OpenSees Model-Building Commands II

Silvia Mazzoni

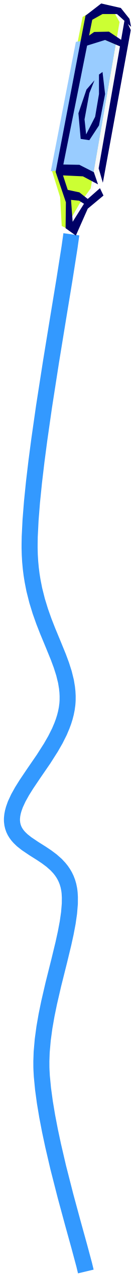
University of California, Berkeley

OpenSees User Workshop

14 August 2006

Recorder Objects

- Node Recorder
- EnvelopeNode Recorder
- MaxNodeDisp Recorder
- Drift Recorder
- Element Recorder
- EnvelopeElement Recorder
- Display Recorder
- Plot Recorder
- playback Command



```
recorder Node <-file $fileName> <-time> <-node ($node1
$node2 ...)> <-nodeRange $startNode $endNode> <-region
$RegionTag> <-node all> -dof ($dof1 $dof2 ...) $respType
```

$fileName file where results are stored. Each line of the file contains the result for a committed state of the domain *(optional, default: screen output)*

-time this argument will place the pseudo time of the as the first entry in the line. *(optional, default: omitted)*

$node1 $node2 ... tags nodes where response is being recorded -- select nodes in domain *(optional, default: all)*

$startNode $endNode tag for start and end nodes where response is being recorded -- range of nodes in domain *(optional, default: all)*

$RegionTag tag for previously-defined selection of nodes defined using the Region command. *(optional)*

all where response is being recorded -- all nodes in domain *(optional & default)*

$dof1 $dof2 ... degrees of freedom of response being recorded.
Valid range is from 1 through ndf , the number of nodal degrees-of-freedom.

$respType defines response type to be recorded. The following response types are available:

disp	displacement
vel	velocity
accel	acceleration
incrDisp	incremental displacement
eigen	eigenvector

```
recorder EnvelopeNode <-file $fileName> <-time> <-node ($node1 $node2 ...)> <-nodeRange $startNode $endNode> <-region $RegionTag> <-node all> -dof ($dof1 $dof2 ...) $respType
```

records the envelope of displacement, velocity, acceleration and incremental displacement at the nodes (translational & rotational). The envelope consists of the following: minimum, maximum and maximum absolute value of specified response type

$fileName file where results are stored. Each line of the file contains the result for a committed state of the domain *(optional, default: screen output)*

-time this argument will place the pseudo time of the as the first entry in the line. *(optional, default: omitted)*

$node1 $node2 ... tags nodes where response is being recorded -- select nodes in domain *(optional, default: all)*

$startNode $endNode tag for start and end nodes where response is being recorded -- range of nodes in domain *(optional, default: all)*

$RegionTag tag for previously-defined selection of nodes defined using the Region command. *(optional)*

all where response is being recorded -- all nodes in domain *(optional & default)*

$dof1 $dof2 ... degrees of freedom of response being recorded.
Valid range is from 1 through ndf , the number of nodal degrees-of-freedom.

….. same arguments as node recorder

recorder MaxNodeDisp $dof $node1 $node2 ...

records the values of the maximum absolute values of the displacement in the prescribed direction of a prescribed set of nodes

$dof displacement degree-of-freedom direction. Valid range is from 1 through ndf, the number of nodal degrees-of-freedom.

$node1 $node2 ... nodes where maximum displacement is being recorded

recorder Element <-file $fileName> <-time> <-ele ($ele1 $ele2 ...)> <-eleRange $startEle $endEle> <-region $regTag> <-ele all> ($arg1 $arg2 ...)

$fileName file where results are stored. Each line of the file contains the result for a committed state of the domain *(optional, default: screen output)*

-time this argument will place the pseudo time of the as the first entry in the line. *(optional, default: omitted)*

$ele1 $ele2 ... tags of elements whose response is being recorded -- selected elements in domain *(optional, default: omitted)*

$startEle $endEle tag for start and end elements whose response is being recorded -- range of selected elements in domain *(optional, default: all)*

$regTag previously-defined tag of region of elements whose response is being recorded -- region of elements in domain *(optional)*

all elements whose response is being recorded -- all elements in domain *(optional & default)*

$arg1 $arg2 ... arguments which are passed to the setResponse() element method

element recorder (output arguments)

All:

globalForce – element resisting force in global coordinates (does not include inertial forces)

recorder Element -file ele1global.out -time -ele 1 globalForce

localForce – element resisting force in local coordinates (does not include inertial forces)

recorder Element -file ele1local.out -time -ele 1 localForce

Section:

section $secNum – request response quantities from a specific section along the element length

$secNum refers to the integration point whose data is to be output

force – section forces

example: recorder Element -file ele1sec1Force.out –time -ele 1 section 1 force

deformation – section deformations

example: recorder Element -file ele1sec1Force.out –time -ele 1 section 1 deformation

stiffness – section stiffness

example: recorder Element -file ele1sec1Force.out –time -ele 1 section 1 stiffness

stressStrain – record stress-strain response.

example: recorder Element -file ele1sec1Force.out –time -ele 1 section 1 fiber $y $z stressStrain

$y	local y coordinate of fiber to be monitored*
$z	local z coordinate of fiber to be monitored*

output.tcl

```
# Record nodal displacements -NODAL DISPLACEMENTS
# ALL displacements at node 1
   recorder Node -file Dnode1.out -time -node 1 -dof 1 2 3 disp;

# Record vertical-y displacement of ALL nodes
   recorder Node -file DNodeALL.out -time -node all -dof 2 disp;

# Record REACTION FORCES - (=forces in element 1)
   recorder Element -file Fel1.out -time -ele 1 localForce
```

Loads - pattern command

```
pattern Plain $patternTag (TimeSeriesType arguments) {
        load (load-command arguments)
        sp (sp-command arguments)
        eleLoad (eleLoad-command arguments)
}
```

$patternTag unique pattern object tag

TimeSeriesType arguments list which is parsed to construct the TimeSeries object associated with the LoadPattern object.

load ... list of commands to construct nodal loads -- the NodalLoad object

sp ... list of commands to construct single-point constraints -- the SP_Constraint object

eleLoad ... list of commands to construct element loads -- the eleLoad object

pattern command (cont.)

load $nodeTag (ndf $LoadValues)

$nodeTag	node on which loads act
$LoadValues	load values that are to be applied to the node. Valid range is from 1 through ndf, the number of nodal degrees-of-freedom.

sp $nodeTag $DOFtag $DOFvalue

$nodeTag	node on which the single-point constraint acts
$DOFtag	degree-of-freedom at the node being constrained. Valid range is from 1 through ndf, the number of nodal degrees-of-freedom.
$DOFvalue	reference value of the constraint to be applied to the DOF at the node.

```
pattern Plain 1 Linear {
     load 3                0.0 -$Pdl 0.0 0.0 0.0   -$Mdl
     load 4                0.0 -$Pdl 0.0 0.0 0.0   +$Mdl
     sp    1 2 -0.001
     eleLoad -ele 3 -type -beamUniform [expr -$Weight/LBeam]
 }
```

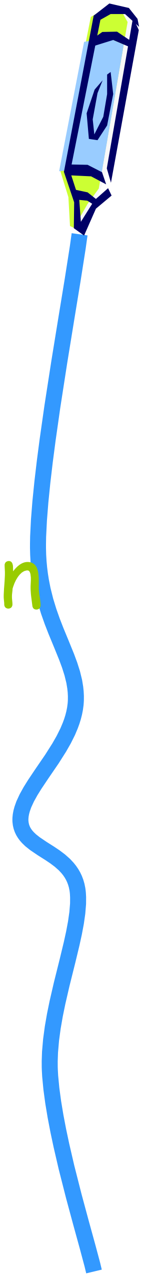
Questions,
or statements!

The OpenSees Community Forum:

http://opensees.berkeley.edu/community/index.php

which can be accessed from:

http://opensees.berkeley.edu



thank you!!!