



DataVolumePanel

constructor:

```
pos_event_gateway= new PositionGateway()
finish_initialization():
  coord_fields= new $CoordFields(..)
  slice_vports= {new XXSlice2DViewport(..),...}
  listeners= slice_producers
    + slice_vports + coord_fields + pos_event_gateway
  gen's = slice_vports + coord_fields + pos_event_gateway
  2DViewport.event_listeners(i..j)= listeners
  $CoordFields.event_listeners(i..j)= listeners
  PosGate.internal_listeners(i..j)= listeners
positionChangeDetected(pe):
  pos_event_gateway.positionChangeDetected_External(pe)
public void addPositionListener( pl, send_event):
  pos_event_gateway.addPositionListener_External( pl)
```

3) DataVolumePanel\$CoordFields

.event_listeners();

constructor:

```
set: world_cursor, world_cursor_nat, atlas_coords,
  voxel_cursor, voxel_cursor_nat, atlas_label
XX_textfield_MNI/NAT/ATLAS.addActionListener()
positionChangeDetected(new PositionEvent(..))
```

actionPerformed(ae): (text[internal]_positionChangeDetect)

```
get: XX_cursor [XX]
voxel_cursor.XX= new_text; pos_event_mask= ..
_firePositionEvent(new pe(..))
```

_firePositionEvent(pe):

```
coord_fields.{event_listeners}.positionChangeDetected(pe)
```

▶ positionChangeDetected(pe): (mouse&final_positionChanged)

```
get: XX_cursor [XX]
```

update: world_cursor.XX with pe.getXX() and world_cursor_nat with pe.getXX_nat()

resync: XX_cursor [XX] not world_nat

update: XX_textfield_MNI/NAI/ATLAS

4) XXSlice2DViewport

```
constructor: super(..)
positionChangeDetected(pe)
  cursor.rcs=pe.xyz*
  _newCursor(r,c,false)
  _firePositionEvent(local_mask)
  mask=local_mask* or ALL ifNa
  pe.xyz=r*c*
  _aid_to_firePositionEvent(pe)
```

2) XXSlicedImageProducer

```
constructor: super(..)
positionChangeDetected(pe)
  world2voxel(pe.xyz)
  crt_slice=vz
```

2B) SlicedImageProducer

```
.crt_slice;
constructor: super(..)
```

1) PositionGateway

```
.internal_listeners(); <-event_listeners
.external_listeners(); <-other DataVolumePanels
constructor:
```

positionChangeDetected(pe): (internal_to_ext)

```
external_listeners.positionChangeDetected(pe)
```

positionChangeDetected_External(pe): (ext_to_internal)

```
internal_listeners.positionChangeDetected(pe)
```

4B) Slice2DViewport

.event_listeners();

cursor; <-world, but locally rotated, so rcs (row,col,slice)

constructor: _updateVportCursorPosition()

_new_cursor(vp_r, vp_c) <- viewport rc, not world rcs

```
voxr,voxc= _vport_to_voxel(vp_r, vp_c)
```

```
x_r,y_c=voxel2world(voxr,voxc)
```

```
_newCursor(x_r,y_c,true)
```

_newCursor(r,c,notify) <- world_rotated, so rcs

```
cursor.rc= r,c
```

```
_updateVportCursorPosition()
```

```
_firePositionEvent(XYmask)
```

_newSlice(s_inc)

```
cursor.s++s_inc
```

```
_firePositionEvent(Zmask)
```

_aid_to_firePositionEvent(pe)

```
{event_listeners}.positionChangeDetected(pe)
```

_update_VportCursorPosition()

```
r,c= _worldLocal2vport(cursor.rc)
```

```
vport_cursor.setPosition(r,c,slice)
```

DataVolumePanel

constructor:
..
positionChangeDetected(pe):
 pos_event_gateway.positionChangeDetected_External(pe)

DataVolumePanel\$CoordFields

constructor:
..
actionPerformed(ae): (text[internal]_positionChangeDetect)
 _firePositionEvent(new pe(..))
 _firePositionEvent(pe): add mask
 coord_fields.{event_listeners}.positionChangeDetected(e)
 positionChangeDetected(pe): (mouse&final_positionChangeDet)

XXSlicedImageProducer

constructor:
..
positionChangeDetected(pe):
 ..

XXSlice2DViewport

.event_listeners();
constructor:
..
positionChangeDetected(pe):
 ..

PositionGateway

.internal_listeners();
.external_listeners();
constructor:
..
positionChangeDetected(pe):
 external_listeners.positionChangeDetected(pe)
positionChangeDetected_External(pe):
 internal_listeners.positionChangeDetected(pe)

DataVolumePanel #2

positionChangeDetected(pe)

DataVolumePanel\$CoordFields #2

positionChangeDetected(pe)

XXSlicedImageProducer #2

positionChangeDetected(pe)

XXSlice2DViewport #2

positionChangeDetected(pe)

PositionGateway #2

positionChangeDetected(pe)
positionChangeDetected_External(pe)

DataVolumePanel

constructor:

```
..
positionChangeDetected(pe):
  pos_event_gateway.positionChangeDetected_External(pe)
```

DataVolumePanel\$CoordFields

constructor:

```
..
actionPerformed(ae): (text[internal]_positionChangeDetect)
..
_firePositionEvent(pe):
..
positionChangeDetected(pe): (mouse&final_positionChangeDet)
```

XXSliceImageProducer

constructor: super(..)
positionChangeDetected(pe)

SliceImageProducer

.crt_slice;
constructor: super(..)

XXSlice2DViewport

constructor: super(..)
positionChangeDetected(pe)
_newCursor(r,c,false)
_firePositionEvent(local_mask) mni OR native
_aid_to_firePositionEvent(pe) mni & native

Slice2DViewport

```
.event_listeners();
cursor; <-world, but locally rotated, so rcs (row,col,slice)
constructor: _updateVportCursorPosition()
_new_cursor(vp_r,vp_c)<- viewport rc, not world rcs mni OR native
_newCursor(x_r,y_c,true)
newCursor(r,c,notify)<- world_rotated, so rcs mni OR native
_updateVportCursorPosition()
_firePositionEvent(XYmask)
_newSlice(s_inc)
_firePositionEvent(Zmask)
_aid_to_firePositionEvent(pe) add to mask
{event_listeners}.positionChangeDetected(pe)
_update_VportCursorPosition()
vport_cursor.setPosition(r,c,slice)
```

PositionGateway

```
.internal_listeners();
.external_listeners();
constructor:
..
positionChangeDetected(pe):
  external_listeners.positionChangeDetected(pe)
positionChangeDetected_External(pe):
  internal_listeners.positionChangeDetected(pe)
```

DataVolumePanel #2

positionChangeDetected(pe)

DataVolumePanel\$CoordFields #2

positionChangeDetected(pe)

XXSliceImageProducer #2

positionChangeDetected(pe)

XXSlice2DViewport #2

positionChangeDetected(pe)

PositionGateway #2

```
positionChangeDetected(pe)
positionChangeDetected_External(pe)
```