OpenSees

Parallel & Distributed Processing With OpenSees

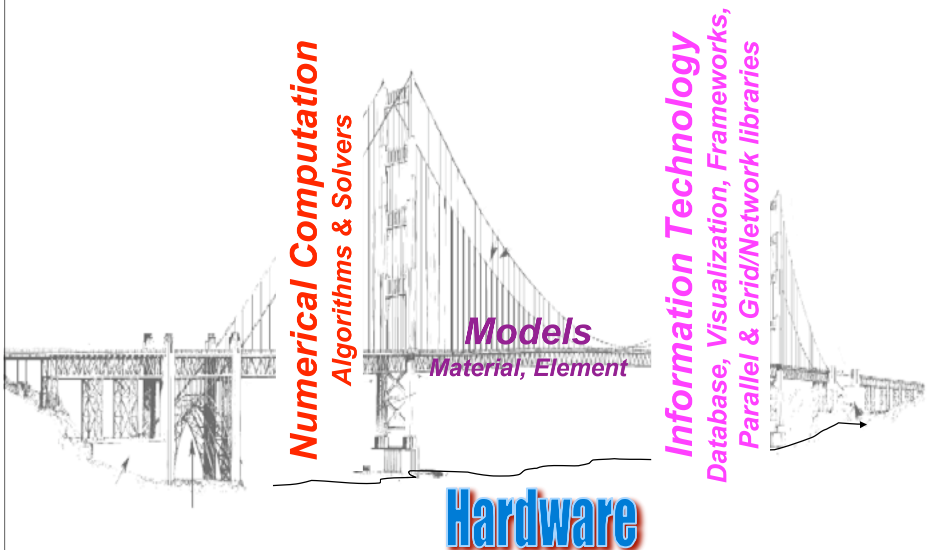
Frank McKenna
UC Berkeley

OpenSeesDays Shanghai 2011

Overview

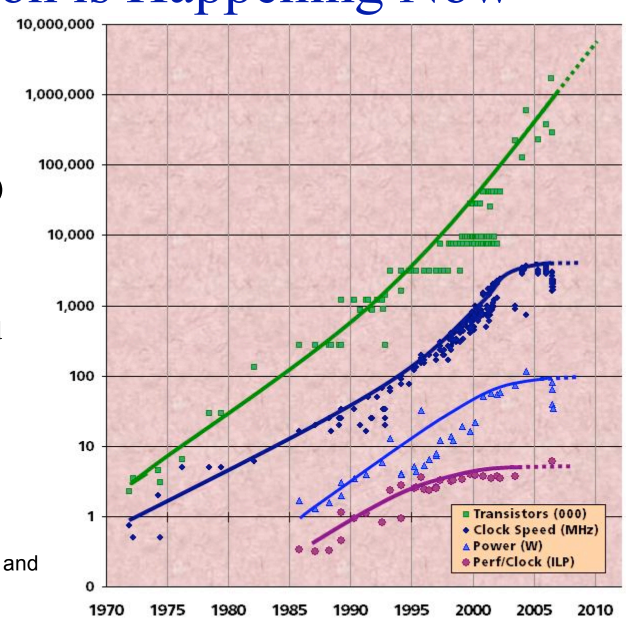
- Hardware Trends
- Parallel Computing & OpenSees
- Cloud Computing & OpenSees

Building Blocks for Simulation



Revolution is Happening Now

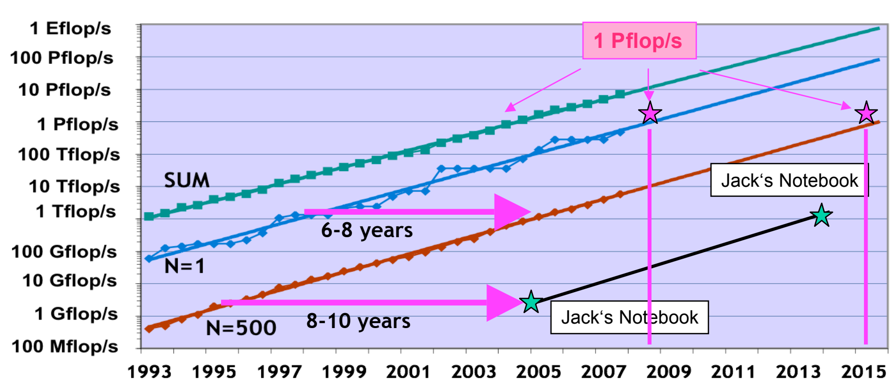
- Chip density is continuing increase ~2x every 1.5-2 years (Moore's law)
  - Clock speed is not
  - Number of processor cores may double instead
- There is little or no more hidden parallelism (ILP) to be found
- Parallelism must be managed by



Source: Intel, Microsoft (Sutter) and Stanford (Olukotun, Hammond)

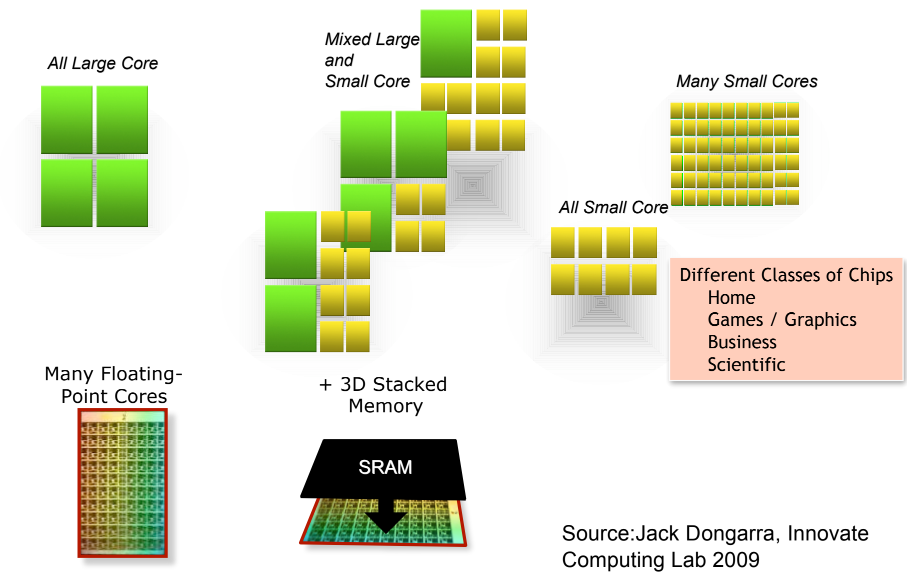
TOP500 SUPERCOMPUTER SITES

Performance Projection



Source:Jack Dongarra, Innovate Computing Lab 2009

Single Machine Architecture?



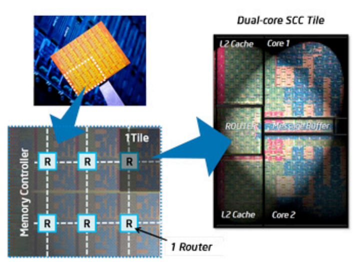
Source:Jack Dongarra, Innovate Computing Lab 2009

Intel's Cloud Processor

Inside the Single-chip Cloud Computer

The name "Single-chip Cloud Computer" reflects the fact that the architecture resembles a scalable cluster of computers such as you would find in a cloud, integrated into silicon.
The research chip features:

- 24 "tiles" with two IA cores per tile
- A 24-router mesh network with 256 GB/s bisection bandwidth
- 4 integrated DDR3 memory controllers
- Hardware support for message-passing

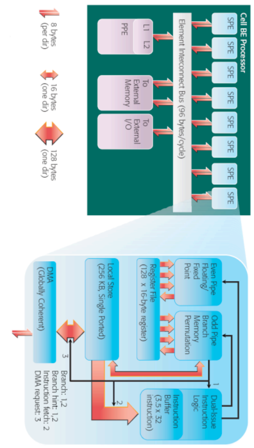


In a sense, the SCC is a microcosm of cloud datacenter. Each core can run a separate OS and software stack and act like an individual compute node that communicates with other compute nodes over a packet-based network.

One of the most important aspects of the SCC's network fabric architecture is that it supports "scale-out" message-passing programming models that have been proven to scale to 1000s of processors in cloud datacenters. Though each core has 2 levels of

Cell Processor

- PlayStation 3 based on “Cell” Processor
- Each Cell contains a PowerPC and 8 self contained vector processing units (SPU’s).
- Power PC at 3.2 GHz
  - DGEMM at 5 Gflop/s
  - Altivec peak at 25.6 Gflop/s
    - Achieved 10 Gflop/s SGEMM
- 8 SPUs
  - 204.8 Gflop/s peak!
  - The catch is that this is for 32 bit floating point; (Single Precision SP)
  - And 64 bit floating point runs at 14.6 Gflop/s total for all 8 SPEs!!



Source:Jack Dongarra, Innovate Computing Lab 2009

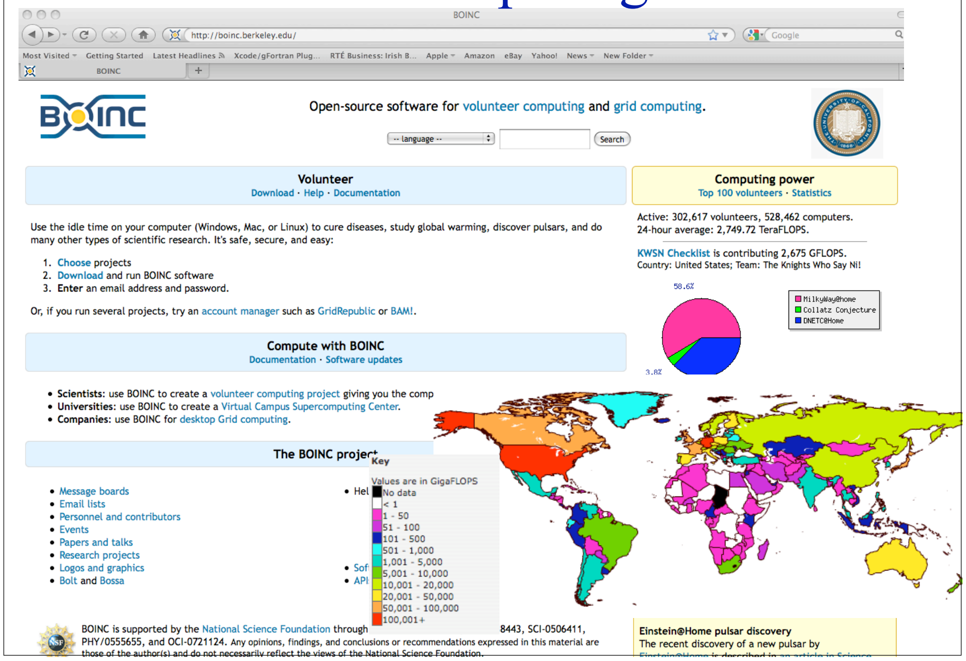
9

Bell’s Law

Bell's Law of Computer Class formation was discovered about 1972. It states that technology advances in semiconductors, storage, user interface and networking advance every decade enable a new, usually lower priced computing platform to form. Once formed, each class is maintained as a quite independent industry structure. This explains mainframes, minicomputers, workstations and Personal computers, the web, emerging web services, palm and mobile devices, and ubiquitous interconnected networks. We can expect home and body area networks to follow this path.

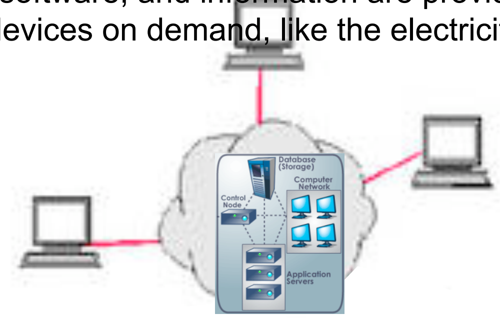
Gordon Bell, http://research.microsoft.com/~GBell/Pubs.htm

Grid Computing



Cloud Computing

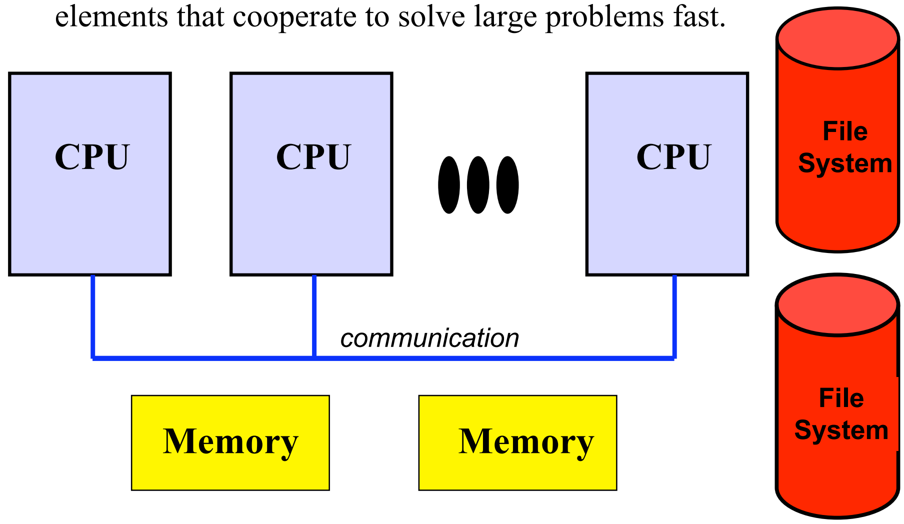
Cloud computing is internet-based computing , whereby shared resources, software, and information are provided to computers and other devices on demand, like the electricity grid. source: wikipedia



•Applications are run and data is stored on remote machines in the cloud.
•User accesses the applications and files using an internet based application, e.g. web browser for Google Docs and NEEShub.

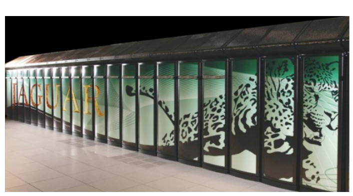
What is a Parallel Computer?

- A *parallel computer* is a collection of processing elements that cooperate to solve large problems fast.



Why should you care?

- They will save you time
- They will allow you to solver larger problems.
- They are here whether you like it or not!



TOP 10 Sites for June 2010

For more information about the sites and systems in the list, click on the links or view the complete list.

Rank	Site	Computer
1	Oak Ridge National Laboratory United States	Jaguar - Cray XT5-HE Opteron Six Core 2.6 GHz Cray Inc.
2	National Supercomputing Centre in Shenzhen (NSCS) China	Nebulae - Dawning TC3600 Blade, Intel X5650, NVidia Tesla C2050 G Dawning
3	DOE/NNSA/LANL United States	Roadrunner - BladeCenter QS22/LS21 Cluster, PowerXCell 8i 3.2 Ghz Opteron DC 1.8 GHz, Voltaire Infiniband IBM
4	National Institute for Computational Sciences/University of Tennessee United States	Kraken XT5 - Cray XT5-HE [illegible] Cray Inc.
5	Forschungszentrum Juelich (FZJ) Germany	[illegible]
6	NASA/Ames Research Center/[illegible] United States	[illegible] X/8400EX, Xeon HT QC 3.0/Xeon Wes [illegible] Infiniband SGI
7	National SuperComputer Center in Tianjin/NUDT China	Tianhe-1 - NUDT TH-1 Cluster, Xeon E5540/E5450, ATI Radeon HD 4 Infiniband NUDT
8	DOE/NNSA/LLNL United States	BlueGene/L - eServer Blue Gene Solution IBM
9	Argonne National Laboratory United States	Intrepid - Blue Gene/P Solution IBM
10	Sandia National Laboratories / National Renewable Energy Laboratory	Red Sky - Sun Blade x6275, Xeon X55xx 2.93 Ghz, Infiniband

BEFORE YOU GET ALL EXCITED

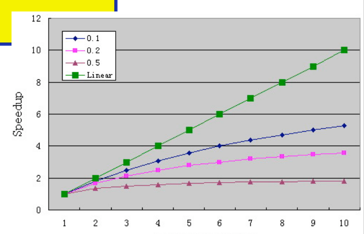
Speedup & Amdahl's Law

$$\boxed{speedup_{PC}(p) = \frac{Time(1)}{Time(p)}}$$

$$\boxed{Speedup_{PC} = \frac{T_1}{\alpha T_1 + \frac{(1-\alpha)T_1}{n}} \rightarrow \frac{1}{\alpha} \ as \ n \rightarrow \infty}$$

Portion of sequential — α

# of processors — n



Improving Real Performance

Peak Performance grows exponentially, a la Moore's Law

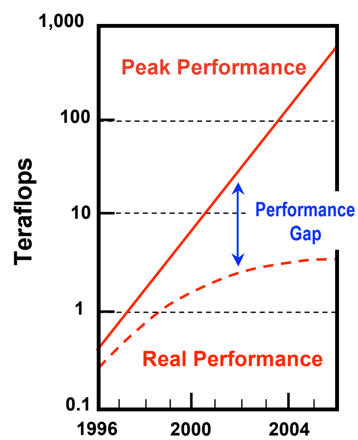
- In 1990's, peak performance increased 100x; in 2000's, it will increase 1000x

But efficiency (the performance relative to the hardware peak) has declined

- was 40-50% on the vector supercomputers of 1990s
- now as little as 5-10% on parallel supercomputers of today

Close the gap through ...

- Mathematical methods and algorithms that achieve high performance on a single processor and scale to thousands of processors
- More efficient programming models and tools for massively parallel supercomputers

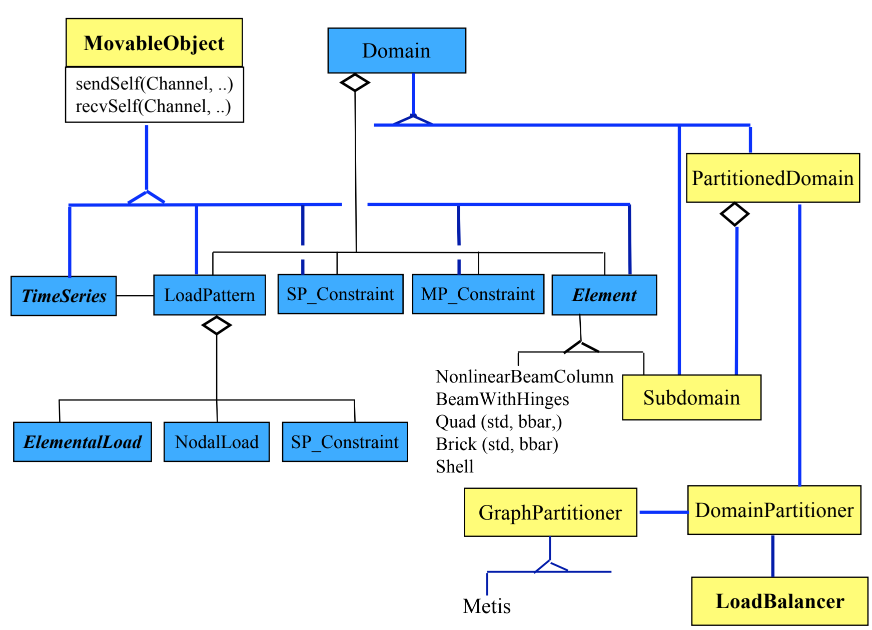


Source: Jim Demmell, CS267 Course Notes

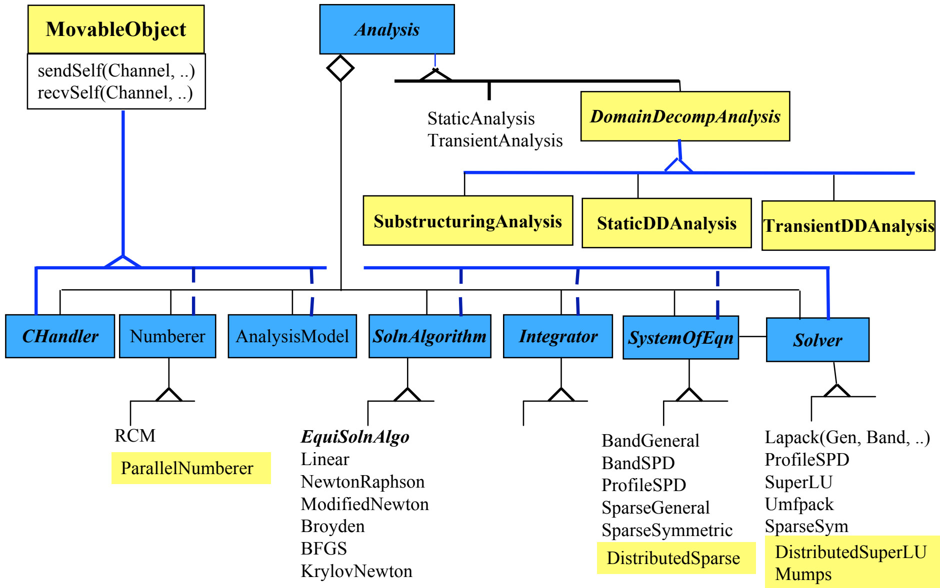
What is OpenSees?

- OpenSees is an Open-Source Software Framework written in C++ for developing nonlinear Finite Element Applications for both sequential and **PARALLEL** environments.

Domain Classes



Analysis Classes



The OpenSees Interpreters

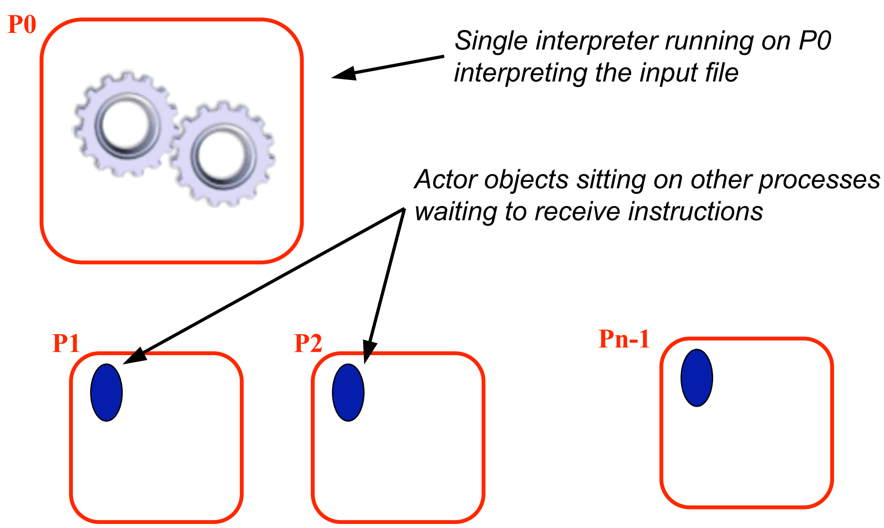
- OpenSees.exe, OpenSeesSP.exe and OpenSeesMP.exe are applications that extend the Tcl interpreter for finite element.

So What are OpenSeesSP.exe and OpenSeesMP.exe ?

Parallel OpenSees Interpreters

- OpenSeesSP: An application for large models which will parse and execute the exact same script as the sequential application. The difference being the element state determination and equation solving are done in parallel.
- OpenSeesMP: An application for **BOTH** large models and parameter studies.

OpenSeesSP: An application for Large Models



Modified Commands

- System command is modified to accept new parallel equation solvers

system Mumps

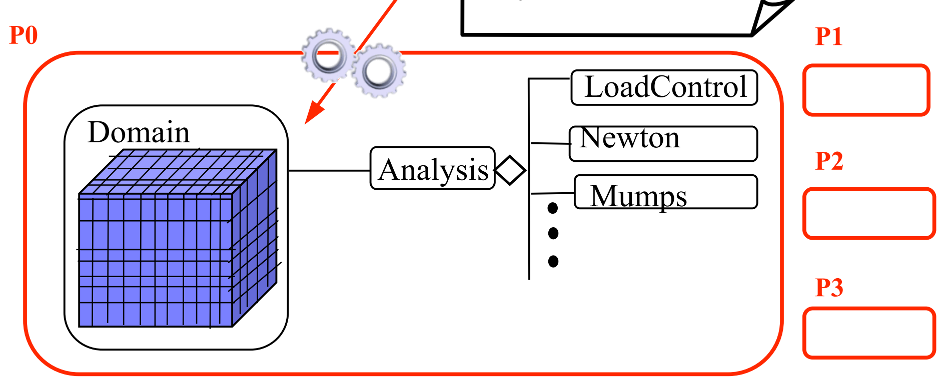
system Diagonal

Model Built and Analysis Constructed in P0

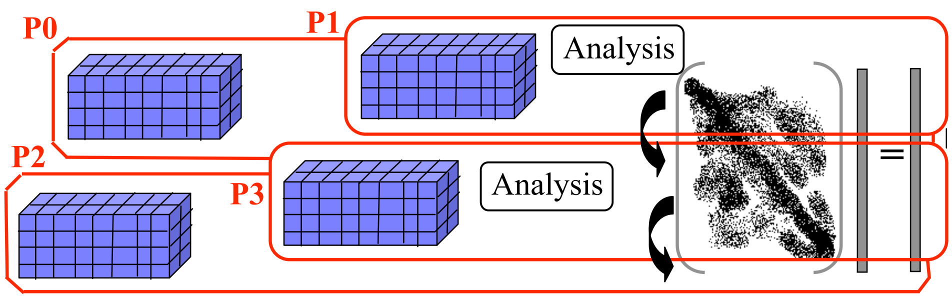
```
#build the model
source model.tcl
#build the analysis
system Mumps
constraints Transformation
numberer Plain
test NormDispIncr 1.0e-12 10 3
algorithm Newton
integrator LoadControl
analysis Static
```

Single interpreter running on P0

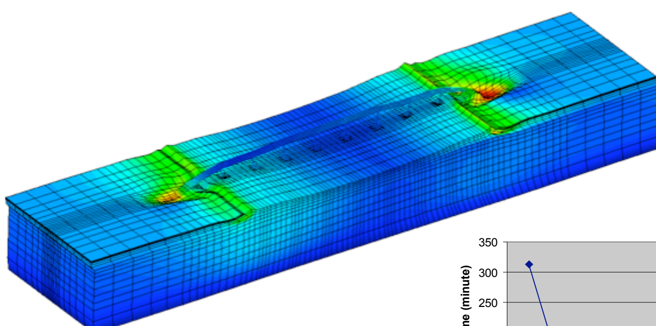
Interpreting the input file



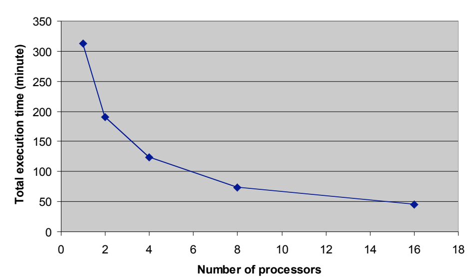
```
#build the model
source modelP.tcl
#build the analysis
system Mumps
constraints Transformation
numberer Plain
test NormDispIncr 1.0e-12 10 3
algorithm Newton
integrator LoadControl
analysis Static
analyze 10
```



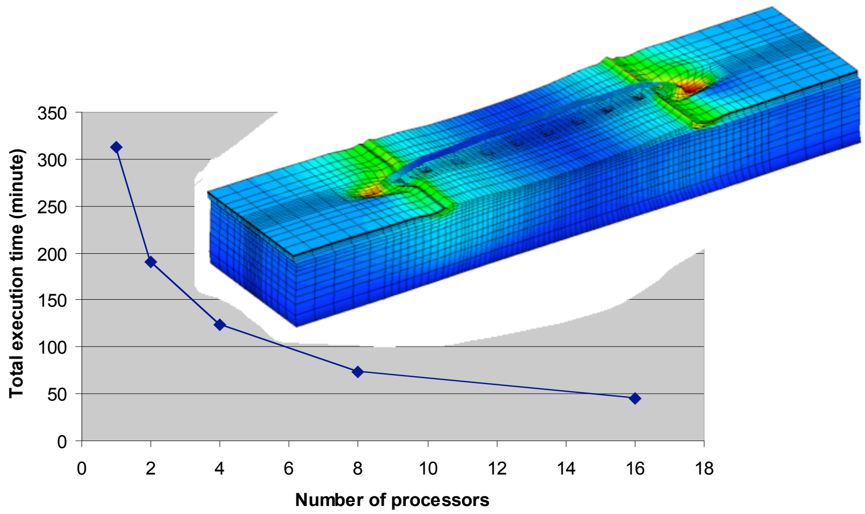
Example Usage:
Humboldt Bay Bridge Model



100,000+ DOF Model
Implicit Integration
Mumps Direct Solver



Example Usage:
Humboldt Bay Bridge Model



Run	el. size (m)	Elements	Nodes	DOFs
A	20	54,026	59,032	156,768
B	10	404,751	424,512	1,193,283
C	5	3,130,301	3,208,822	9,307,563
D	2.5	24,615,801	24,928,842	73,515,123

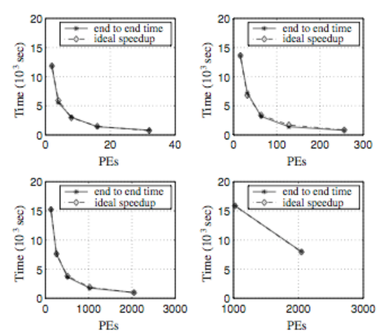
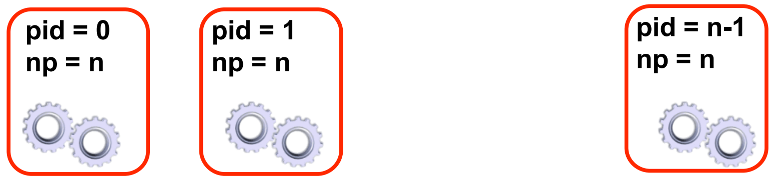


Fig. 18 Fixed-size, scalability plot at SDSC's DataStar. Upper row is runs A (*left*) and B (*right*), lower row is runs C (*left*) and D (*right*) (Table 3)

OpenSeesMP: An application for Large Models and Parameter Studies



Each process is running an interpreter and can determine it's unique process number and the total number of processes in computation

Based on this script can do different things

```
# source in the model and analysis procedures
set pid [getPID]
set np [getNP]

# build model based on np and pid
source modelP.tcl
doModel {$pid $np}

# perform gravity analysis
system Mumps
constraints Transformation
numberer Parallel
test NormDispIncr 1.0e-12 10 3
algorithm Newton
integrator LoadControl 0.1

analysis Static

set ok [analyze 10]
return $ok
```

New Commands added to OpenSeesMP:

- A Number of new commands have been added:

1. getNP   returns number of processes in computation.
2. getPID   returns unique pocess id {0,1, .. NP-1}
3. send -pid pid? data   pid = { 0, 1, .., NP-1}
4. recv -pid pid? variableName   pid = {0,1 .., NP-1, ANY}
5. barrier
6. domainChange

- These commands have been added to ALL interpreters (OpenSees, OpenSeesSP, and OpenSeesMP)

Example

ex2.tcl

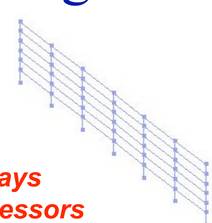
```
set pid [getPID]
set np [getNP]
if {$pid == 0 } {
   puts "Random:"
   for {set i 1 } {$i < $np} {incr i 1} {
      recv -pid ANY msg
      puts "$msg"
   }
} else {
   send -pid 0 "Hello from $pid"
}
barrier
if {$pid == 0 } {
   puts "\nOrdered:"
   for {set i 1 } {$i < $np} {incr i 1} {
      recv -pid $i msg
      puts "$msg"
   }
} else {
   send -pid 0 "Hello from $pid"
}
```

Terminal — bash — 80x32

```
bin> mpirun -np 10 OpenSeesMP ex2.tcl

          OpenSees -- Open System For Earthquake Engineering Si
         Pacific Earthquake Engineering Research Center -- 1.7.

             (c) Copyright 1999,2000 The Regents of the Univers
                              All Rights Reserved
      (Copyright and Disclaimer @ http://www.berkeley.edu/OpenSe


Random:
Hello from 1
Hello from 3
Hello from 5
Hello from 6
Hello from 8
Hello from 2
Hello from 4
Hello from 7
Hello from 9
Ordered:
Hello from 1
Hello from 2
Hello from 3
Hello from 4
Hello from 5
Hello from 6
Hello from 7
Hello from 8
Hello from 9
```

Steel Building Study

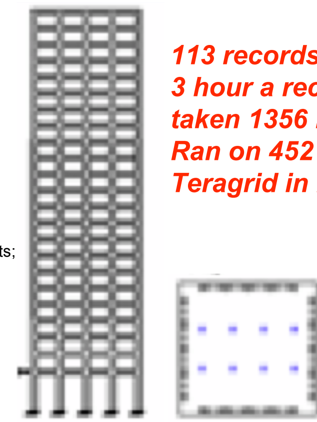
```
set pid [getPID]
set np [getNP]
set recordsFileID [open "peerRecords.txt" r]
set count 0;

foreach gMotion [split [read $recordsFileID] ¥n] {
 if {[expr $count % $np] == $pid} {

 source model.tcl
 source analysis.tcl

 set ok [doGravity]

 loadConst -time 0.0

 set gMotionList [split $gMotion "/"]
 set gMotionDir [lindex $gMotionList end-1]
 set gMotionNameInclAT2 [lindex $gMotionList end]
 set gMotionName [string range $gMotionNameInclAT2 0 end-4 ]

 set Gaccel "PeerDatabase $gMotionDir $gMotionName -accel 384.4 -dT dT -nPts nPts"
 pattern UniformExcitation 2 1 -accel $Gaccel

 recorder EnvelopeNode -file $gMotionDir$gMotionName.out -node 3 4 -dof 1 2 3 disp

 doDynamic [expr $dT*$nPts] $dT

  wipe
 }

 incr count 1;
}
```



7200 records
2 min a record
240 hours or 10 days
Ran on 2000 processors
on teragrid in less than 15 min.

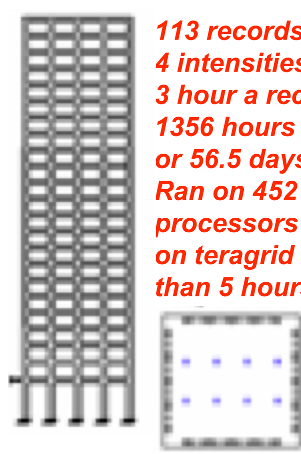
Concrete Building Study

```
set pid [getPID]
set np [getNP]
set count 0;
source parameters.tcl
source ReadSMDFileNewFormat.tcl;
foreach GMfile $iGMFile {
   foreach Factor1248 $iFactor1248 {

      if {[expr $count % $np] == $pid} {

          set inFile $GMdir/$GMfile.AT2
          set outFile $GMdir/$GMfile.g3;
          ReadSMDFileNewFormat $inFile $outFile dt npts;

          wipe
          source GravityAnalysisScript.tcl

          loadConst -time 0.0;
          wipeAnalysis

          source  EQ_Recorder.tcl
          source EQAnalysisScript.tcl

         if {$ok == 0} {
             puts "Process $pid $GMfile x $Factor1248 FINISHED OK modelTime [getTime]]"
          } else {
             puts "Process $pid $GMfile x $Factor1248 FINISHED FAIL modeTime [getTime] desiredTime $TmaxAnalysis]"
          }
          incr count 1
      }
   }
}
```



113 records, 4 intensities
3 hour a record, would have taken 1356 hours or 56.5 days
Ran on 452 processors of a Teragrid in less than 5 hours.

Concrete Building Study



113 records,
4 intensities
3 hour a record
1356 hours
or 56.5 days
Ran on 452
processors
on teragrid in less
than 5 hours.

Modified Commands

- Some existing commands have been modified to allow analysis of large models in parallel:
  1. numberer

     numberer ParallelPlain

     numberer ParallelRCM

  2. system

     system Mumps <-ICNTL14 %?>

  3. integrator

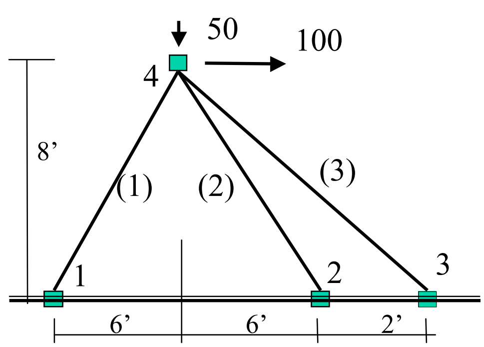
     integrator ParallelDisplacementControl node? Dof? dU?

- Use these ONLY IF PARALLEL MODEL

Example Parallel Model:

ex4.tcl

```
set pid [getPID]
set np [getNP]
if {$np != 2}  exit

model BasicBuilder -ndm 2 -ndf 2
uniaxialMaterial Elastic 1 3000
if {$pid == 0} {
   node 1   0.0  0.0
   node 4  72.0 96.0
   fix 1 1 1
   element truss 1 1 4 10.0 1
   pattern Plain 1 "Linear" {
      load 4 100 -50
   }
} else {
   node 2 144.0  0.0
   node 3 168.0  0.0
   node 4  72.0 96.0
   fix 2 1 1
   fix 3 1 1
   element truss 2 2 4 5.0 1
   element truss 3 3 4 5.0 1
}
```



	E	A
1	3000	10
2	3000	5
3	3000	5

Example Parallel Analysis:

```
#create the recorder
recorder Node -file node4.out.$pid -node 4 -dof 1 2 disp

#create the analysis
constraints Transformation
numberer ParallelPlain
system Mumps
test NormDispIncr 1.0e-6 6 2
algorithm Newton
integrator LoadControl 0.1
analysis Static

#perform the analysis
analyze 10

# print to screen node 4
print node 4
```

```
Terminal — bash — 86x31
bin> mpirun -np 2 OpenSeesMP ex4.tcl


         OpenSees -- Open System For Earthquake Engineering Simulation
        Pacific Earthquake Engineering Research Center -- 1.7.5

            (c) Copyright 1999,2000 The Regents of the University of California
                                All Rights Reserved
    (Copyright and Disclaimer @ http://www.berkeley.edu/OpenSees/copyright.html)


 Node: 4
        Coordinates  : 72 96
        commitDisps: 0.530093 -0.177894
        Velocities   : 0 0
         unbalanced Load: 100 -50
        ID : 0 1

Process Terminating 0

 Node: 4
        Coordinates  : 72 96
        commitDisps: 0.530093 -0.177894
        Velocities   : 0 0
         unbalanced Load: 0 0
        ID : 0 1

Process Terminating 1
bin> diff node4.out.0 node4.out.1
bin>
```

Parallel Displacement Control and domainChange!

ex5.tcl

```
source ex4.tcl

loadConst - time 0.0

if {$pid == 0} {
    pattern Plain 2 "Linear" {
        load 4 1 0
    }
}

domainChange

integrator ParallelDisplacementControl 4 1 0.1
analyze 10
```

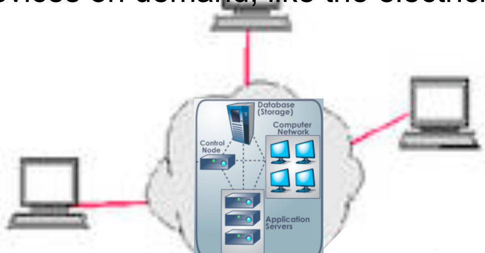
```
Terminal — ba
        Pacific Earthquake Engineering Research Center -- 1.

            (c) Copyright 1999,2000 The Regents of the Unive
                                All Rights Reserved
    (Copyright and Disclaimer @ http://www.berkeley.edu/Oper


 Node: 4
        Coordinates  : 72 96
        commitDisps: 0.530093 -0.177894
        Velocities   : 0 0
         unbalanced Load: 100 -50
        ID : 0 1

 Node: 4
        Coordinates  : 72 96
        commitDisps: 0.530093 -0.177894
        Velocities   : 0 0
         unbalanced Load: 0 0
        ID : 0 1

 Node: 4
        Coordinates  : 72 96
        commitDisps: 1.53009 -0.194007
        Velocities   : 0 0
         unbalanced Load: 200.668 -50
        ID : 0 1

Process Terminating 0

 Node: 4
        Coordinates  : 72 96
        commitDisps: 1.53009 -0.194007
        Velocities   : 0 0
         unbalanced Load: 0 0
```

Things to Watch For

1. Deadlock (program hangs)
   – send/recv messages
   – Opening files for writing & not closing them
2. Race Conditions (different results every time run problem)
   – parallel file system.
3. Load Imbalance
   – poor initial task assignment.

Cloud Computing

Cloud computing is internet-based computing , whereby shared resources, software, and information are provided to computers and other devices on demand, like the electricity grid. source: wikipedia

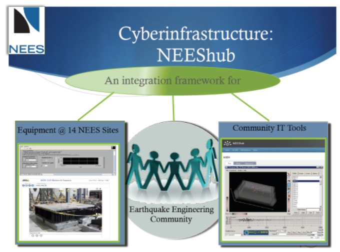


•Applications are run and data is stored on remote machines in the cloud.
•User accesses the applications and files using an internet based application, e.g. web browser for Google Docs and NEEShub.

Pros & Cons?

NEEShub

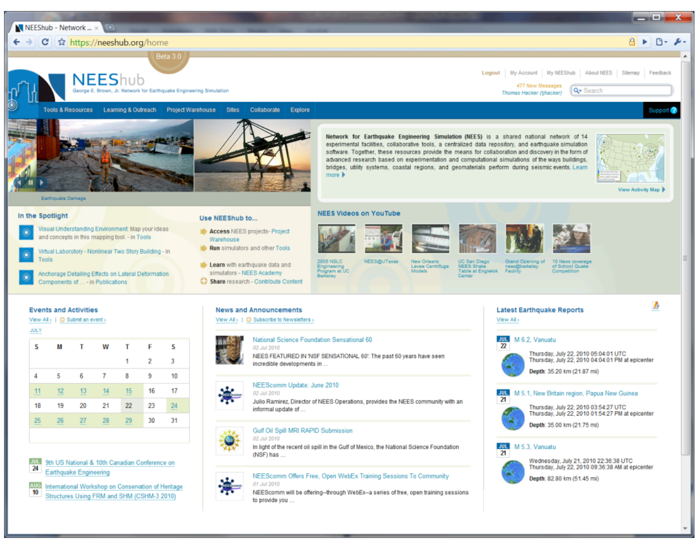
- The power behind NEES at http://nees.org
- Maintained and developed at Purdue by NEEScomm
- Built using proven HUBzero technology (nanoHUb > 100,000 users)
- A science gateway for education and research in earthquake engineering



Through a browser engineers can:

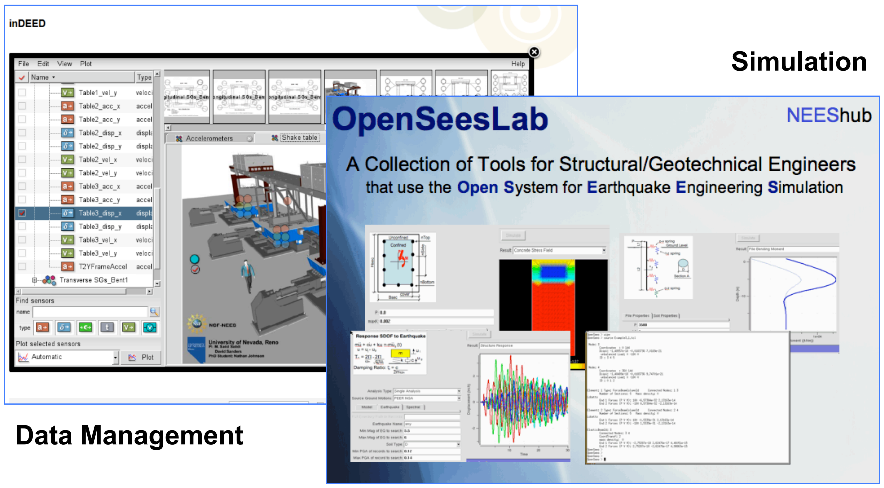
- Upload and view experimental data
- Browse online seminars and courses
- Launch sophisticated tools using remote computational resources (OpenSeesLab)

NEEShub (First Release July 2010)



NEEShub Tools and Resources

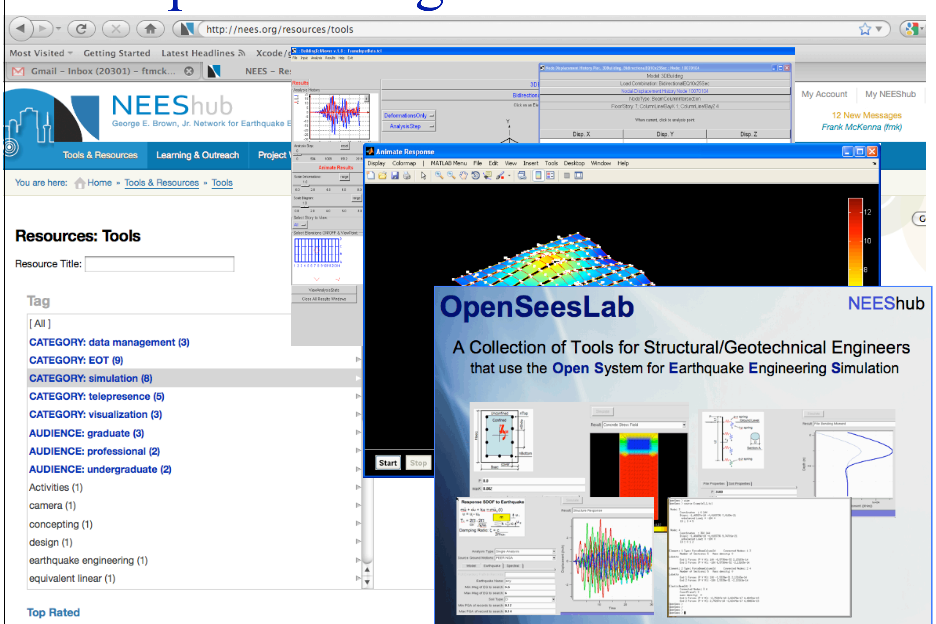
Documents, Learning Objects, Series & TOOLS



Simulation

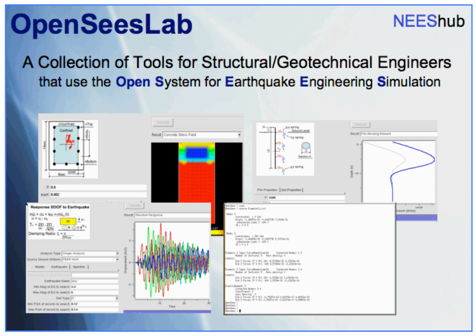
Data Management

http://nees.org/resources/tools



The OpenSeesLab tool:

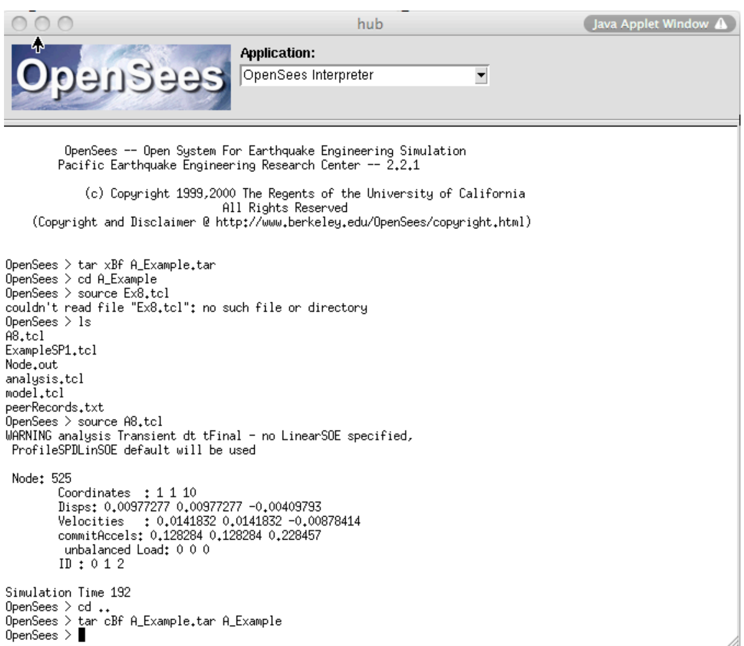
http://nees.org/resources/tools/openseeslab



Is a suite of Simulation Tools powered by OpnSees for:

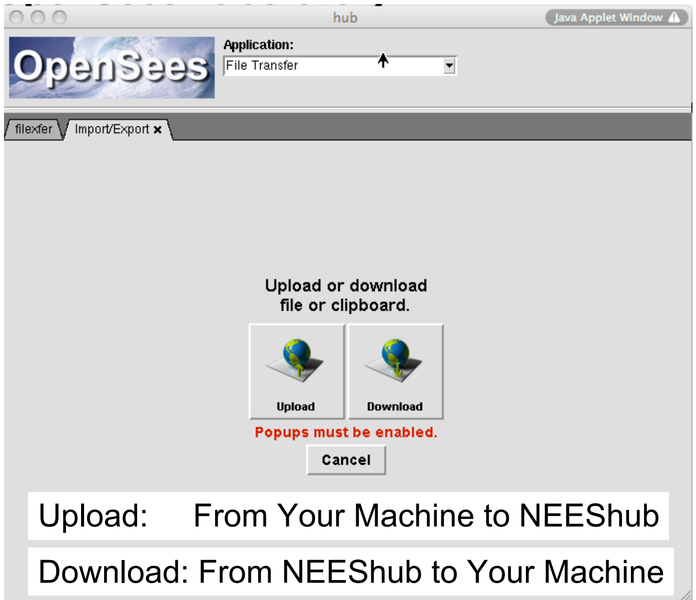
1. Submitting OpenSees scripts (input files) to HUB resources
2. Educating students and practicing engineers
3. Performing useful tasks

OpenSees Interpreter Tool



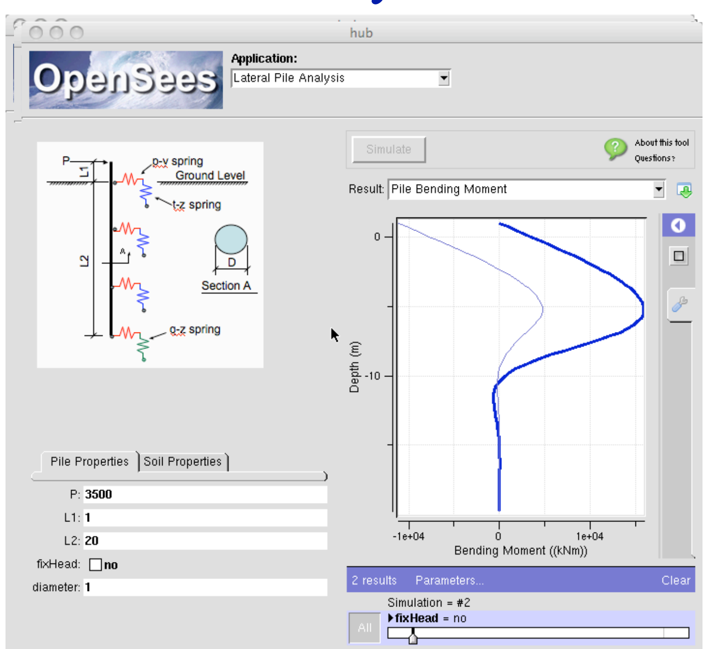

```
         OpenSees -- Open System For Earthquake Engineering Simulation
        Pacific Earthquake Engineering Research Center -- 2.2.1

            (c) Copyright 1999,2000 The Regents of the University of California
                                 All Rights Reserved
    (Copyright and Disclaimer @ http://www.berkeley.edu/OpenSees/copyright.html)

OpenSees > tar xBf A_Example.tar
OpenSees > cd A_Example
OpenSees > source Ex8.tcl
couldn't read file "Ex8.tcl": no such file or directory
OpenSees > ls
A8.tcl
ExampleSP1.tcl
Node.out
analysis.tcl
model.tcl
peerRecords.txt
OpenSees > source A8.tcl
WARNING analysis Transient dt tFinal - no LinearSOE specified,
 ProfileSPDLinSOE default will be used

 Node: 525
        Coordinates  : 1 1 10
        Disps: 0.00977277 0.00977277 -0.00409793
        Velocities   : 0.0141832 0.0141832 -0.00878414
        commitAccels: 0.128284 0.128284 0.228457
         unbalanced Load: 0 0 0
        ID : 0 1 2

Simulation Time 192
OpenSees > cd ..
OpenSees > tar cBf A_Example.tar A_Example
OpenSees >
```

File Transfer Tool



Lateral Pile Analysis

http://opensees.berkeley.edu/wiki/index.php
/Laterally-Loaded_Pile_Foundation
Chris McGann U. Washington



NEEShub things to know:

1. Anyone can get an account (it's free!)
2. You can have 5 sessions running at once
3. The sessions stay alive until you kill them
4. With each session you get a new data directory, some tools by default will store their information there.
5. You have 1GB storage by default (it is expandable!)

Any Questions?