Interactive Supercomputing's Star-P Platform: Parallel MATLAB & MPI Classroom Study

Alan Edelman
MIT and ISC

Parry Husbands
LBNL and ISC

Steve Leibman
MIT and ISC

INTER*CTIVE supercomputing

Company

- Background:
  - Founded in 2004, venture-backed
  - M.I.T. spin-off
  - Exclusive technology license
  - Parallel Computing Harder than most realize:
  - Technology: Star-P software platform supporting automatic parallelization and interactive execution of desktop technical applications on parallel servers
  - Not just a parallel MATLAB
- Market:
  - Value prop: reduction in time-to-solution for large and complex problems
  - Can plug in existing parallel and serial software seamlessly



INTERACTIVE supercomputing

Star-P™ Enables Easy Parallel Computing on Multi-core Servers and Clusters.

Today, with MATLAB® environment.

INTER*CTIVE supercomputing

Easiest Parallel use of MATLAB

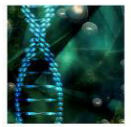
- Run MATLAB on each machine

The Parallel MATLABS (no one such beast)

multiMATLAB
Cornell Multitasking Toolbox
DP-Toolbox
MPITB/PVMTB
MATmarks
MatlabMPI
pMatlab
MULTI Toolbox
Paralize
PMI
PLab
Parmatlab
DistributePP

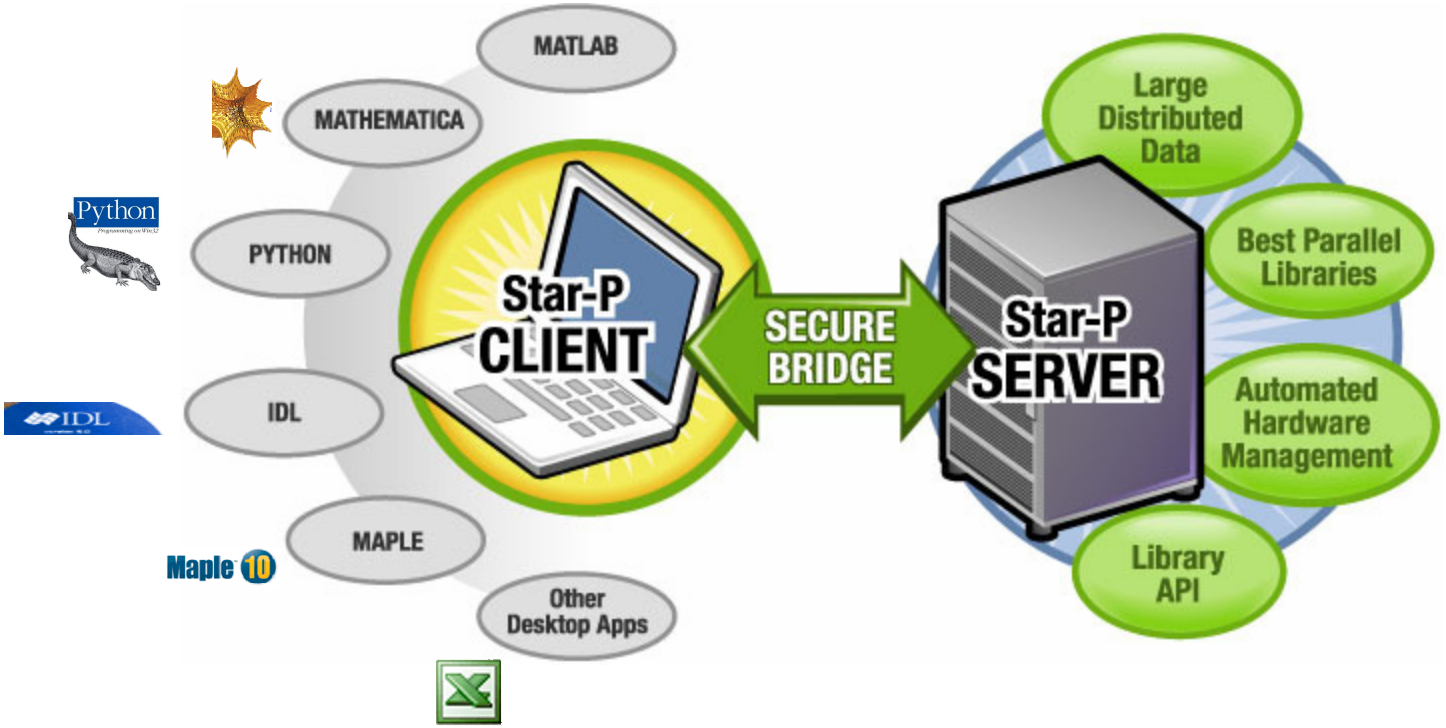
Netsolve
DLab
Matpar
PLAPACK
Paramat
Otter
RTExpress
ParAL
FALCON
CONLAB
MATCH
Menhir
MATHWORKS MATLAB
MATHWORKS Distributed Computing Toolbox
MATHWORKS Cleve Moler's Vision

Star-P with the MATLAB client environment!


INTERACTIVE supercomputing

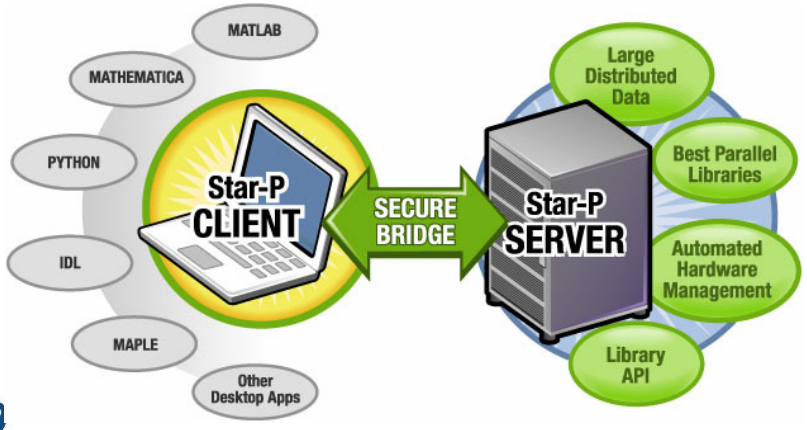
The Client (a math lab) is the browser!



INTERACTIVE supercomputing

Client-Server Parallel Computing

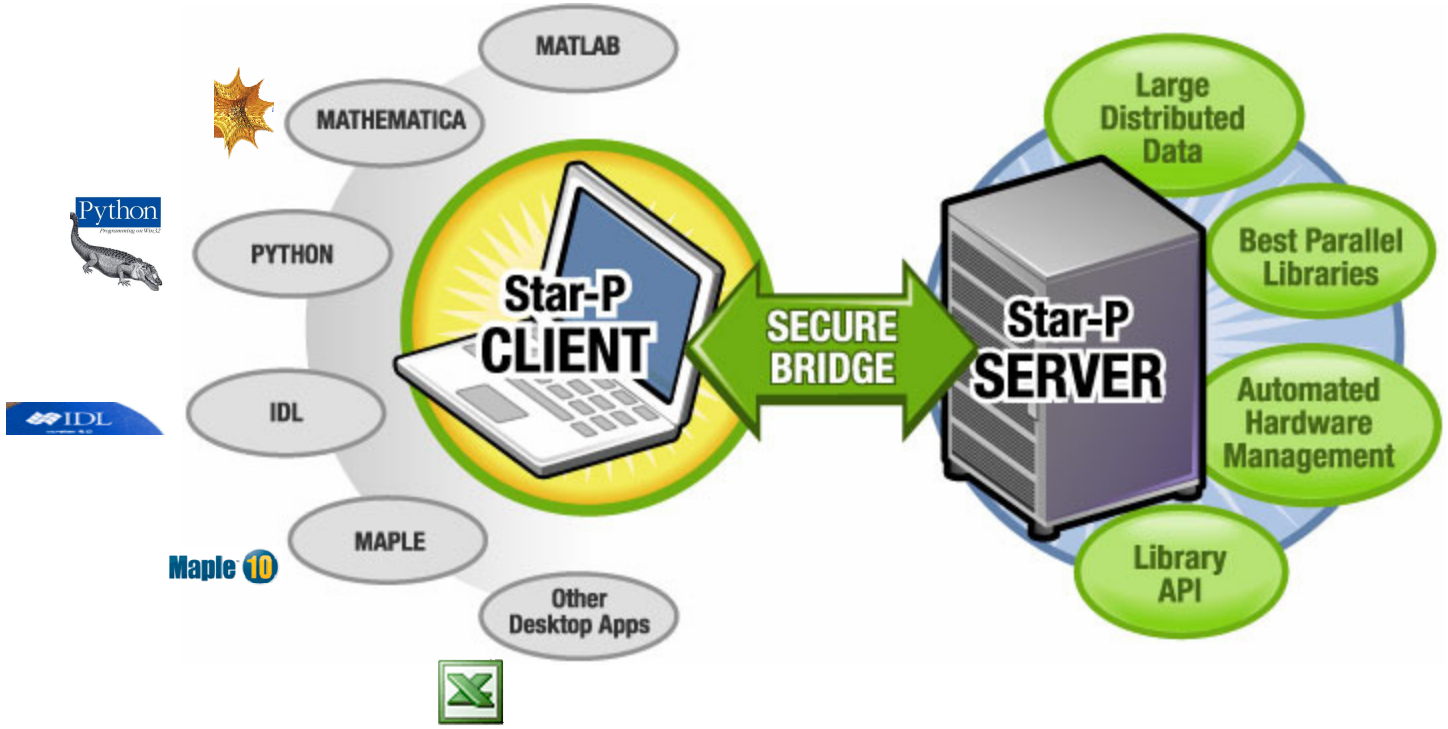
- Your bank & financial data
- Your email
- Your travel
- Your photos
- 2006: MIT students hw grades
- Your parallel computing



INTERACTIVE supercomputing

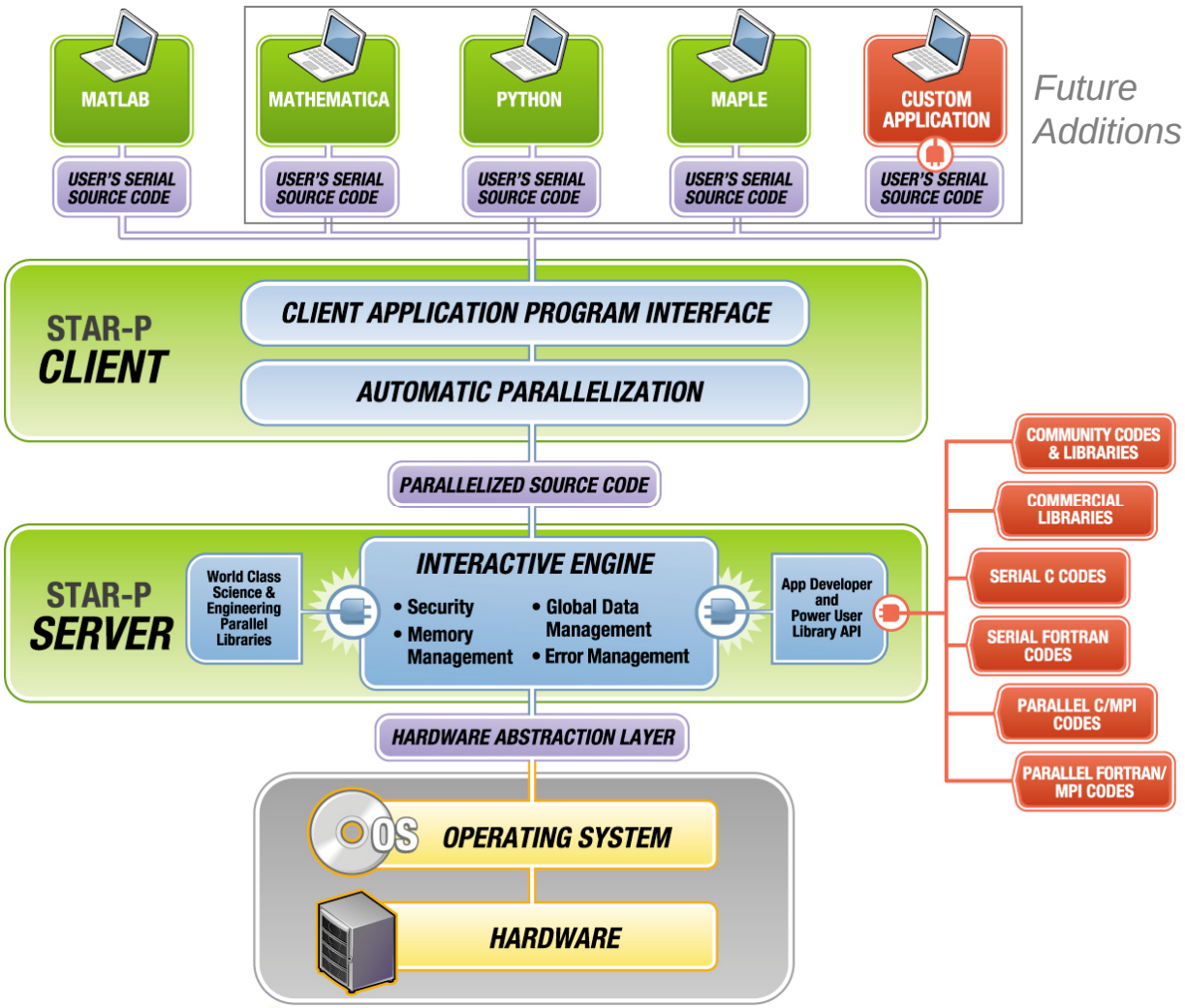
Client-Server Parallel Computing

Platform for automatic parallelization
and interactive execution of desktop apps
on HPCs



INTERACTIVE supercomputing

The Key to Star-P™ Value: Architecture



INTERACTIVE supercomputing

Client-Server Software

- Client interacts with HLL environments
- Distributed server for SMP and cluster systems

Computing Modes

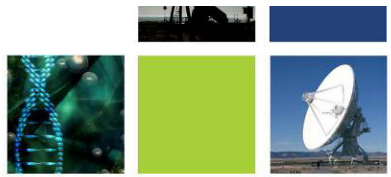
- Serial & parallel computing
- Data- and Task-parallel
- Extensions via API/SDK

Ease of Use

- Simple Star-P commands

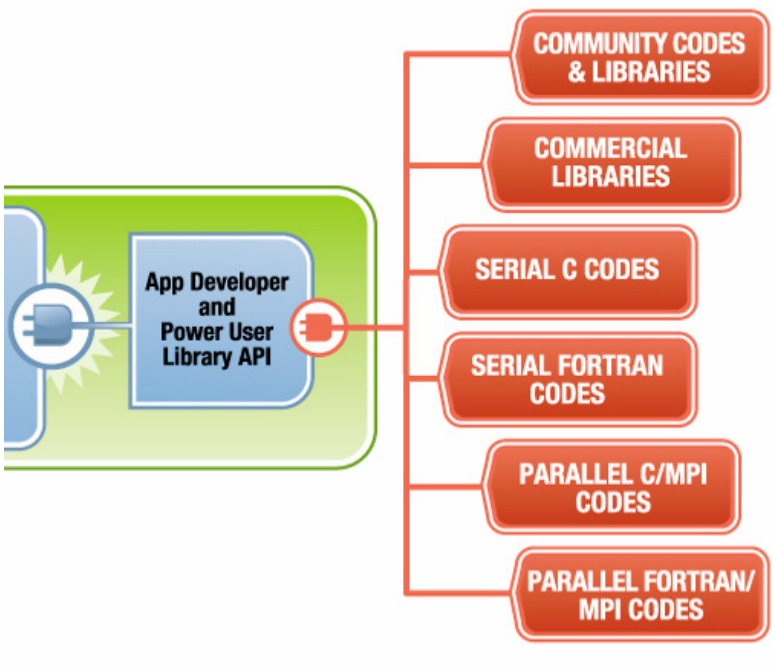
Software Platform

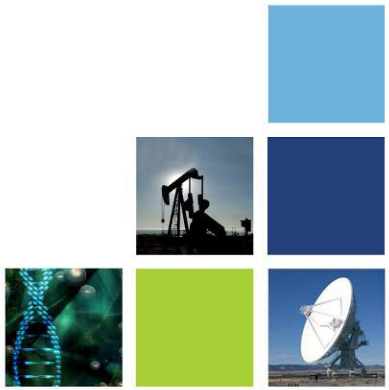
- Multiple HLLs and applications *(future)*



Plug into Star-P through Server API

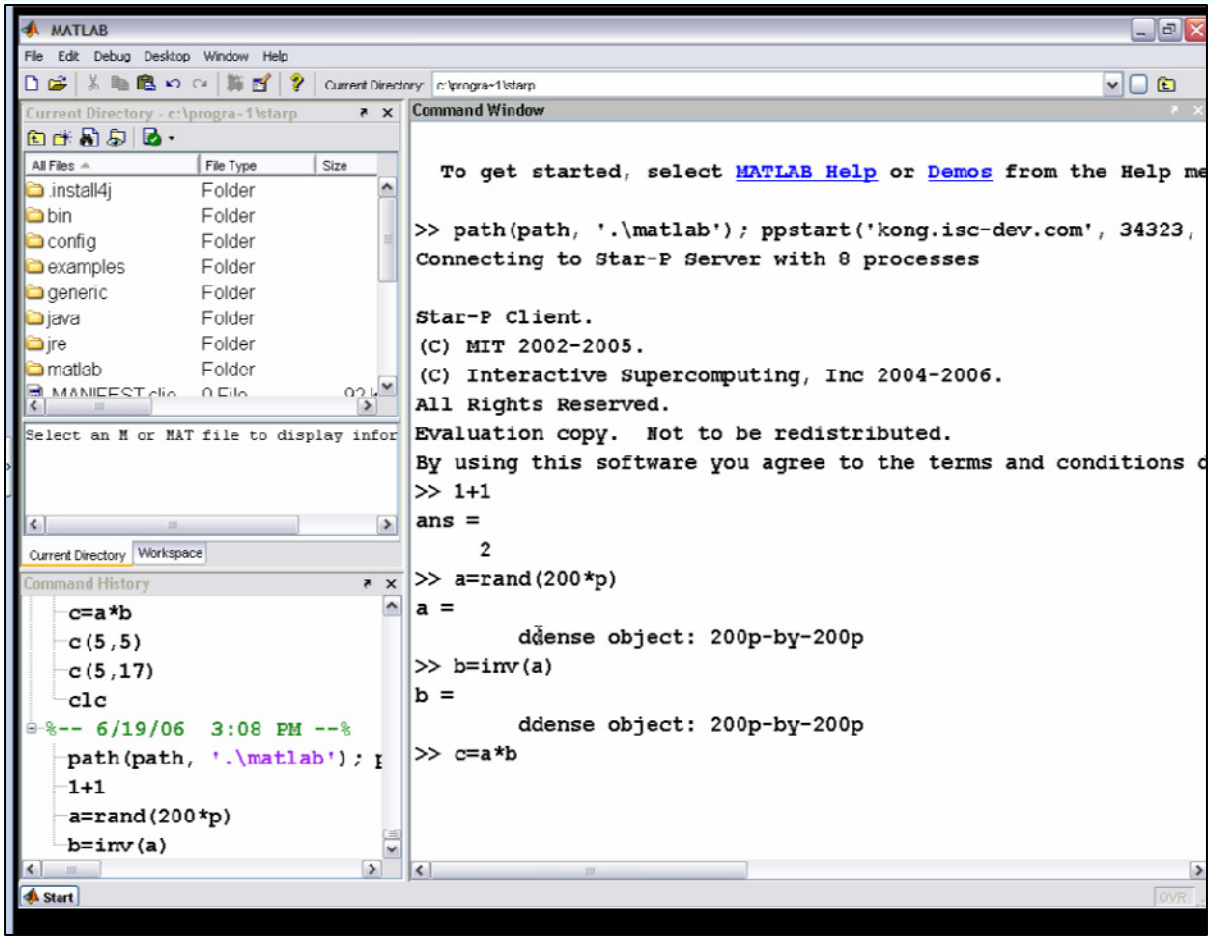
- Through MATLAB, access:
  - Your own library functions
  - Specialized hardware (FPGA's)
- Serial and parallel codes
  - Coarse-grained "multiply effect"
  - Parallel codes
- Started in MPI?
  - Not too late. Just plug it in and keep moving forward. Access from MATLAB
- Have an old serial fortran code?
  - Run it with multiple paramaters on different processors. Access from MATLAB

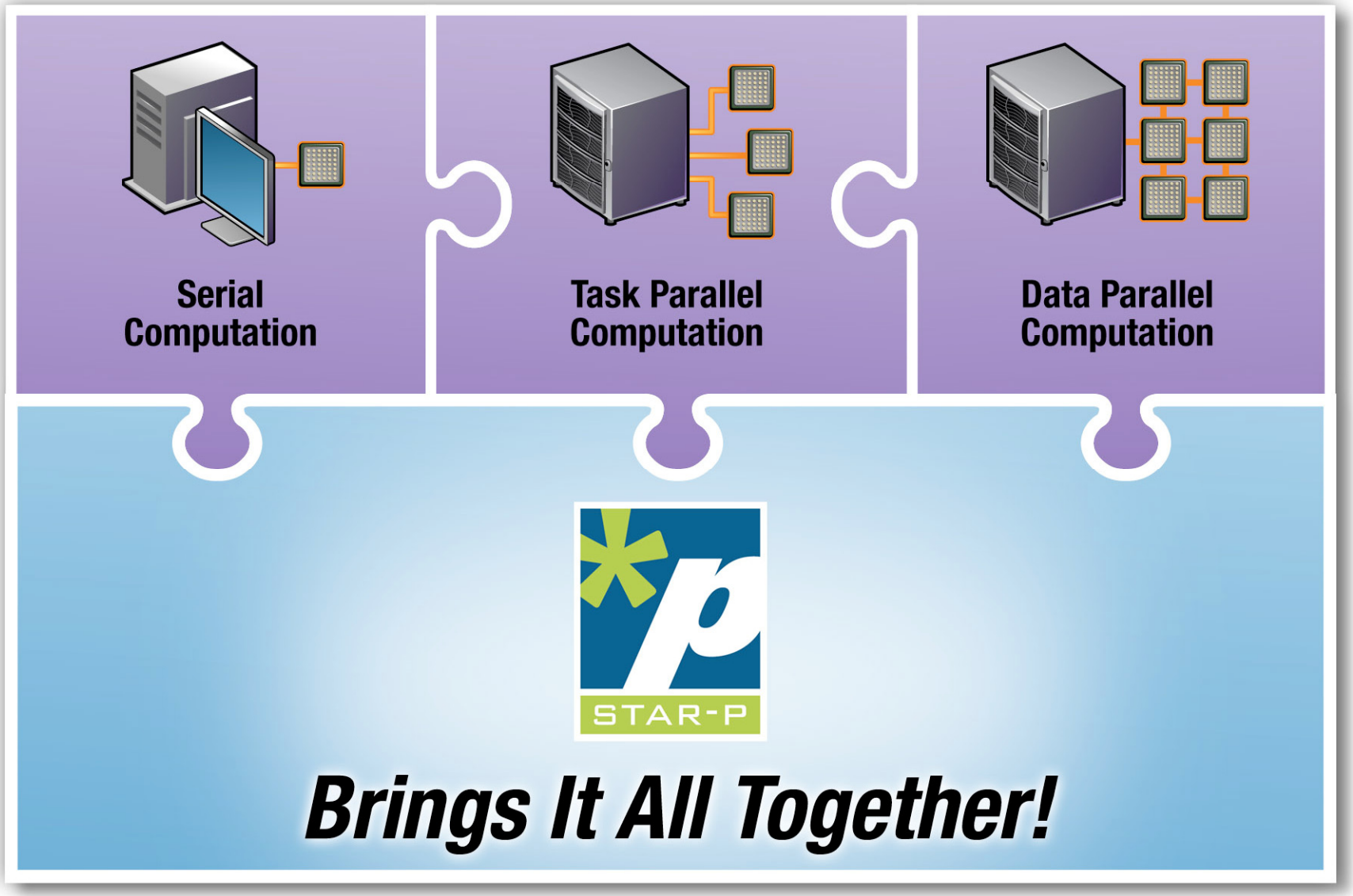




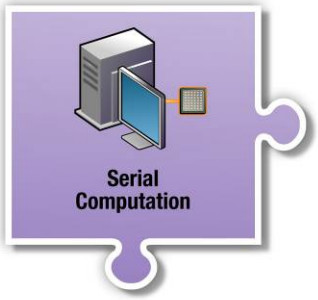
INTERACTIVE supercomputing

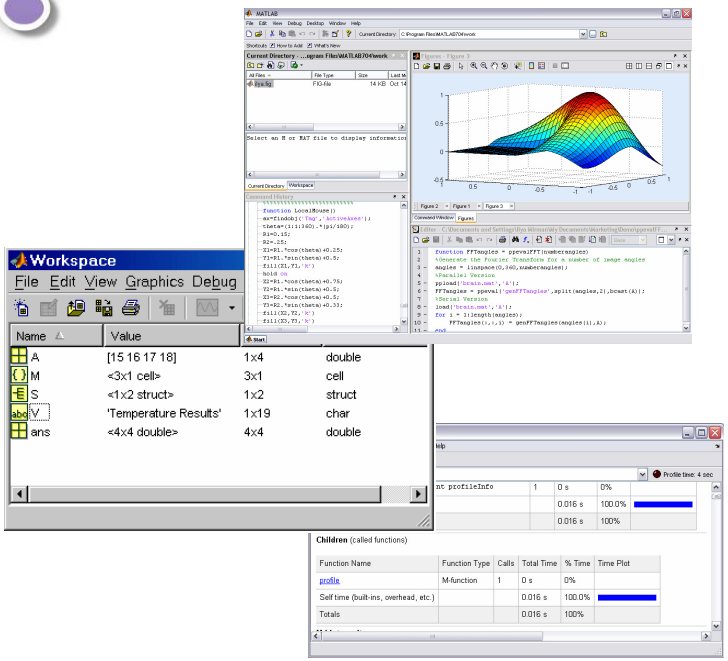
Video


MATLAB
File Edit Debug Desktop Window Help
Current Directory: c:\progra~1\starp
Current Directory - c:\progra~1\starp
All Files
File Type
Size
.install4j Folder
bin Folder
config Folder
examples Folder
generic Folder
java Folder
jre Folder
matlab Folder
Select an M or MAT file to display infor
Current Directory
Workspace
Command History
c=a*b
c(5,5)
c(5,17)
clc
%-- 6/19/06 3:08 PM --%
path(path, '.\matlab'); p
1+1
a=rand(200*p)
b=inv(a)
Start
Command Window
To get started, select MATLAB Help or Demos from the Help me
>> path(path, '.\matlab'); ppstart('kong.isc-dev.com', 34323,
Connecting to Star-P Server with 8 processes
Star-P Client.
(C) MIT 2002-2005.
(C) Interactive Supercomputing, Inc 2004-2006.
All Rights Reserved.
Evaluation copy. Not to be redistributed.
By using this software you agree to the terms and conditions o
>> 1+1
ans =
2
>> a=rand(200*p)
a =
ddense object: 200p-by-200p
>> b=inv(a)
b =
ddense object: 200p-by-200p
>> c=a*b


Serial
Computation
Task Parallel
Computation
Data Parallel
Computation
STAR-P
Brings It All Together!

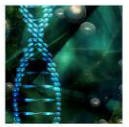
Serial Computing in Star-P™





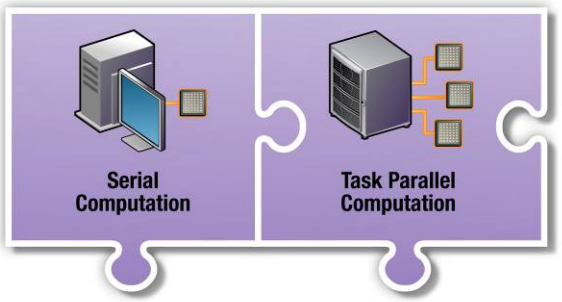
- Use MATLAB
  - File Editor
  - Profiler
  - Debugger
  - Array Editor
  - Desktop
  - Visualization
  - Small Calculations
- Computations taking less than .5 seconds



INTERACTIVE supercomputing

Task Parallel Computing in Star-P™

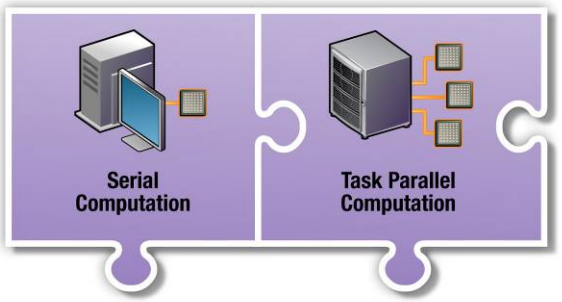


- Data size < 100MB
- Execution time > .5 second
- Code separable in time
- Embarrassingly parallel apps
- Incorporate Star-P's **ppeval**

```
1  %Generate the Fourier Transform on 10 degree spacing
2  angles = linspace(0,360,37);
3  %Serial Version
4  load('brain.mat','A');
5  for i = 1:length(angles);
6      FFTangles(:,:,i) = genFFTangles(angles(i),A);
7  end
8
9
10
```

INTERACTIVE supercomputing

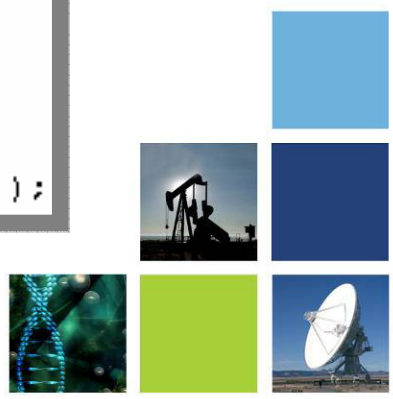
Task Parallel Computing in Star-P™



- Data size < 100MB
- Execution time > .5 second
- Code separable in time
- Embarrassingly parallel apps
- Incorporate Star-P's **ppeval**

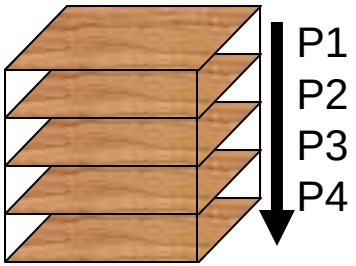
```
1  %Generate the Fourier Transform on 10 degree spacing
2  angles = linspace(0,360,37);
3  %Serial Version
4  load('brain.mat','A');
5  for i = 1:length(angles);
6      FFTangles(:,:,i) = genFFTangles(angles(i),A);
7  end
8  %Parallel Version
9  ppload('brain.mat','A');
10 FFTangles = ppeval('genFFTangles',split(angles),bcast(A));
```

INTERACTIVE supercomputing

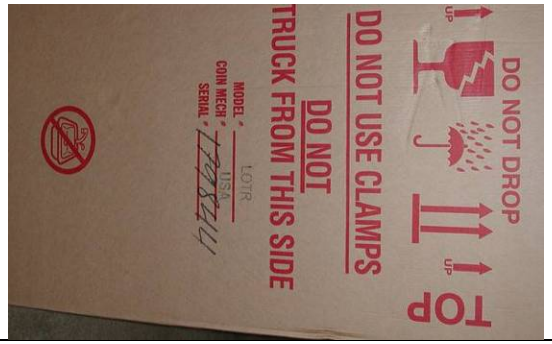


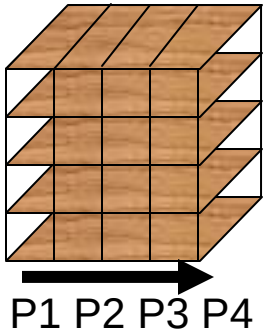
ppeval syntax (parallel function)

- a=rand(500,500,200*p);
- [u,s,v]=ppeval('svd',a); % default svd on z-dim



- a=rand(500,500*p,200);
- [u,s,v]=ppeval('svd',a); % default svd on z-dim anyway

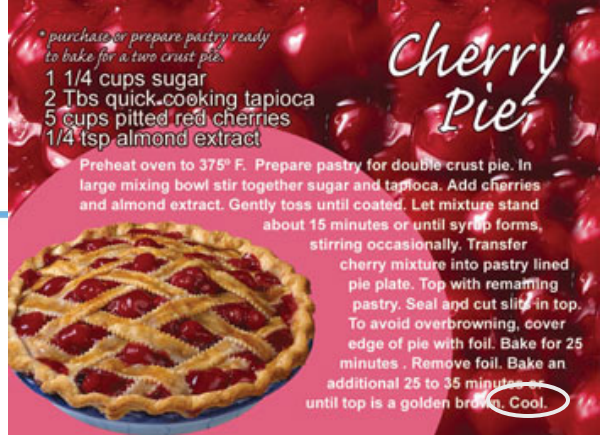




Answer does not depend on distribution:

Parallel computers need shapes to enter from all sides.

Pi Recipe

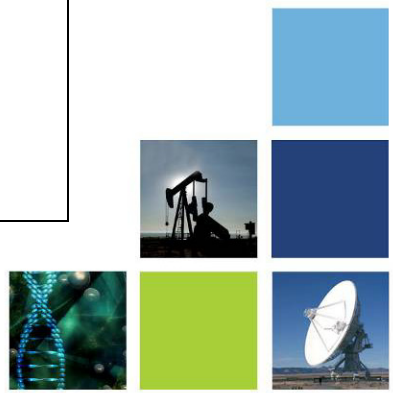



```
>> n=8;   k=1:n;
>> sum(ppeval('quad','4./(1+x.^2)', (k-1)/n, k/n))
```

Parallel Evaluate Pieces of pi:
$\int 4/(1+x^2)\,dx$ *on* $[0,1/8],[1/8,2/8],\ldots,[7/8,1]$ *and sum.*

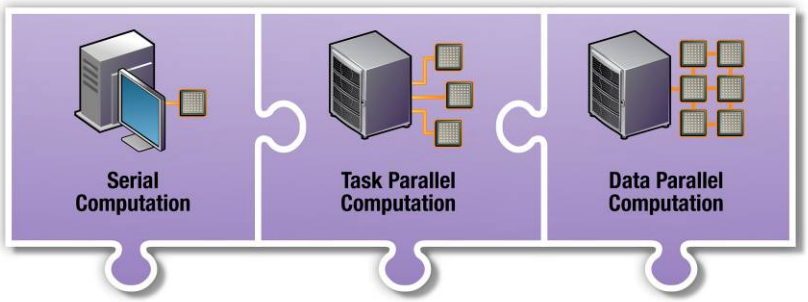
```
ans =
   3.14159265358979
```

Abstraction: Independent of number of processors or processes!
Abstraction: Parameters automatically moved to server!



INTERACTIVE supercomputing

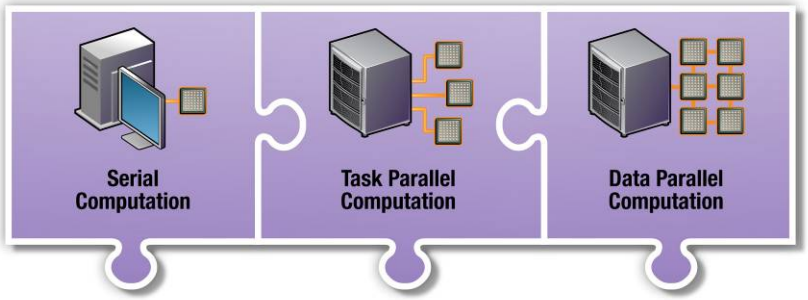
Data Parallel Computing in Star-P™




```
n=10000

A = rand(n, n);

x = randn(n, 1);

y = zeros(size(x));

while norm(x-y) / norm(x) > 1e-11
    y = x;
    x = A*x;
    x = x / norm(x);
end;
```

- Data sizes >100MB
- Execution time > .5 second
- Data not separable
- Operations on vectors and matrices
- Incorporate ***p**
  - Global parallelism
  - Variables become parallel
  - Propagation occurs
    - Results are parallel
    - Functions performed on parallel data

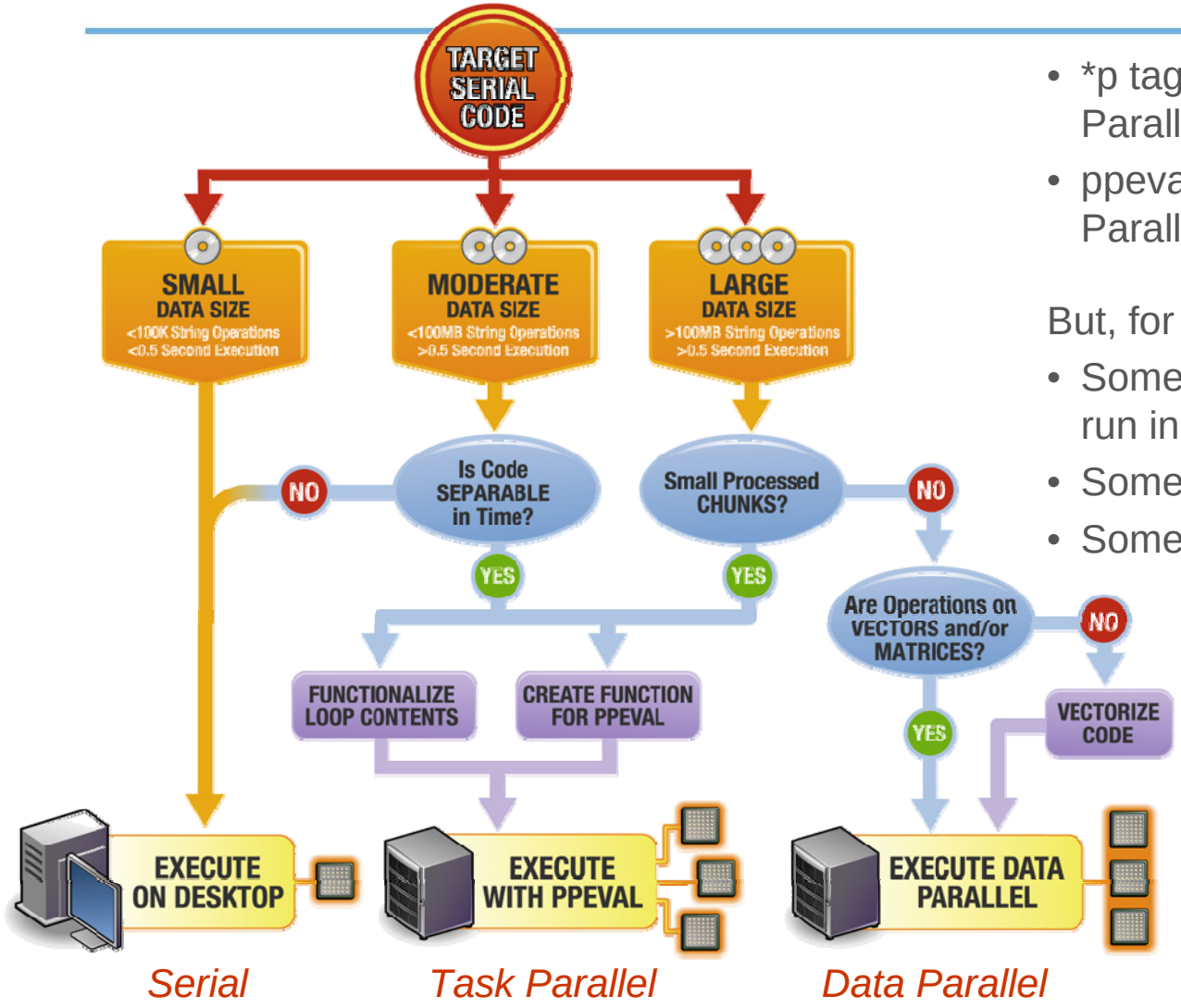
Data Parallel Computing in Star-P™




```
% explicitly parallel with *p
n=10000*p

% implicitly parallel
A = rand(n, n);

% implicitly parallel
x = randn(n, 1);

% implicitly parallel
y = zeros(size(x));

while norm(x-y) / norm(x) > 1e-11
    y = x;
    x = A*x;
    x = x / norm(x);
end;
```

- Data sizes >100MB
- Execution time > .5 second
- Data not separable
- Operations on vectors and matrices
- Incorporate ***p**
  - Global parallelism
  - Variables become parallel
  - Propagation occurs
    - Results are parallel
    - Functions performed on parallel data

Programming for Best Performance - 1



- *p tag triggers automatic Data Parallel computation
- ppeval triggers automatic Task Parallel computation

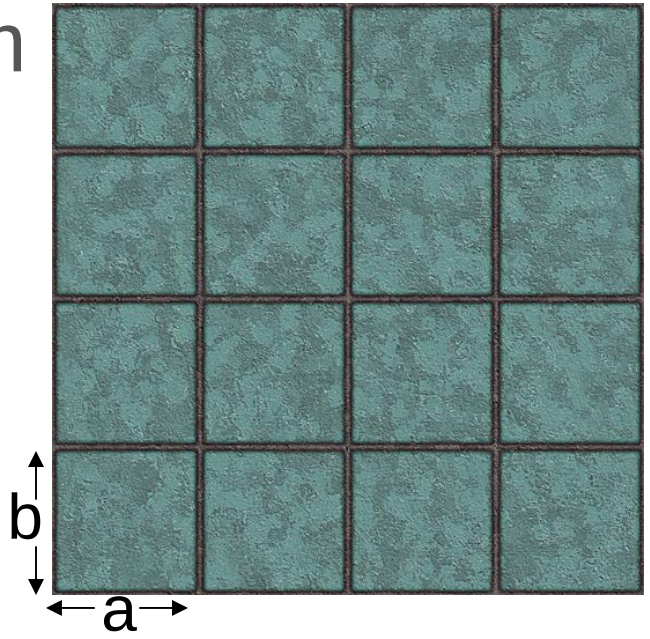
But, for high performance:

- Some program segments are best run in Serial mode
- Some in Task Parallel mode
- Some in Data Parallel mode

INTERACTIVE supercomputing

Classroom Homework

- The Buffon Needle Problem



Buffon(1,1,.5,1000*p)

```
function z=Buffon(a,b,l, trials)
r=rand(trials,3);
x=a*r(:,1)+l*cos(2*pi*r(:,3)); y=b*r(:,2)+l*sin(2*pi*r(:,3));
inside = (x >= 0) & (y>=0) & (x <= a) & (y <= b);
buffonpi=(2*l*(a+b) - l^2)/ (a*b*(1-sum(inside)/trials));
```

Classroom Experiment

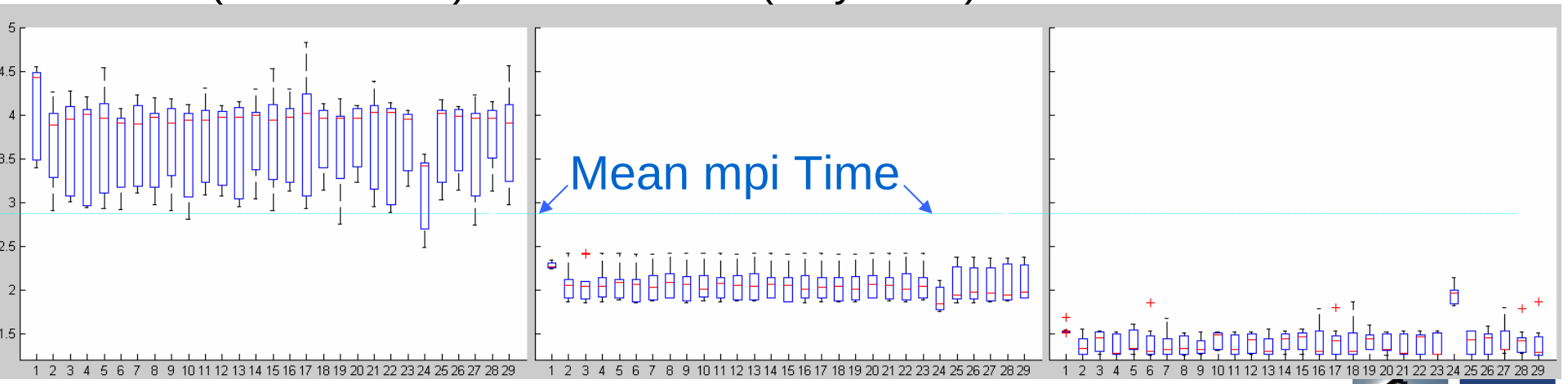
- A data collector's dream:
  - 29 students, each code run in MPI and three versions of Star-P. Some students more skilled with MPI than others.



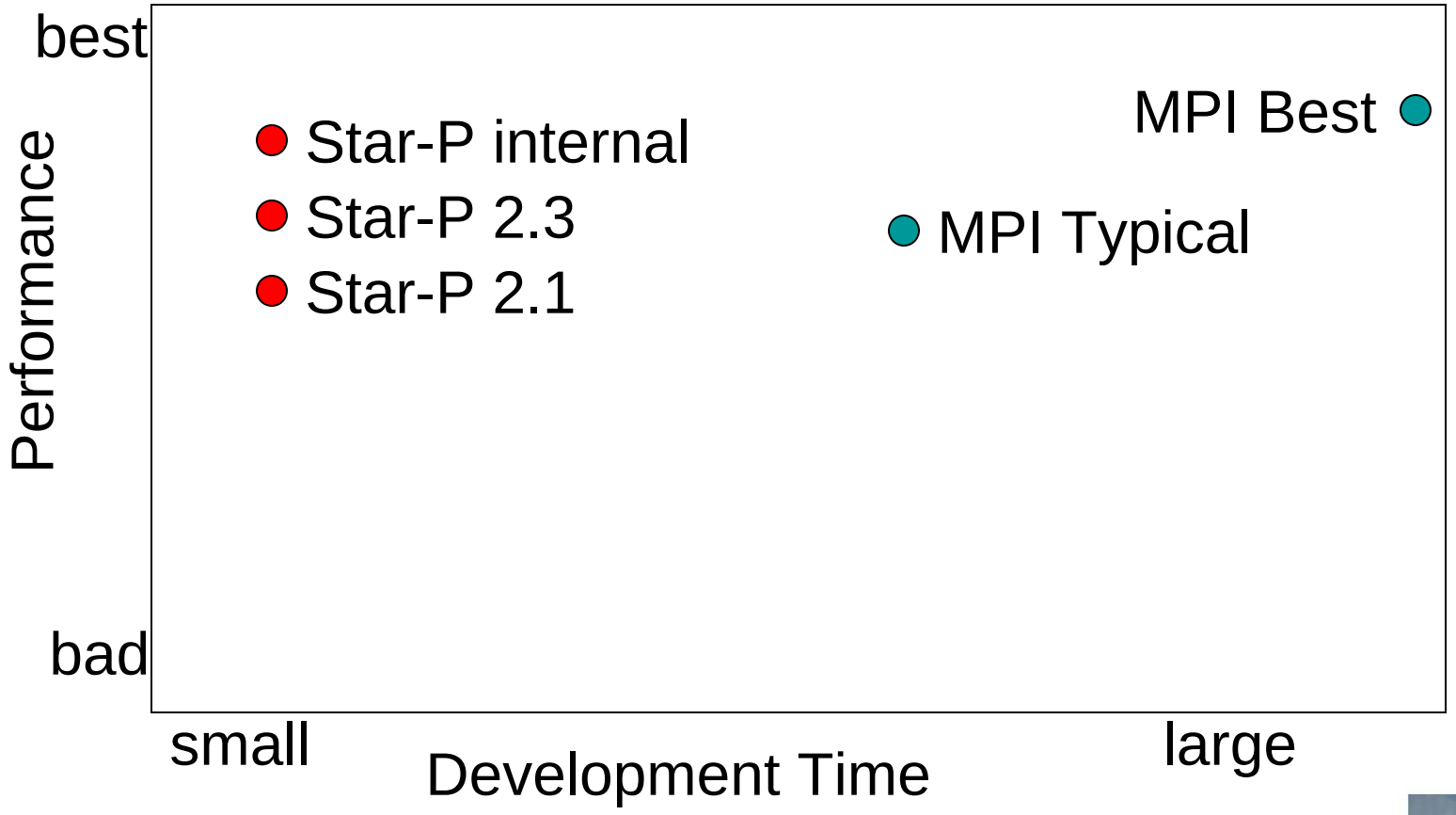
Classroom Experiment

- A data collector's dream:
  - 29 students, each code run in MPI and three versions of Star-P. Some students more skilled with MPI than others.

Star-P 2.1 (March 2006) Star-P 2.3 (May 2006) Star-P internal



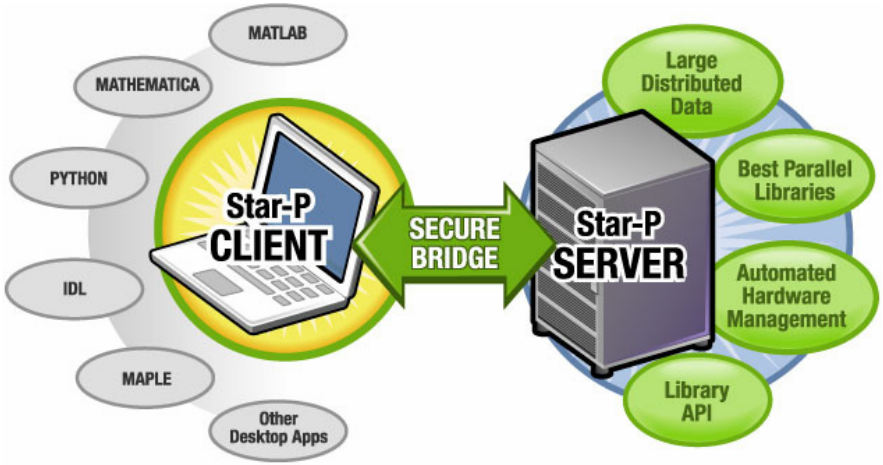
Productivity Study – Kepner diagram



Desktop Applications You Love
The High Performance You Crave

INTERACTIVE SUPERCOMPUTING

"Parallel Computing done right"

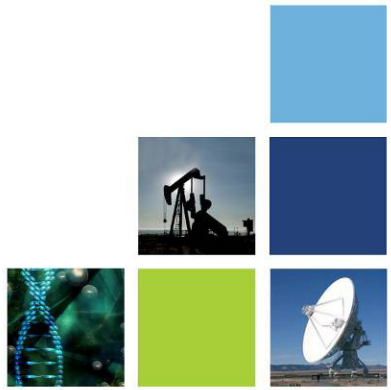


INTERACTIVE supercomputing

Information

www.interactivesupercomputing.com

edelman@mit.edu



INTERACTIVE supercomputing

Star-P™ System Configurations - 1

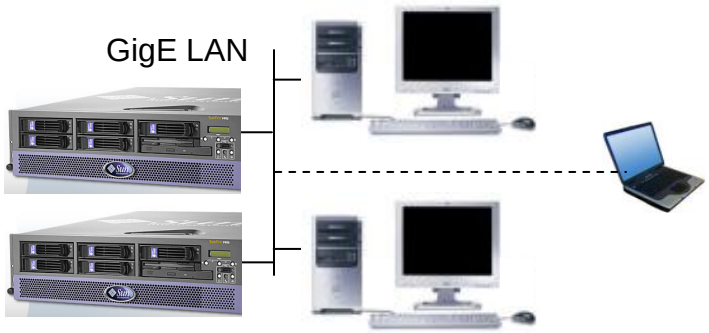
x86/64 Architectures: Opteron and/or Xeon 5100

Multi-core SMP Servers

Example:
Client – Linux/Windows desktop/laptop
Server – Sun Fire X4600, 8 dual-core Opterons (16 cores), SUSE Linux

Star-P Server – 8-socket license
Star-P Client – unlimited number of users
Local or remote access

Multi-core Clusters



Example:
Client – Linux/Windows desktop/laptop
Servers – 4x HP ProLiant BL25p (16 Opteron cores), SUSE
2x SGI Altix XE (8 Xeon 5100 cores), Redhat

Star-P Server – 12-socket license
Star-P Client – unlimited number of users
Local or remote access

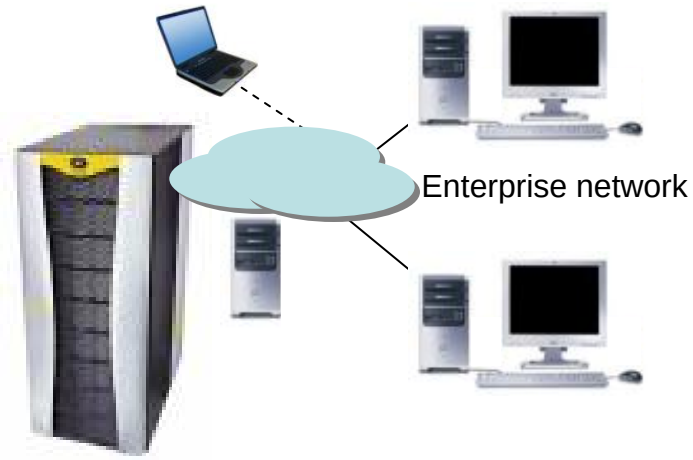
Example Systems (SMP Servers and Clusters)

Opteron	Xeon 5100
HP: ProLiant BL25p, DL145G, DL385, DL585	SGI: Altix XE
Sun: SunFire x4100, x4200, x4600	HP: ProLiant BL20p G4, DL140 G3, DL360 G5, DL380 G5
Newisys: 4300-E,	Dell: PowerEdge 1950, 1955, 2950
Verari: 2510,	Penguin: Relion 1600, 2600
Penguin: Altus 3400	Verari:RM2220, VB1220

Star-P™ System Configurations - 2

IA64 (Itanium) Architecture

Traditional HPC Servers / SMP



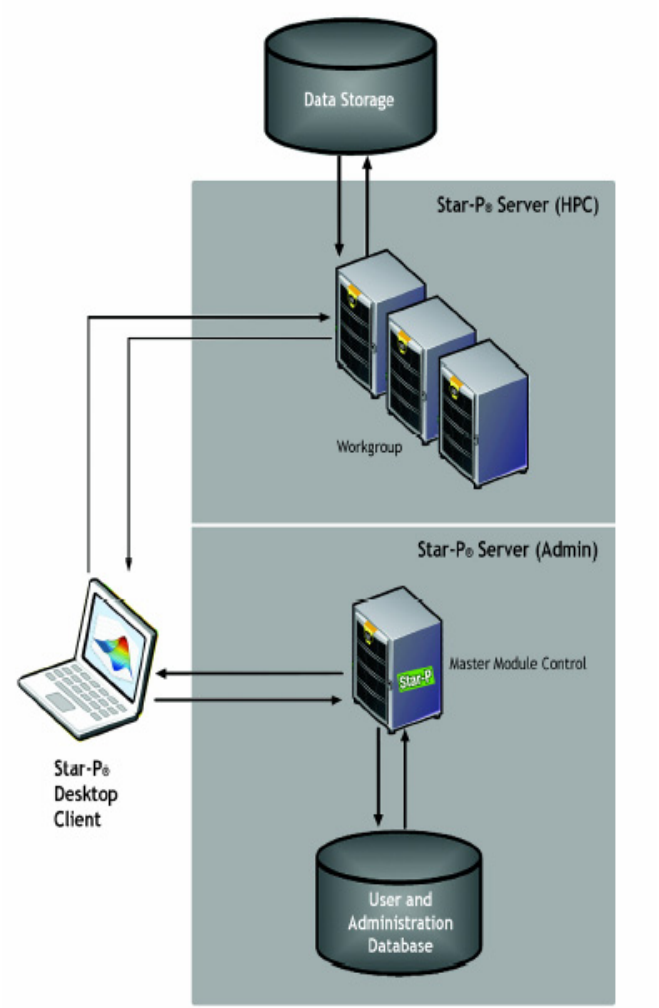
Example:
Client – Linux/Windows desktop/laptop
Server – SGI Altix 32 CPUs, NUMAflex Architecture, SGI ProPak 4 (SUSE SLES9 Linux)

Star-P Server – 32-CPU license
Star-P Client – unlimited number of users,
Star-P Admin Server software for managed access and resource allocation

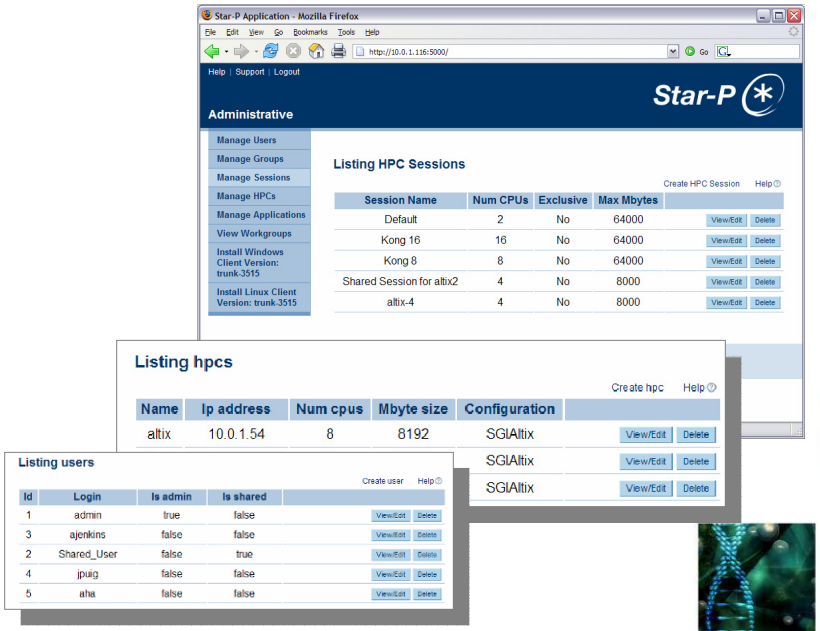
Example IA64 Systems:
SGI Altix 350, SGI Altix 450

INTERACTIVE supercomputing

Star-P Architecture - Logical



- Client
- Workgroup server(s)
- Master Control Module
- User & Admin database
- Data Storage



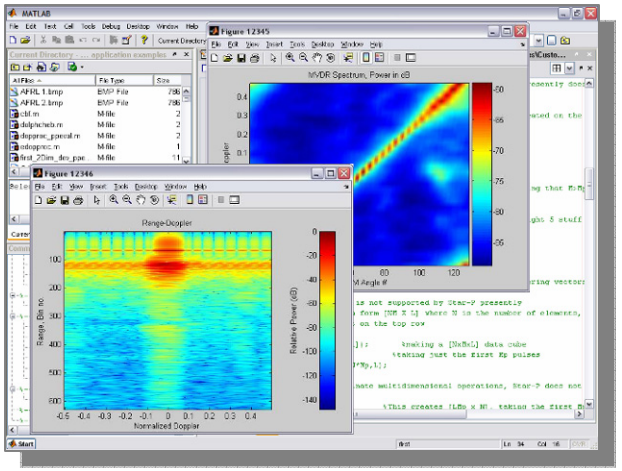
INTERACTIVE supercomputing

Applications by Industries

INTERACTIVE supercomputing

Radar Signal Processing

- Air Force Labs in Rome, NY
- Application: Radar Analysis & System Design
- Challenge: analysis of growing data sets
  - Satellite-based
  - Real time
- Star-P Solution:
  - Reuse existing MATLAB codes
  - Solve larger problems (TB's)
  - Interactive "what if" scenarios

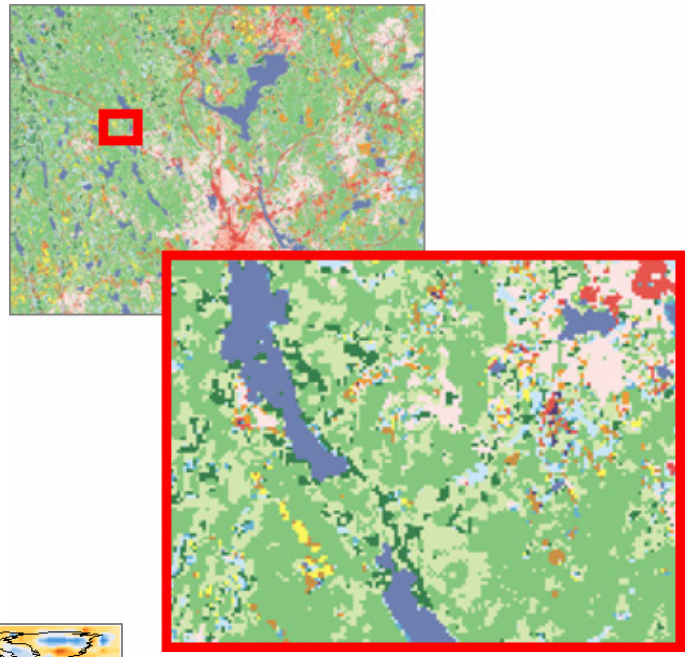


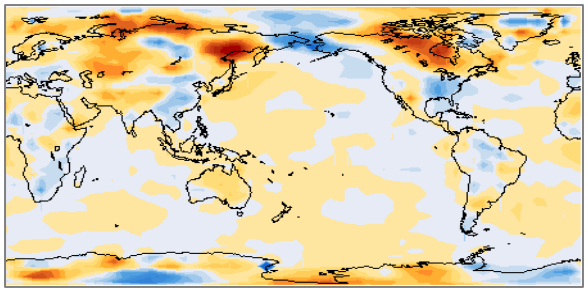
Econometric Modeling

- Columbia University's Earth Institute

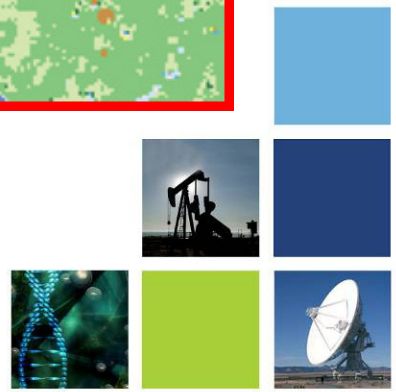
THE EARTH INSTITUTE AT COLUMBIA UNIVERSITY

- Application:
  - Understanding interactions of climate, crop selection, and impact on local populations
  - Development of public policy, insurance, relief programs
- Star-P Solution:
  - Interactive development of complex statistical model
  - Scale to enormous data sets





INTERACTIVE supercomputing



Molecular Simulation

- Department of Chemistry, M.I.T.
- Application:
  - molecular modeling of thermodynamic properties from first principles
  - Impacts smog, weather patterns
- Star-P Solution:
  - Transparent parallelization of existing MATLAB models
  - Global array syntax to solve large systems of equations with 16-P Altix