

# Sections and cutaways

## Why we need an improved Boolean Processor

John Allison

University of Manchester

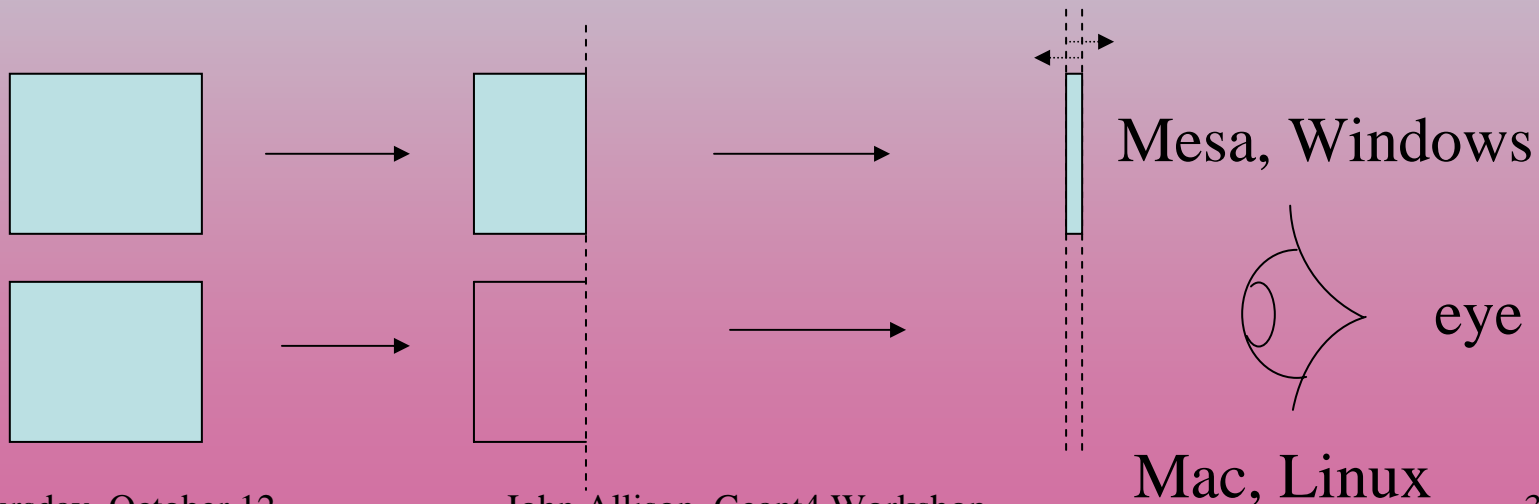
# DAWNCUT

- DAWNCUT is available as ever
  - Make a .prim file with DAWNFILE, then invoke dawncut, then view with dawn
- Make a section by successive application
  - `dawncut 0 1 0 1 g4_00.prim g4_00_0.prim`  
`dawncut 0 -1 0 1 g4_00_0.prim g4_00_1.prim`  
`dawn g4_00_1.prim`
- Produces closed shapes
  - (but sometimes nice to see inside in surface mode)

# Sections (DCUT)

- `/vis/viewer/set/sectionPlane` only ever worked on OpenGL (Mesa)
- Bad news: it doesn't work on today's OpenGL (at least Mac and Linux; OK on Windows)
  - OpenGL's clip plane clips too much

Single clip plane    Back-to-back clip planes



# Cutaways

- [DAWNCUT](#) is available as ever
  - but cannot see inside in surface mode
- Good news: the aforementioned change of behaviour means
  - we can implement “see-inside” cutaways in OpenGL (Mac and Linux)

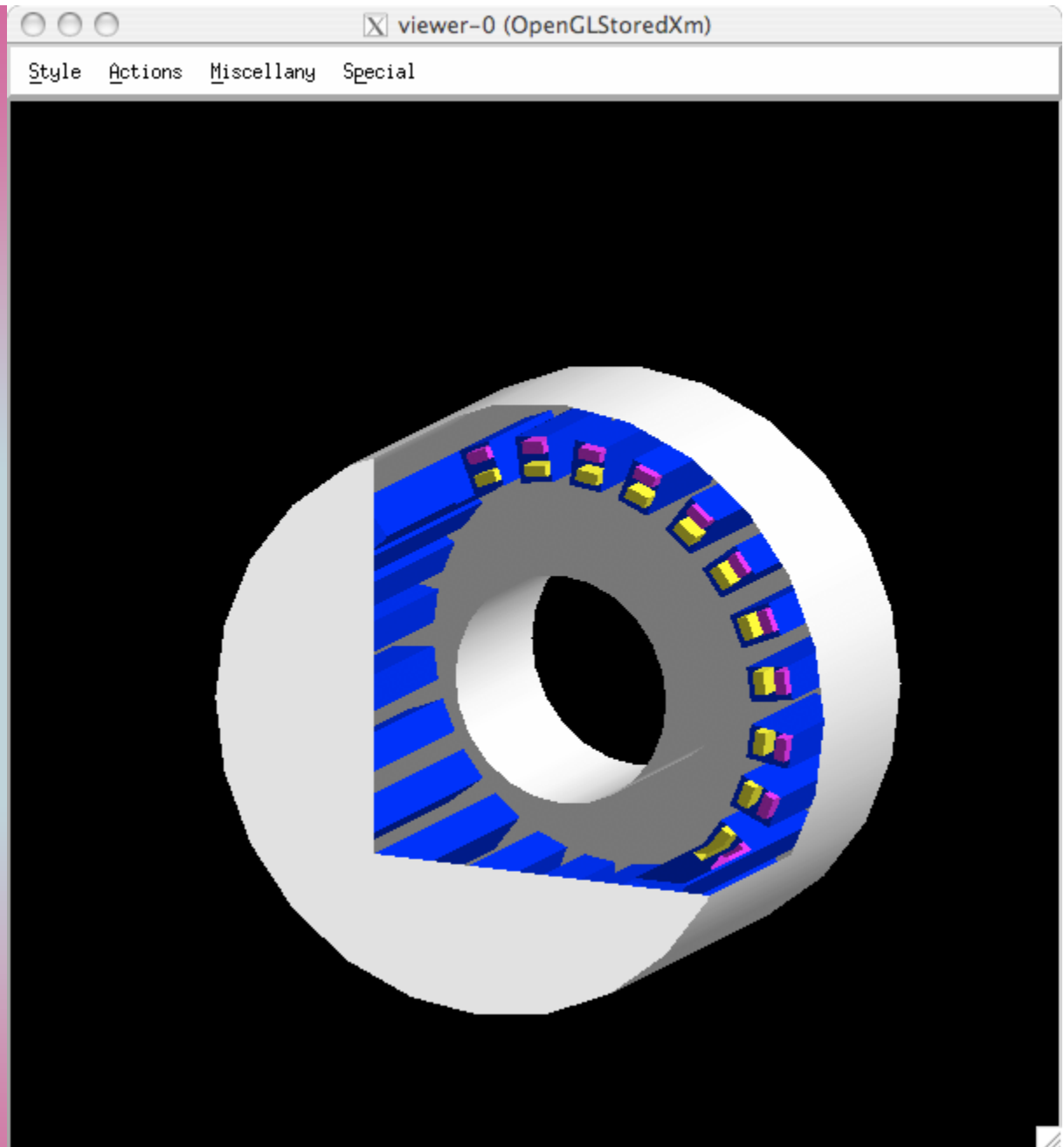
# New commands

- New commands control up to 3 cutaway planes
  - `/vis/viewer/addCutawayPlane`  
`/vis/viewer/changeCutawayPlane`  
`/vis/viewer/clearCutawayPlanes`  
`/vis/viewer/set/cutawayMode`  
`union|intersection`
  - Implemented only by OpenGL driver at present

Union of three  
cutaways

OpenGL  
immediate and  
stored mode. Fast  
interaction in  
stored Xm mode.

Can animate — see  
commands on next  
slide



# Animation commands

```
/control/verbose 2
/vis/verbose confirmations
/run/initialize
/vis/open OGLSX 600x600-0+0
/vis/drawVolume tube_phys
/vis/viewer/set/viewpointThetaPhi 30 30
/vis/viewer/set/style surface
/vis/geometry/set/visibility tube_L ! true
/vis/geometry/set/visibility divided_tube_L ! false
/vis/geometry/set/visibility sub_divided_tube_L ! false
/vis/geometry/set/colour daughter_box_L ! blue
/vis/viewer/addCutawayPlane -220 0 0 cm -1 0 0
/vis/viewer/addCutawayPlane 0 -24 0 cm 0 -1 0
/vis/viewer/addCutawayPlane 0 0 -1 cm 0 0 -1
#/vis/oglx/set/printEPS true
/control/verbose 0
/vis/verbose quiet
/control/loop cutaway.loop x -150 -250 -0.1
```

where cutaway.loop is

```
/vis/viewer/changeCutawayPlane 0 {x} 0 0 cm -1 0 0
/vis/viewer/zoom 1.001
/vis/viewer/update
```

## Cutaway movie

```
(/vis/oglx/set/printEPS true)
```

Shown here: mpeg4  
movie

[G4OpenGL\\_cutaway.mp4.mpg](#)  
or [G4OpenGL\\_cutaway.mpg](#)

QuickTime™ and a  
decompressor  
are needed to see this picture.

See parallel session  
presentation: [Making  
a movie](#)



(Another new  
command)

```
/vis/viewer/set/explode
```

```
Factor
```

```
G4OpenGL\_explode.mp4.mpg or
```

```
G4OpenGL\_explode.mpg
```

Animation:

```
/control/loop explode.loop f 1
```

```
2 0.001
```

where `explode.loop` is

```
/vis/viewer/set/
```

```
explodeFactor
```

```
{f} -200 0 0 cm
```

QuickTime™ and a  
decompressor  
are needed to see this picture.

# Boolean Processor

- Proposal: use to implement generic sections and cutaways
  - Extend to all drivers
  - Insulate from OpenGL ambiguities
- Operates on G4 Polyhedron objects available in principle for each Geant4 shape
  - Currently doesn't do cutaways
  - Not all shapes implemented
  - Other problems...

# Current problems

- Pathological cases cause failure and/or incorrect polyhedron objects
  - Subtraction of polyhedra with coincident or nearly coincident faces
  - Cascaded operations (Booleans of Booleans)
- Continuing user problem reports
  - Signs of user fatigue
- Sections and cutaways particularly troublesome
  - Prototype use code disabled

# Proposal

- [G4Polyhedron Discussion Document](#)
  - John Allison and Evgueni Tchernaiiev
  - October 2005!!
  - Needs your support
- Proposal
  - Reimplement existing functionality, including
    - Improved robustness
    - Caching of normals (for speed)
  - Offer the option of user-supplied normals (for space saving and speed)
  - Add a new Boolean operation – cut – that creates open polyhedra for cutaway views
  - Complete the implementation of polyhedron representations of all Geant4 shapes