OpenSees, VEES, and XML: Visualization and Model Archiving

Alisa Neeman, Boris Jeremić, Alex Pang, Adam Markowitz

University of California, Santa Cruz and University of California, Davis

8/29/2006
What Does Geomechanics Need?

- **Run-time** visualization
- **New** techniques
- **Iterate** on a model
- **Design**, **Visualize**, **Simulate**
- **Archive** of models

8/29/2006
XML is like money - it works best if you pass it around.

Set Material Properties, Boundary Conditions

Simulate

Visualize

parse

mesh
We’re Not Starting from Zero:

- Object-Oriented C++/Tcl FEM simulator
  - Data types organized in **natural hierarchy**
  - Runs on Linux and Windows
- All the data we might want to visualize:
  - Stress, strain, displacement, acceleration, elements, materials
XML Schema follows OpenSees Hierarchy

OpenSees

ID type

Node id=2

Crd1  Crd2  Crd3  ndof

Element id=3

EightNode Brick

Material id=5

NDMaterial
XML parsing code creates a running OpenSees simulation

How about the 687 other OpenSees classes???
Code that writes code!!

- Java code (XMLFactoryBuilder.java) parses a C++ constructor and creates:
  - XML schema entry
  - C++ factory class
  - OpenSees printXMLModel function
That’s all good, but what about the visualization?
Software Reuse aids visualization

OpenSees **full** of helpful data structures:
- Graph, elements, nodes, Iterator!

Scalar, vector, tensor data **right at hand**

Mix & Match: individual elements visualized using different techniques
- Stress colored by volumetric or distortional change
- Probabilistic damage measure glyphs
- Thresholding to find interesting data, assess accuracy of simulation
- Displaced mesh
VEES

Draw methods as objects
   Each finite element assigned draw object
      Just a few different geometries
          (2 node, 4 node, 8 node…)
Easy to add new types
VEES also rides directly on OpenSees Tcl interpreter
Info Viz approach

1. Overview (entire domain)
2. Zoom and Filter
3. Details on Demand
Filter by color/value and invert
Filter by element type - list automatically generated
Where we are today

XML
- Basic framework in place
- Lots of non-research manpower needed to complete XML but automation should help a lot

VEES
- Basic framework in place
- GUI coming along
- details-on-demand will be part of thesis research
Thanks!

Working pre-alpha release can be found at NEESforge: VEES project

Help us help you! The wish list is long and we need to prioritize.

http://neesforge.nees.org/projects/vees/

This work funded by a GAANN fellowship