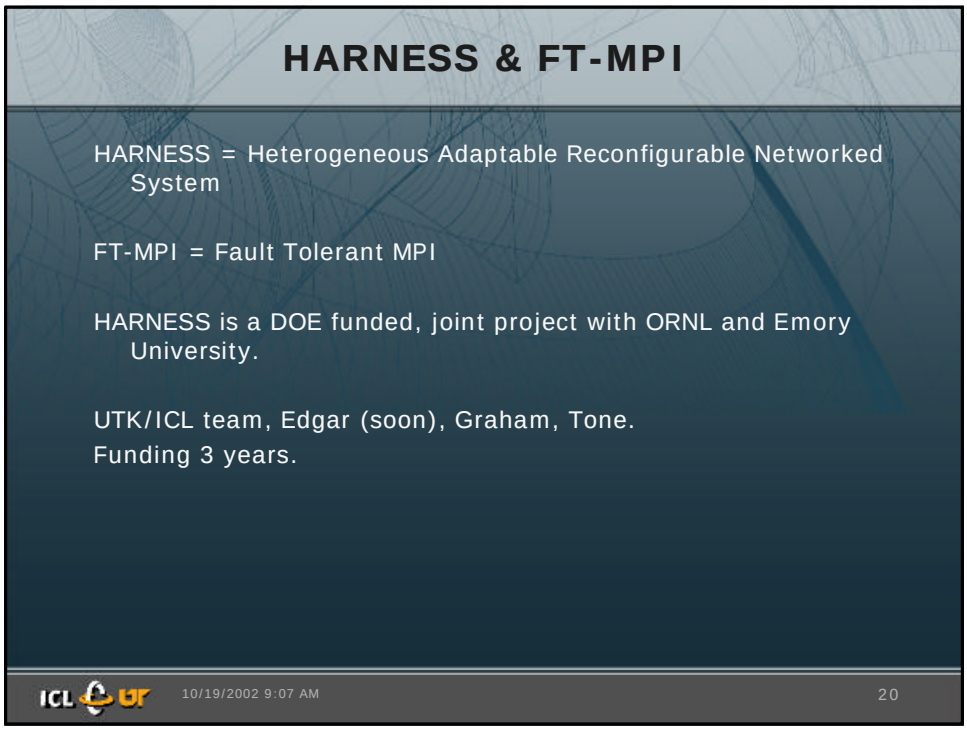

PAPI Staff
Current Staff Members
» Jack Dongarra
» Kevin London
» Philip Mucci
» Shirley Moore
» Keith Seymour
» Dan Terpstra
» Haihang You
» Min Zhou
Former Staff Members
» Qichao Dong
» Cricket Deane
» Nathan Garner
» George Ho
» Leelinda Parker
» Thomas Spencer
» Long Zhou
ICL UT
10/19/2002 9:07 AM
16

PAPI Users

Tools currently using PAPI

- Deep/MPI
- Scalea
- SvPablo
- TAU
- Vprof


HARNESS & FT-MPI
Graham Fagg
320 Claxton
fagg@cs.utk.edu
http://icl.cs.utk.edu/harness
ICL UT
Innovative Computing Laboratory
COMPUTER SCIENCE DEPARTMENT
UNIVERSITY OF TENNESSEE


HARNESS & FT-MPI
HARNESS = Heterogeneous Adaptable Reconfigurable Networked System
FT-MPI = Fault Tolerant MPI
HARNESS is a DOE funded, joint project with ORNL and Emory University.
UTK/ICL team, Edgar (soon), Graham, Tone.
Funding 3 years.
ICL UT
10/19/2002 9:07 AM
20

Whats HARNESS?

- Once upon a time.. We built s/w in a big block of modules. Each module did a different thing.. But they all got linked into a single executable.
  - Example PVM a message passing library.
- So when we needed some new functionality we wrote the new code, and recompiled a new executable.

Whats HARNESS?

- HARNESS is a back-plane/skeleton
- Build parts as you need them, put them on a web repository or in a local directory.
- When you need something load them dynamically and then maybe throw them away...
- Think of kernel modules but for a distributed system that does parallel RPC and message passing.
- NOT JAVA, its faster C, C++, F90 etc

Whats FT-MPI

- MPI is the Message Passing Interface standard.
- FT-MPI is an implementation of that.
- But..
  - MPI programs were designed to live on reliable supercomputers.
  - Modern machines and clusters are made from many thousands of commodity CPUs.
  - MTBFtotal = MTBFnode * number of nodes
    - MTBFtotal < my large application simulating the weather
- In English, modern jobs on modern machines have a high chance of failure and as they get bigger it will just get worse…

What is FT-MPI

- FT-MPI extends MPI and allows applications to decide what to do when an error occurs:
  - restarting a failed node
  - continuing with a lesser number of nodes
- Other MPI implementations either just abort everything OR they use check-pointing to “roll back” which is expensive.

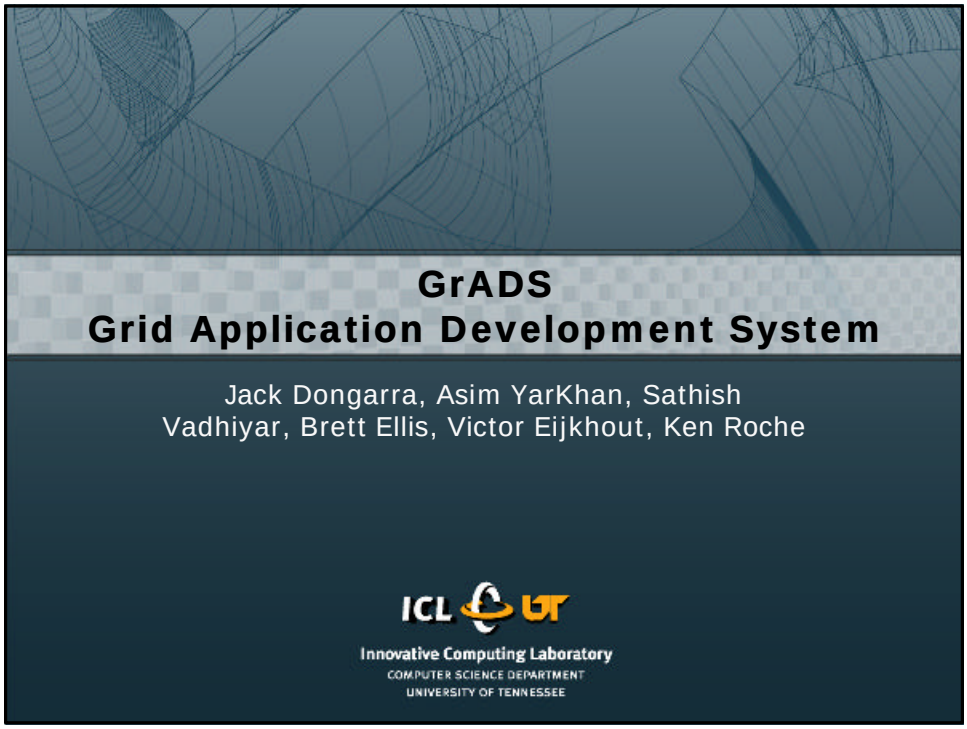
Research stuff

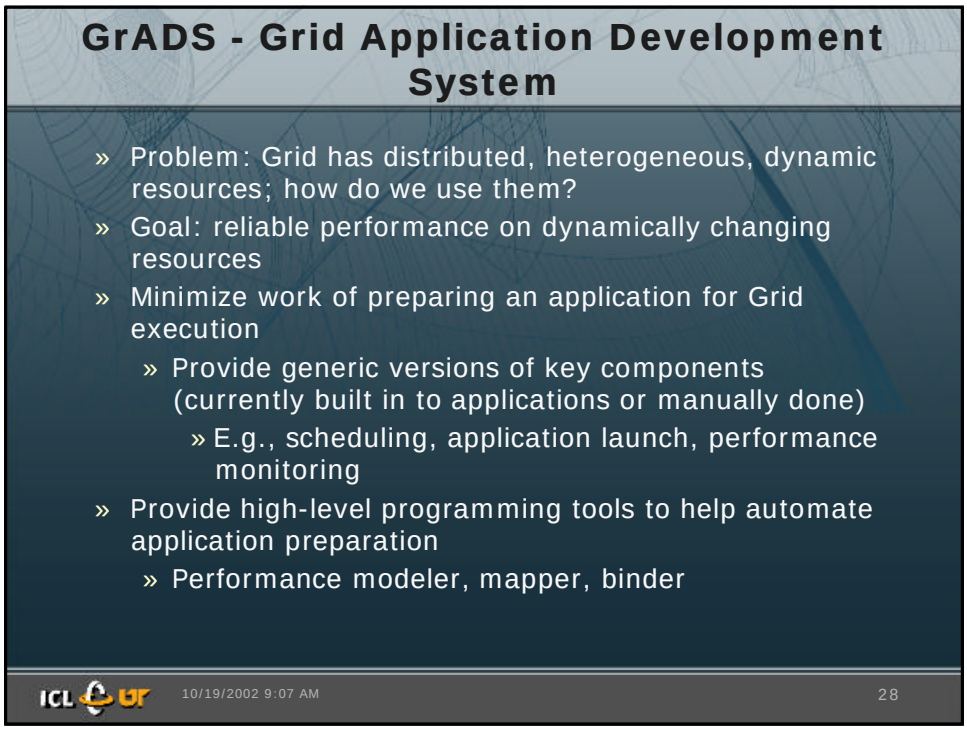
- HARNESS
  - Distributed algorithms for coherency
  - Management of plug-ins
  - High speed parallel RPCs
- FT-MPI
  - Many2many [collective/group] communications, buffer management, new algorithms of numeric libraries
  - Fault state management
- Skills you would use:
  - networking (TCP/sockets), systems (threads/posix calls)

Contact info:

Graham Fagg
320 Claxton
Phone 974-5790

Email: fagg@cs.utk.edu
Web: http://icl.cs.utk.edu/harness


GrADS
Grid Application Development System
Jack Dongarra, Asim YarKhan, Sathish Vadhiyar, Brett Ellis, Victor Eijkhout, Ken Roche
ICL UT
Innovative Computing Laboratory
COMPUTER SCIENCE DEPARTMENT
UNIVERSITY OF TENNESSEE

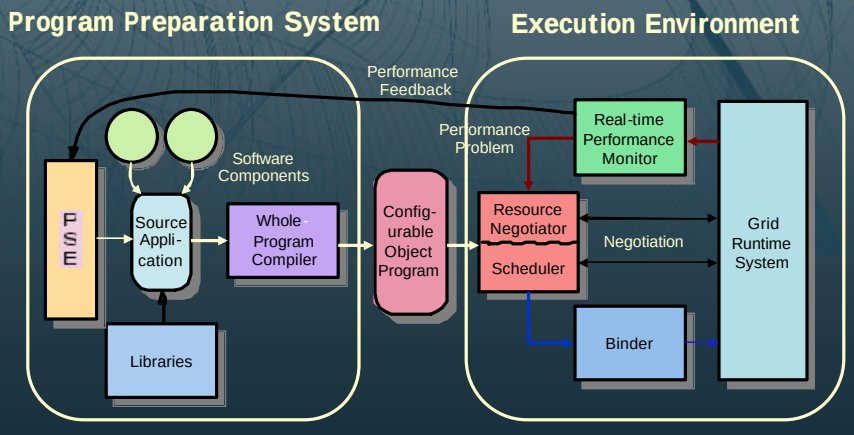

GrADS - Grid Application Development System
» Problem: Grid has distributed, heterogeneous, dynamic resources; how do we use them?
» Goal: reliable performance on dynamically changing resources
» Minimize work of preparing an application for Grid execution
» Provide generic versions of key components (currently built in to applications or manually done)
» E.g., scheduling, application launch, performance monitoring
» Provide high-level programming tools to help automate application preparation
» Performance modeler, mapper, binder
ICL UT
10/19/2002 9:07 AM
28

People in GrADS

- Principal Investigators
  - Francine Berman, UCSD
  - Andrew Chien, UCSD
  - Keith Cooper, Rice
  - Jack Dongarra, Tennessee
  - Ian Foster, Chicago
  - Dennis Gannon, Indiana
  - Lennart Johnsson, Houston
  - Ken Kennedy, Rice
  - Carl Kesselman, USC ISI
  - John Mellor-Crummey, Rice
  - Dan Reed, UIUC
  - Linda Torczon, Rice
  - Rich Wolski, UCSB
- Other Contributors
  - Dave Angulo, Chicago
  - Henri Casanova, UCSD
  - Holly Dail, UCSD
  - Anshu Dasgupta, Rice
  - Sridhar Gullapalli, USC ISI
  - Charles Koelbel, Rice
  - Anirban Mandal, Rice
  - Gabriel Marin, Rice
  - Mark Mazina, Rice
  - Celso Mendes, UIUC
  - Otto Sievert, UCSD
  - Martin Swany, UCSB
  - Satish Vadhiyar, Tennessee
  - Asim YarKhan, Tennessee

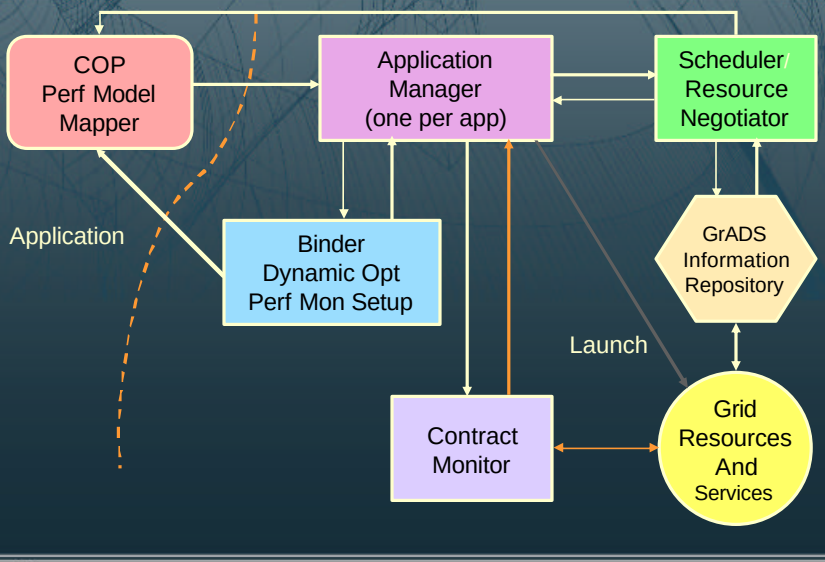
ICL UT 10/19/2002 9:07 AM 29

GrADSoft Architecture



ICL UT 10/19/2002 9:07 AM 30

GrADS Program Execution System



Research Areas

- Automatically generating performance models (e.g. for ScaLAPACK) on Grid resources
- Evaluating Performance “Contracts”
- Near Optimal Scheduling (execution) on the Grid
- Rescheduling for changing resources
- Checkpointing and fault tolerance
- High-latency tolerant algorithms (SANS ideas)
- Porting applications/libraries to GrADS framework
- Developing generic GrADSoft interfaces (API’s)

How To Be A Mathematician In A CS Department And Still Have Fun

Victor Eijkhout

eijkhout@cs.utk.edu

We get $D_{jj} \downarrow d_0$, $G_{i+j,j} \uparrow g_i$, $H_{j,k+i} \uparrow h_i$ where d_0, g_i, h_i satisfy the limit equations

$$d_0 = x_0 - \sum_{k=1}^{\min(p_-,p_+)} \frac{g_k h_k}{d_0}, \tag{3a}$$

$$g_i = x_{-i} + \sum_{k=1}^{p_- - i} \frac{g_{i+k} h_k}{d_0}, \tag{3b}$$

$$h_i = x_i + \sum_{k=1}^{p_+ - i} \frac{g_k h_{i+k}}{d_0}. \tag{3c}$$

Now let $p = \max(p_-, p_+)$, let $g_i = 0$ for $p_- < i \le p$ and $h_i = 0$ for $p_+ < i \le p$. Then

$$\begin{aligned}
&d_0 \left(x_0 - \sum_{i=1}^{p_-} x_{-i} - \sum_{i=1}^{p_+} x_i \right) \\
&= d_0^2 + \sum_{k=1}^{\min(p_-,p_+)} g_k h_k - d_0 \sum_{i=1}^{p_-} g_i + \sum_{i=1}^{p_-} \sum_{k=1}^{p_- - i} g_{i+k} h_k - d_0 \sum_{i=1}^{p_+} h_i + \sum_{i=1}^{p_+} \sum_{k=1}^{p_+ - i} g_k h_{i+k} \\
&= d_0^2 + \sum_{k=1}^{p} g_k h_k - d_0 \sum_{i=1}^{p_-} g_i + \sum_{k=1}^{p_- - 1} \sum_{i=1}^{k-1} g_k h_i - d_0 \sum_{i=1}^{p_+} h_i + \sum_{k=1}^{p_+ - 1} \sum_{i=k+1}^{p_+} g_k h_i \\
&= d_0^2 - d_0 \sum_{i=1}^{p_-} g_i - d_0 \sum_{i=1}^{p_+} h_i + \sum_{k=1}^{p_-} \sum_{i=1}^{p_+} g_k h_i = \left(d_0 - \sum_{i=1}^{p_-} g_i \right) \left(d_0 - \sum_{i=1}^{p_+} h_i \right) \\
&= \left(d_0 - \sum_{i=1}^{p} g_i \right) \left(d_0 - \sum_{i=1}^{p} h_i \right)
\end{aligned}$$

gives that

$$\begin{aligned}
\left(x_0 - \sum_{i=1}^{p_-} x_{-i} - \sum_{i=1}^{p_+} x_i \right) &= \frac{1}{d_0} \left(d_0 - \sum_{i=1}^{p_-} g_i \right) \left(d_0 - \sum_{i=1}^{p_+} h_i \right) \\
&= \left(\sqrt{d_0} - \sum_{i=1}^{p_-} \frac{g_i}{\sqrt{d_0}} \right) \left(\sqrt{d_0} - \sum_{i=1}^{p_+} \frac{h_i}{\sqrt{d_0}} \right).
\end{aligned} \tag{4}$$

If x_0 is increased by a small amount, the lhs (which is positive) increases. From the recursion formulae it follows that d_0 will also increase, whereas all g_i and h_i will decrease. Hence both factors of the rhs will increase; as their product increases and they are of equal sign they must both be positive. Qed

2.2 Some elementary estimates for Toeplitz matrices

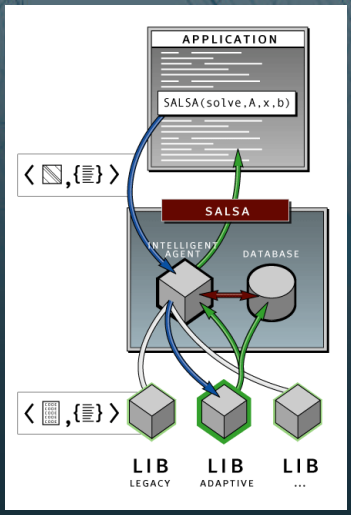
In the case $X = (x_{i-j})$ the coefficients introduced in (1a,b,c) are readily estimated. From (3a) we estimate d_0:

$$d_0 = x_0 - \sum \frac{g_k h_k}{d_0} \ge x_0 - \frac{(\sum g_k)(\sum h_k)}{d_0} \ge x_0 - d_0$$

2 Decay rates of inverses 21

The SALSA Project

- Self-Adaptive Linear Solver Architecture
- Traditional approach: user picks library routine,
- calls.
  - All decisions up to user
- Need for intelligent middleware to assist the user in
- picking the best library call
  - One extreme: use as black box
  - Less extreme: the user supplies hints, wishes, annotations
- Intelligence is developed over time: feedback of results into a database
  - Tuning of heuristics.



ICL UT 10/19/2002 9:07 AM 35

To Contact Us:

- Send email to dongarra@cs.utk.edu
- http://icl.cs.utk.edu/

ICL UT 10/19/2002 9:07 AM 36