NEESgrid Building the National Virtual Collaboratory for Earthquake Engineering.

OpenSees and NEESgrid Simulation Component User Workshop

Presented by the OpenSees Community and NEESgrid Simulation Project Team

September 2-3, 2004

Sponsored by the National Science Foundation
through the **Pacific Earthquake Engineering Research Center**
and the **NEESgrid System Integration Project**

Context

- PEER Research Program in Performance-Based Earthquake Engineering.
- Development of enabling technology is expected of NSF research centers.
- Many researchers and students for PEER Thrust Area "Simulation and Information Technology" are contributing to technology development and application.
- NEESgrid has adopted OpenSees as the simulation component for the NEES system integration project.

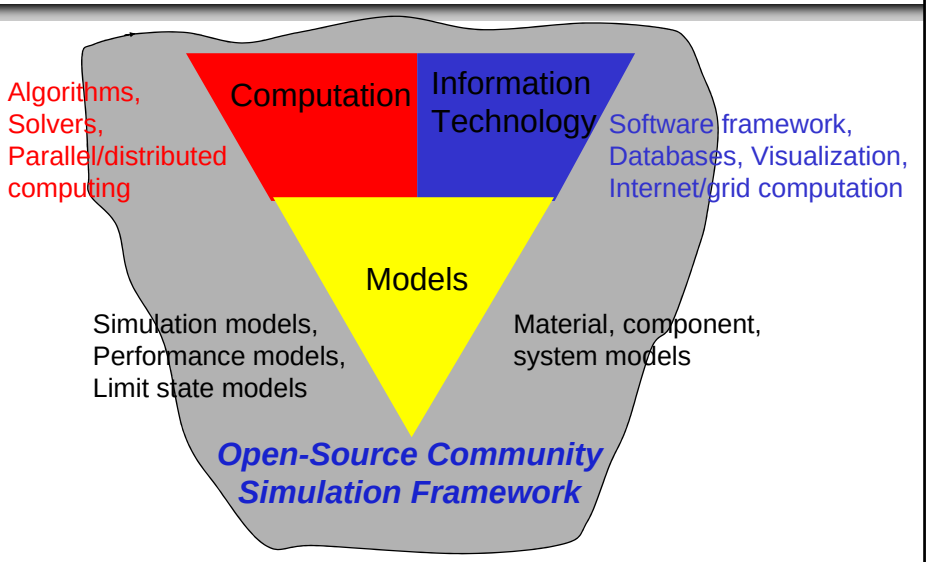
Observations on Current Situation

- Tight binding of models in research and commercial codes is an impediment to new research and implementation of models for professional practice.
- Embedding of computational procedures in codes makes it difficult to experiment and take advantage of computing technology:
  - Parallel and distributed computers
  - Computational grids
- "Closed-source" is the norm, whereas other fields have adopted "open-source" software for communities users.

What is OpenSees?

- A software *framework* for simulation applications in earthquake engineering using finite element methods. OpenSees is not a code.
- A communication mechanism within PEER, and beyond, for exchanging and building upon research accomplishments.
- As open-source software, it has the potential for a community code for earthquake engineering.

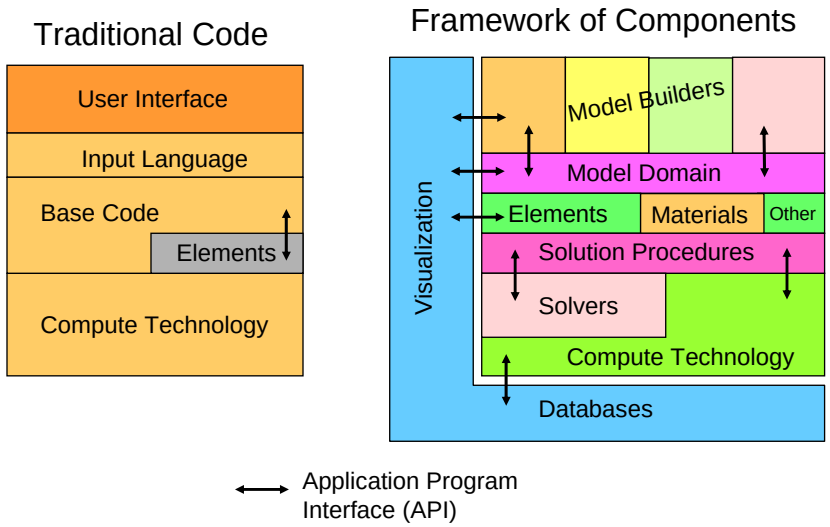
Conceptual Approach for Simulation



Software Framework

- A *framework* is a set of cooperating software components for building applications in a specific domain.
- A framework dictates the architecture of the application. It must represent the design decisions common to the application domain.
- A frameworks is based on the assumption that an architecture will work for most applications within the domain.
- Loose-coupling of components within the framework is essential for extensibility and re-useability for applications.
- Examples: Visualization (GLUT), MS Office, compilers ...
- *A framework is not a "code"*

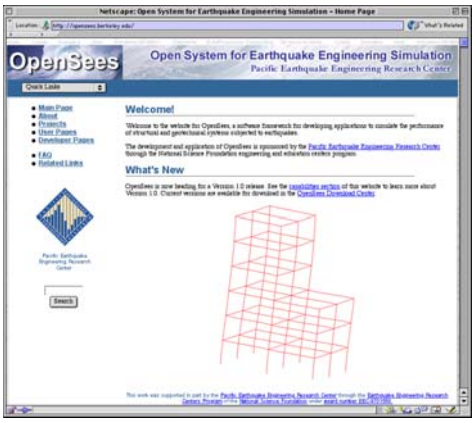
Simulation Software Alternatives



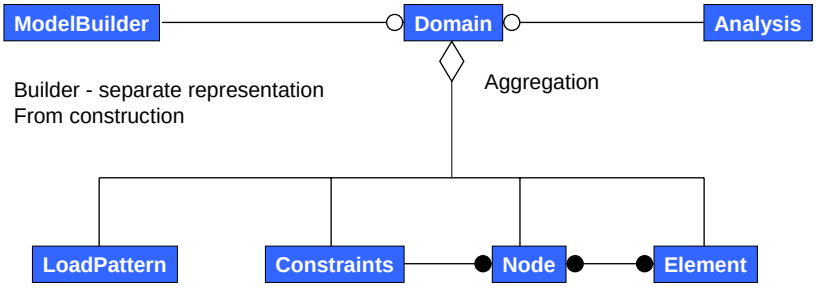
OpenSees — Open System for Earthquake Engineering Simulation
Pacific Earthquake Engineering Research Center

http://opensees.berkeley.edu

- OpenSees has been under development by PEER since before 1997.
- Core group of developers and users.
- PEER will continue research and development for PBEE applications in OpenSees.
- Copyrighted by UC Regents and free for use.
- Google search hits as proxy for interest (5/4/04):
  - Drain[2-3]d-97
  - OpenSees-6020
  - ABAQUS-17,000
  - SAP2000-19,000

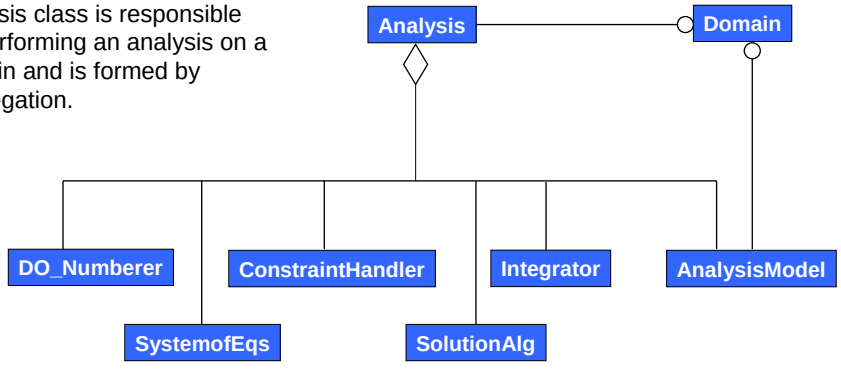


Structural Models as Aggregation Pattern

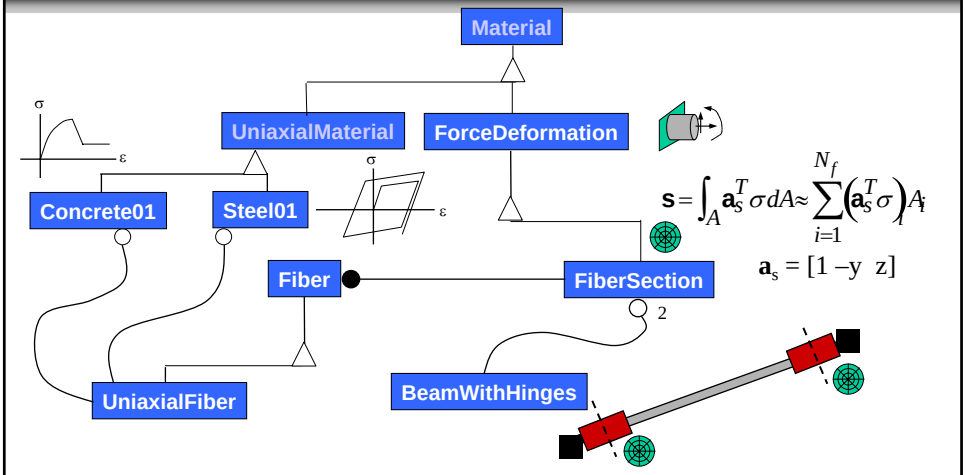


Analysis Class for Simulation

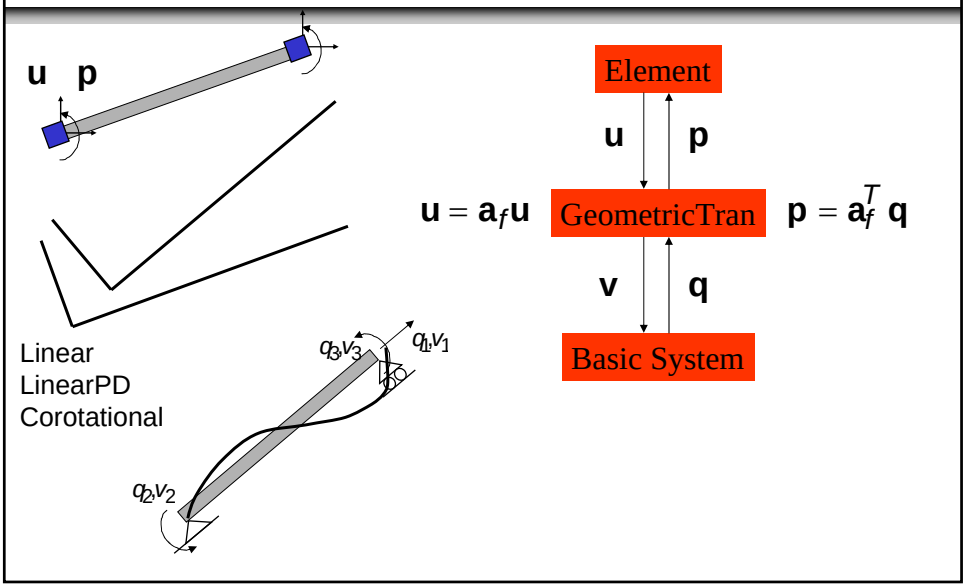
Analysis class is responsible for performing an analysis on a domain and is formed by Aggregation.



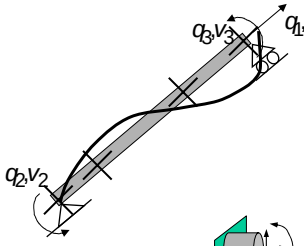
Form Follows Mechanics



Beam-Column Models I



Beam-Column Models II

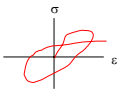


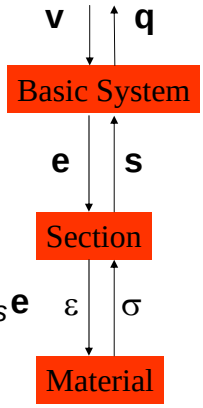
Displacement

$$\mathbf{e} = \mathbf{a}\mathbf{v}$$

$$\mathbf{q} = \int_0^L \mathbf{a}^T \mathbf{s}\, dx$$

e, s





Force

$$\mathbf{s} = \mathbf{b}\mathbf{q}$$

$$\mathbf{v} = \int_0^L \mathbf{b}^T \mathbf{e}\, dx$$

$$\varepsilon = \mathbf{a}_S \mathbf{e}$$

$$\mathbf{s} = \int_A \mathbf{a}_S^T \sigma\, dA$$

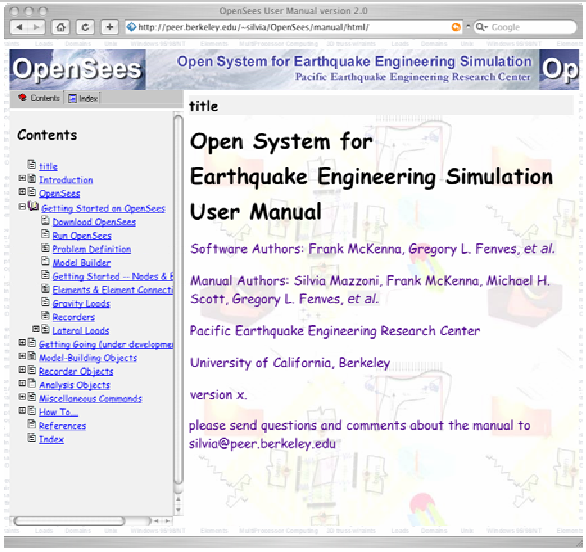
$$\sigma = \sigma(\varepsilon)$$

No assumptions are made on section or material behavior; each level in the hierarchy can be defined independently of other levels

Scripting Models

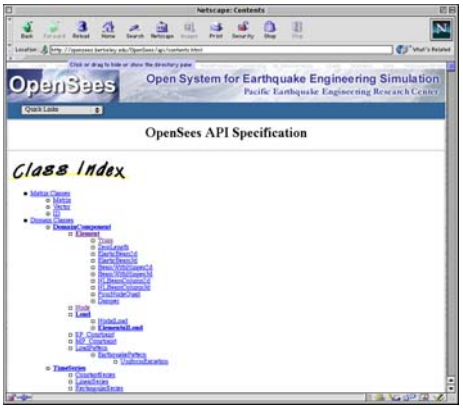
```
1.  wipe
2.  source Units.tcl;                          # define units
3.  source ParamList.tcl;                      # load up parameter values
4.  source GMFiles.tcl;                        # load up ground-motion filenames

5.  foreach Xframe $iXframe Hcol $iHcol Lcol $iLcol Lbeam $iLbeam        FRAME
                GIblc $iGIblc GrhoCol $iGrhoCo GPcol $iGPcol GMfact $iGMfact {
6.      source Static.tcl; # load procedure for static analysis
7.      source Dynamic.tcl;            # load procedure for dynamic analysis
8.      puts FRAME$Xframe.....FRAME$Xframe.....
9.      puts STATIC_ANALYSIS
10.     Static $Xframe $Hcol $Lcol $Lbeam $GIblc $GrhoCol $GPcol $GMfact ;
11.     puts DYNAMIC_ANALYSIS
12.     foreach GroundFile $iGroundFile {                    GROUND MOTION
13.         puts GroundMotion$GroundFile
14.         Dynamic $Xframe $Hcol $Lcol $Lbeam $GIblc $GrhoCol $GPcol $GMfact $GroundFile;
15.     }
16. }
```

Online User Guide

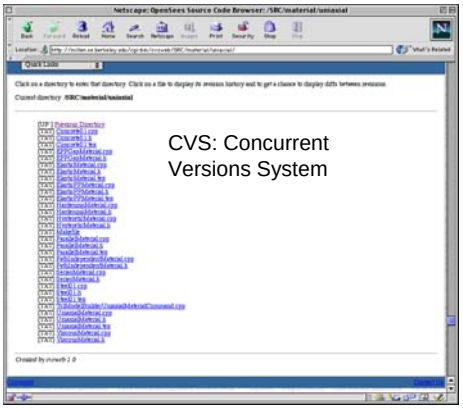


Framework Design/Source for Developers

Class Specification Application Program Interface



Source Code Viewing/Updating With CVS

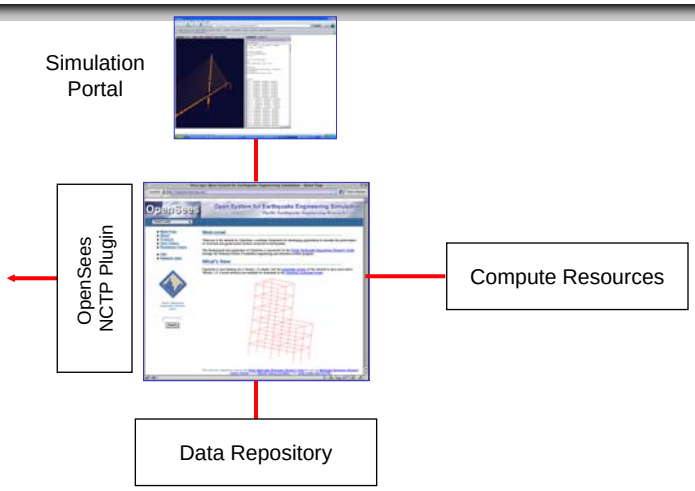


CVS: Concurrent Versions System

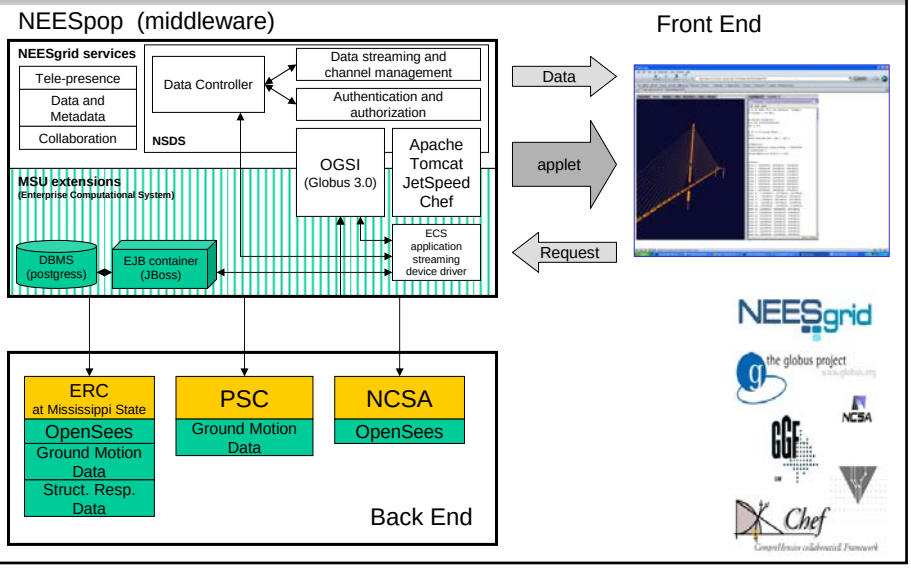
PEER Testbed Project
Applications of OpenSees

- All four testbeds utilized OpenSees
- The testbeds pushed the state-of-the-art in nonlinear modeling and presented challenging simulation problems
- Testbeds were used to:
  - Validate models
  - Investigate convergence and computational performance
  - Support PEER framework, including reliability computation
  - Identify improvements in models and simulation methods

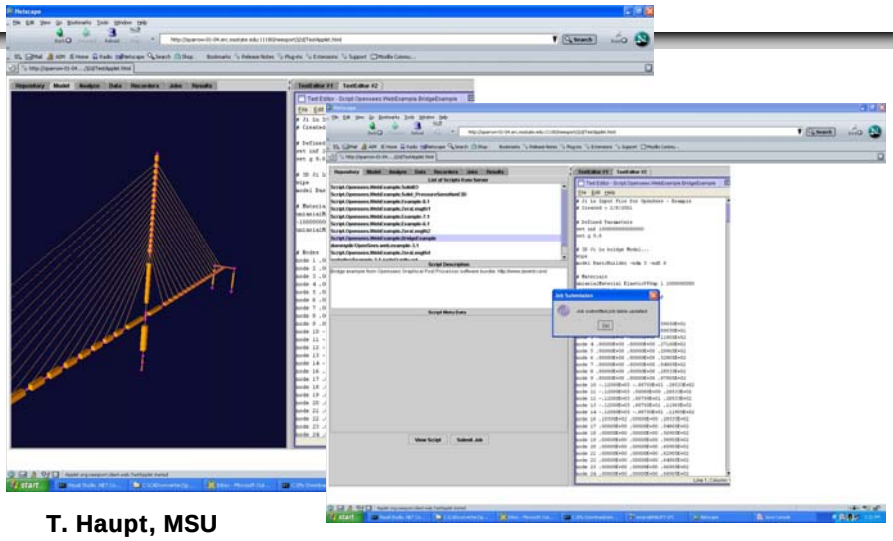
NEESgrid Simulation Overview



Portal Architecture



NEESgrid OpenSees Web Portal



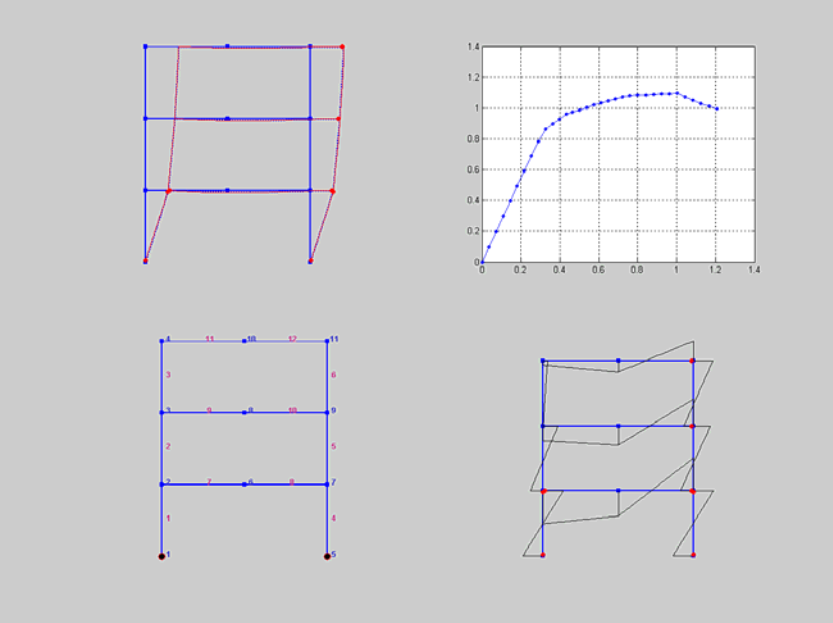
T. Haupt, MSU with PEER

Fedeas[Mat]Lab Toolbox for Structural Simulation

- Ease of developing new solution strategies, elements and material models.
- Ability to add new capabilities (e.g. pre- and post-processing, visualization).
- Efficient for concept development and validation, interface with controllers and DAQ for experimental-computational integration.
- Access to vast array of Matlab toolboxes.

Relation Between FedeasLab and OpenSees

- FedeasLab is easy to learn and use:
  - Concept development and validation time is short
  - No need to learn a sophisticated computer language: non-specialists are welcome to participate
  - Can tap to the vast array of available Matlab functions
- Scripting is built into Matlab; OpenSees uses Tcl as one scripting language
- Matlab is an interpreted language, therefore FedeasLab is not well suited for large-scale applications.
- FedeasLab and OpenSees are based on modular programming
  - FedeasLab libraries may be compiled with Matlab Compiler and integrated to OpenSees
  - Integration of C or C++ code and Matlab functions may be directly possible for inter-operability



Objective of Workshop

- Describe modeling and analysis capability, including hierarchy of system, element, section, material
- Overview of applications, structural and geotechnical
- Show specific examples of nonlinear analysis
- Introduce NEESgrid web portal for OpenSees
- Introduce FEDEASLab
- Provide hands-on starting-point for simulation tools

Organization of Workshop

- Tcl command language; what is Tcl?
- Modeling commands
  - System, element, section, material hierarchy
- Analysis commands
  - Standard and advanced methods
- Basic examples and use of parameters
- Advanced structural and geotechnical applications
- NEESgrid OpenSees portal
- FEDEASLab introduction

What Should be Your Expectations?

- OpenSees is a research tool at this time, but fairly stable for regular use
- As with any nonlinear analysis, it requires careful consideration of model and interpretation of results
- It is under continual development by students, faculty and other researchers
- User interface development lags behind computational technology
- It is not bullet-proof
- An investment of time and learning is required
- The OpenSees *open-source community* requires give and take to be successful

Use the Community Forum!

Bookmark the Message Board, opensees.berkeley.edu/phpBB2/index.php and use it often.

Thanks to:

- **Silvia Mazzoni**
- **Frank McKenna**
- Veronica Padilla
- Ingrid Berg
- Parshaw Vaziri
- Boris Jeremic
- Filip Filippou
- Tomasz Haupt, MSU
- Bill Spencer, UIUC
- … all the speakers

and the National Science Foundation