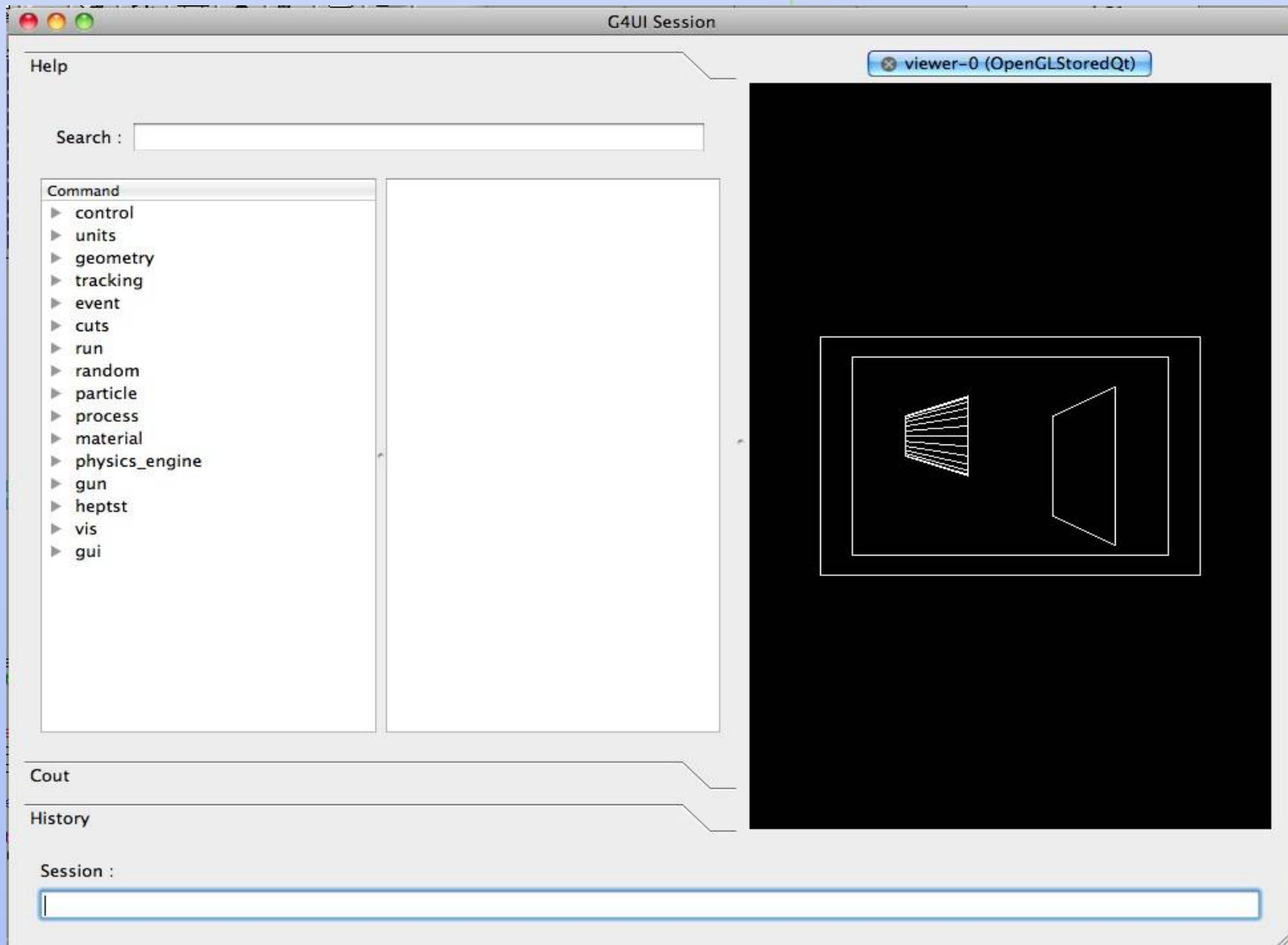




Updates to Qt Visualization

Laurent Garnier
On behalf of Visualisation and interfaces WG

Qt Visualisation Driver – geant4-09-05 version -



Qt Visualisation Driver – geant4-09-06 version ? -

Qt driver is originally in two parts :

- User interface in « interface » package
- Viewers in « visualization » package

This modularity allow users to define their own user interface or to integrate only viewer in their own program.

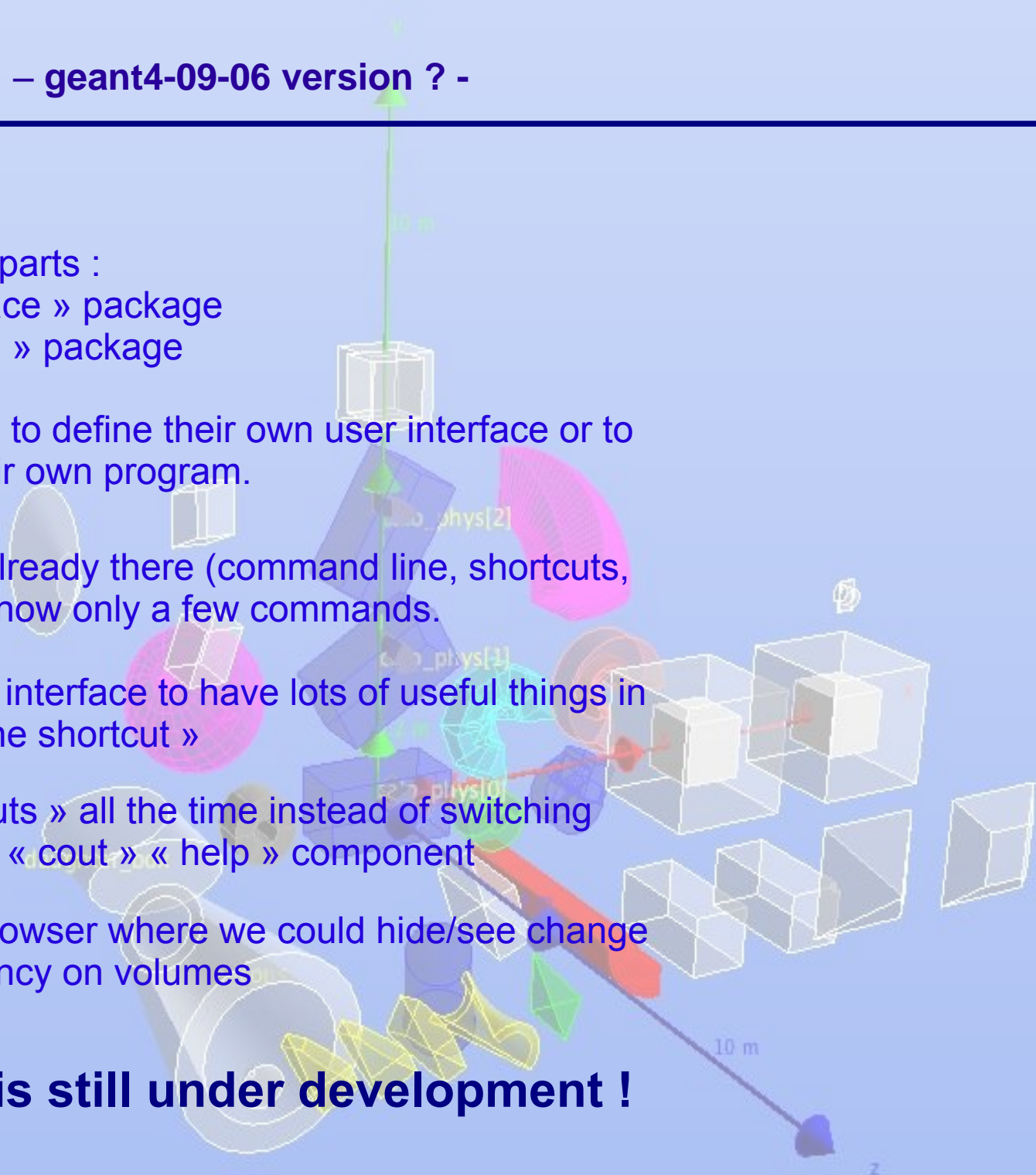
Lots of functionalities are already there (command line, shortcuts, context menu), but users know only a few commands.

⇒ Try to redefine user interface to have lots of useful things in « one click » or « one shortcut »

⇒ Allow to see « outputs » all the time instead of switching between « history » « cout » « help » component

⇒ Add a scene tree browser where we could hide/see change color and transparency on volumes

Everything is still under development !



Qt Visualisation Driver – geant4-09-06 version ? -

The screenshot displays the G4UI Session interface. The main window, titled "viewer-0 (OpenGLStoredQt)", shows a 3D visualization of two shapes: a white cone labeled "Shape1" and a grey rectangular prism labeled "Shape2". A coordinate system with x, y, and z axes is visible, with "10 cm" markings on the y and z axes. The text "Mon Sep 10 15:35:00 2012" is displayed in the top right corner of the viewer. The "Geant4" logo is shown in orange at the bottom left, and "exampleB1" is shown in green at the bottom right. A small 3D wireframe box is visible on the right side of the viewer.

The left sidebar contains a "viewer" tab with a "viewer-0 (OpenGLStoredQt)" sub-tab. Below the tabs is a list of objects with checkboxes:

- Axes
- Date
- Frame
- Logo
- Logo2D
- Scale
- Text
- Text2D
- Touchables

Below the list is a "Touchables slider" with a "Show all" button on the left and a "Hide all" button on the right. A search field with a "select item(s)" button is located below the slider.

The bottom left of the interface shows a "Session :" label and an empty text input field.

The bottom right of the interface shows an "Output" window with the following text:

```
/vis/viewer/set/style surrface
/vis/viewer/set/hiddenMarker true
/vis/viewer/set/viewpointThetaPhi 120 150
#
# Re-establish auto refreshing and verbosity:
/vis/viewer/set/autoRefresh true
/vis/viewer/refresh
```

Below the output window is a "clear" button and a "Filter :" label followed by an empty text input field.

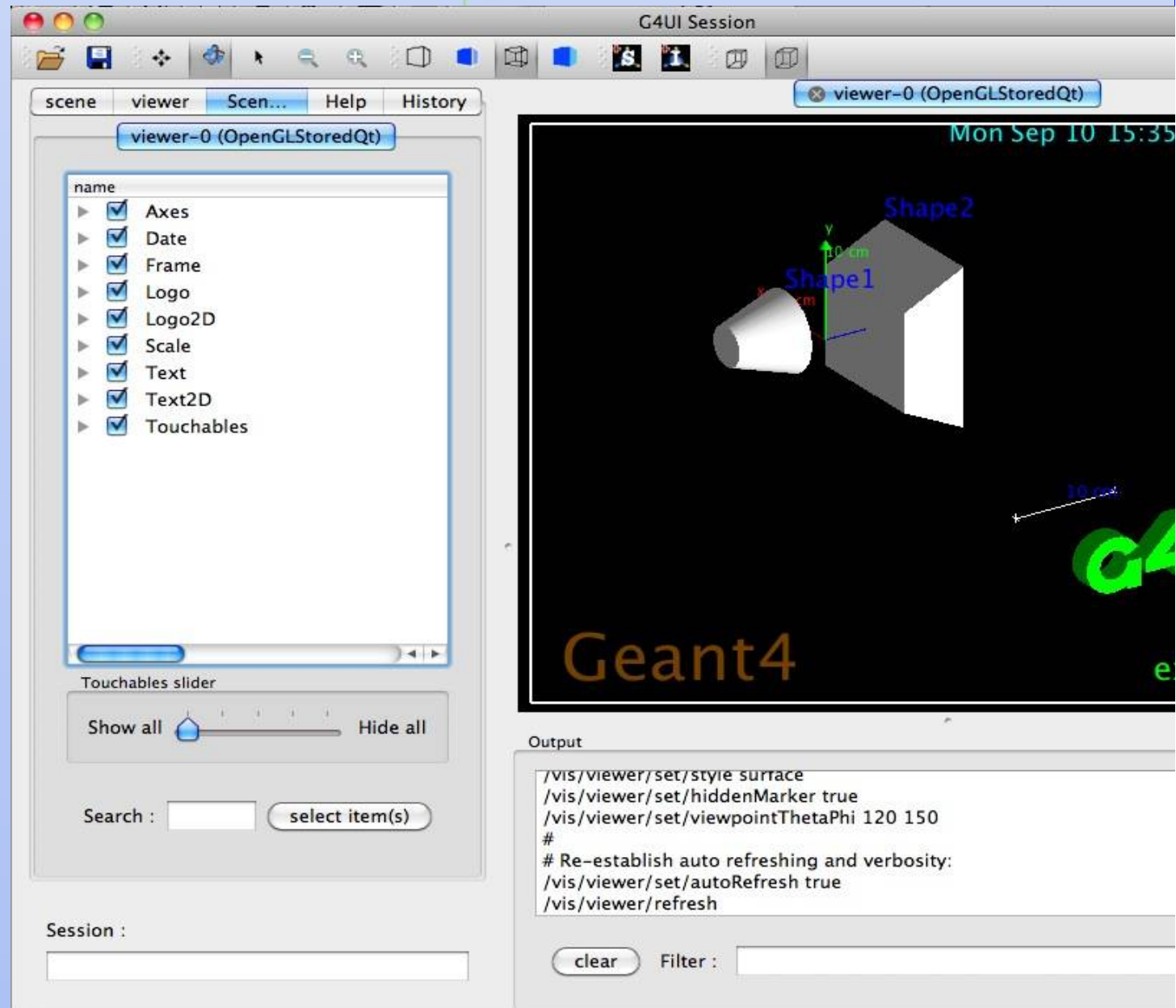
Qt Visualisation Driver – geant4-09-06 version ? -

What's new ?

- Icon toolbar
- Left widget tabstack
- Scene tree browser

All previous components are still there :

- Help widget
- Output window
- History
- Viewers
- Session



Qt Visualisation Driver – Test202 example -

The screenshot displays the G4UI Session application window. The title bar reads "G4UI Session". The interface is divided into several sections:

- Menu Bar:** Includes "scene", "viewer", "Scen...", "Help", and "History".
- Viewer Tab:** Labeled "viewer-0 (OpenGLStoredQt)".
- Left Panel:** A tree view under "name" with the following items checked:
 - ▶ Axes
 - ▶ Date
 - ▶ Frame
 - ▶ Text
 - ▶ Touchables
- Touchables slider:** A slider labeled "Touchables slider" with "Show all" on the left and "Hide all" on the right. The slider is currently positioned near "Show all".
- Search:** A search box with the text "Search : " and a "select item(s)" button.
- Session:** A text input field labeled "Session :".
- 3D Viewport:** A large blue window showing a 3D visualization of a particle detector simulation. The date "Mon Sep 10 15:38:17 2012" is displayed in the top right corner. The visualization includes various components such as "calo_phys[2]", "calo_phys[1]", "daughter", "phys", and "con".
- Output Panel:** A text area at the bottom right containing the following log messages:

```
G4VisManager::SetCurrentviewer: viewer now viewer-0 (OpenGLStoredQt)
G4VisManager::SetCurrentSceneHandler: scene handler now "scene-handler-0"
/vis/scene/add/text 0 4000 0 mm 12 0 0 calo_phys[2]
Text "calo_phys[2]" has been added to scene "scene-0".
/vis/scene/notifyHandlers
G4VisManager::SetCurrentViewer: viewer now viewer-0 (OpenGLStoredQt)
G4VisManager::SetCurrentSceneHandler: scene handler now "scene-handler-0"
Traversing scene data...
Traversing scene data...
Viewer "viewer-0 (OpenGLStoredQt)" of scene handler "scene-handler-0"
refreshed at request of scene "scene-0".
G4VisManager::SetCurrentViewer: viewer now viewer-0 (OpenGLStoredQt)
```
- Buttons:** A "clear" button and a "Filter:" input field are located at the bottom of the output panel.

Qt Visualisation Driver – Test202 example – scene tree

The screenshot displays the Qt Visualisation Driver interface, titled "G4UI Session". The interface is divided into several sections:

- Scene Tree:** A hierarchical list of objects in the scene. The root is "viewer-0 (OpenGLStoredQt)". Underneath, there are categories like "Axes", "Date", "Frame", "Text", and "Touchableables". The "Touchableables" category is expanded to show "expHall_P [0]", which contains various objects such as "B [0]", "BoxInBox [0]", "C [0]", "PD_physical [0]", "Polycone_Test [0]", "Polyhedra_Test [0]", "Test orb [0]", and several "TestTessCut-phys [0]" and "TestTrapCut-phys [0]" objects. Each item has a checkbox to toggle its visibility.
- Viewer:** A 3D rendering window titled "viewer-0 (OpenGLStoredQt)" showing a complex 3D scene. The scene includes a date stamp "Mon Sep 10 15:38:17 2012" in the top right corner. The 3D view shows various colored objects (spheres, cones, boxes) and text labels like "calo_phys[2]", "calo_phys[1]", "daughter", and "conf".
- Output:** A text area displaying the log of the visualization process. The log shows messages from "G4VisManager" regarding setting the current viewer and scene handler, adding text to the scene, and refreshing the viewer. The text includes: "G4VisManager::SetCurrentviewer: viewer now viewer-0 (OpenGLStoredQt)", "G4VisManager::SetCurrentSceneHandler: scene handler now 'scene-handler-0'", "/vis/scene/add/text 0 4000 0 mm 12 0 0 calo_phys[2]", "Text 'calo_phys[2]' has been added to scene 'scene-0'.", "/vis/scene/notifyHandlers", "G4VisManager::SetCurrentViewer: viewer now viewer-0 (OpenGLStoredQt)", "G4VisManager::SetCurrentSceneHandler: scene handler now 'scene-handler-0'", "Traversing scene data...", "Traversing scene data...", "Viewer 'viewer-0 (OpenGLStoredQt)' of scene handler 'scene-handler-0' refreshed at request of scene 'scene-0'.", "G4VisManager::SetCurrentViewer: viewer now viewer-0 (OpenGLStoredQt)".
- Touchableables slider:** A slider control labeled "Touchableables slider" with "Show all" and "Hide all" buttons.
- Search:** A search input field with a "select item(s)" button.
- Session:** A text input field for the session name.
- Clear and Filter:** A "clear" button and a "Filter:" input field.

Qt Visualisation Driver – Test202 example – changing color and transparency

The screenshot displays the Qt Visualisation Driver interface for a 3D scene. The main window, titled "G4UI Session", shows a 3D visualization of a scene with various objects, including a pink sphere, a blue cube, a red cylinder, and a yellow object. A date stamp "Mon Sep 10 15:38:17 2012" is visible in the top right of the viewer. The left sidebar contains a tree view of the scene's objects, with "PD_physical [0]" selected. Below the tree is a "Touchables slider" set to "Show all". A "Get color a..." dialog box is open in the foreground, showing a color wheel and an "Opacity" slider set to 100%. The bottom of the interface features a console window with log messages and a search filter.

viewer-0 (OpenGLStoredQt)

scene viewer Scen... Help History

viewer-0 (OpenGLStoredQt)

name

- Axes
- Date
- Frame
- Text
- Touchables
 - expHall_P [0]
 - B [0]
 - BoxInBox [0]
 - C [0]
 - PD_physical [0]
 - Polycone_Test [0]
 - Polyhedra_Test [0]
 - Test orb [0]
 - TestTessCut-phys [0]
 - TestTessCut-phys [0]
 - TestTrapCut-phys [0]
 - TestTrapCut-phys [0]

Touchables slider

Show all Hide all

Search : select item(s)

Session :

viewer: viewer now viewer-0 (OpenGLStoredQt)
SceneHandler: scene handler now "scene-handler-0"
00 0 mm 12 0 0 calo_phys[2]
een added to scene "scene-0".
Viewer: viewer now viewer-0 (OpenGLStoredQt)
SceneHandler: scene handler now "scene-handler-0"
Traversing scene data...
Viewer "viewer-0 (OpenGLStoredQt)" of scene handler "scene-handler-0"
refreshed at request of scene "scene-0".
G4VisManager::SetCurrentViewer: viewer now viewer-0 (OpenGLStoredQt)

clear Filter :

Qt Visualisation Driver – Test202 example – touchables slider

The screenshot displays the G4UI Session interface. The main window is titled "viewer-0 (OpenGLStoredQt)" and shows a 3D visualization of a particle detector. The scene contains various geometric objects, including a pink sphere, a blue cylinder, and several rectangular boxes. A red line segment is visible, and a scale bar indicates 10 m. The date and time "Mon Sep 10 15:41:41 2012" are displayed in the top right corner of the viewer.

On the left side, there is a "Touchable slider" control. The slider is currently set to "Show all" and is labeled "Touchable slider". Below the slider, there is a search field and a "select item(s)" button.

The "Output" window at the bottom right displays the following text:

```
drawing style: not forced, auxiliary edge visibility: not forced
line segments per circle: not forced.
time range: (-1.79769e+308,1.79769e+308)
G4AttValue pointer is zero, G4AttDef pointer is zero
/vis/scene/notifyHandlers
NOTE: The scene, "scene-0", of viewer "viewer-0 (OpenGLStoredQt)"
of scene handler "scene-handler-0" has changed. To see effect,
"/vis/viewer/select viewer-0" and "/vis/viewer/rebuild".
G4VisManager::SetCurrentViewer: viewer now viewer-0 (OpenGLStoredQt)
G4VisManager::SetCurrentSceneHandler: scene handler now "scene-handler-0"
Traversing scene data...
Traversing scene data...
```

Qt Visualisation Driver – Test202 example – looking for a volume

The screenshot shows the G4UI Session application window. The main 3D view displays a complex 3D model of a particle detector with various components labeled, including "daughter_box", "e-cons", and several "phys" volumes. A red cylinder and a blue line are overlaid on the model. The top right of the view shows the date and time: "Mon Sep 10 15:44:02 2012".

The left sidebar contains a tree view of objects:

- name
- ▶ Axes
- ▶ Date
- ▶ Frame
- ▶ Text
- ▶ Touchables
 - ▼ expHall_P [0]
 - ▶ B [0]
 - ▶ BoxInBox [0]
 - ▶ C [0]
 - ▶ PD_physical [0]
 - ▶ Polycone_Test [0]
 - ▶ Polyhedra_Test [0]
 - ▶ Test orb [0]
 - ▶ TestTessCut-phys [0]
 - ▶ TestTessCut-phys [0]
 - ▶ TestTrapCut-phys [0]
 - ▶ TestTrapCut-phys [0]

Below the tree view is a "Touchables slider" with a "Show all" button on the left and a "Hide all" button on the right. A search bar contains the text "Poly" and a "select item(s)" button.

The bottom left shows a "Session:" label and an empty text input field.

The bottom right contains an "Output" window with the following text:

```
Refreshing viewer "viewer-0 (OpenGLStoredQt)" ...
Viewer "viewer-0 (OpenGLStoredQt)" refreshed.
(You might also need "/vis/viewer/update".)
/vis/viewer/set/projection p
Projection style of viewer "viewer-0 (OpenGLStoredQt)" set to perspective
with half angle 30 degrees.
/vis/viewer/refresh
Refreshing viewer "viewer-0 (OpenGLStoredQt)"...
Viewer "viewer-0 (OpenGLStoredQt)" refreshed.
(You might also need "/vis/viewer/update".)
0x11a31ec700x11a4750900x11a4750900x11a4f4590Traversing scene data...
Traversing scene data...
```

Below the output window is a "clear" button and a "Filter:" label followed by an empty text input field.

Qt Visualisation Driver – Test202 example – scene/viewer component

The screenshot displays the G4UI Session application window. The title bar reads "G4UI Session". The interface is divided into several sections:

- Menu Bar:** Includes "scene", "viewer", "Scen...", "Help", and "History".
- Left Panel (Control Panel):**
 - add:** Contains an "Apply" button.
 - scale:** A section for scaling objects with fields for "length" (set to 1), "unit" (set to auto), "direction" (set to auto), "Choose color" (with a "Change color" button), "auto/manual" (set to auto), and "xmid", "ymid", "zmid" (all set to 0). The "unit" field is set to "m". An "Apply" button is at the bottom.
 - text:** Fields for "x" and "y" (both set to 0).
 - Session:** A label "Session:" followed by an empty input field.
- Viewer Window (viewer-0 (OpenGLStoredQt)):** Displays a 3D scene with various geometric objects. Labels within the scene include "daughter_box", "e-cons", and "c_p_phys[0]", "c_p_phys[1]", "c_p_phys[2]". A red cylinder is prominent. A timestamp "Mon Sep 10 15:43:20 2012" is visible in the top right of the viewer.
- Output Panel:** A text area showing the following log messages:

```
Refreshing viewer viewer-0 (OpenGLStoredQt) ...  
Viewer "viewer-0 (OpenGLStoredQt)" refreshed.  
(You might also need "/vis/viewer/update".)  
/vis/viewer/set/hiddenEdge 0  
Drawing style of viewer "viewer-0 (OpenGLStoredQt)" set to hsr - hidden surfaces removed  
/vis/viewer/refresh  
Refreshing viewer "viewer-0 (OpenGLStoredQt)"...  
Traversing scene data...  
Traversing scene data...  
Viewer "viewer-0 (OpenGLStoredQt)" refreshed.  
(You might also need "/vis/viewer/update".)
```
- Bottom Panel:** Contains a "clear" button and a "Filter:" input field.

Qt Visualisation Driver – What next ? –

viewer-0 (OpenGLStoredQt)

Mon Sep 10 15:43:20 2012

add

Apply

scale

1

auto

auto

color Change color

auto/manual auto

Print

id 0

ymid 0

zmid 0

unit m

Apply

text

x 0

y 0

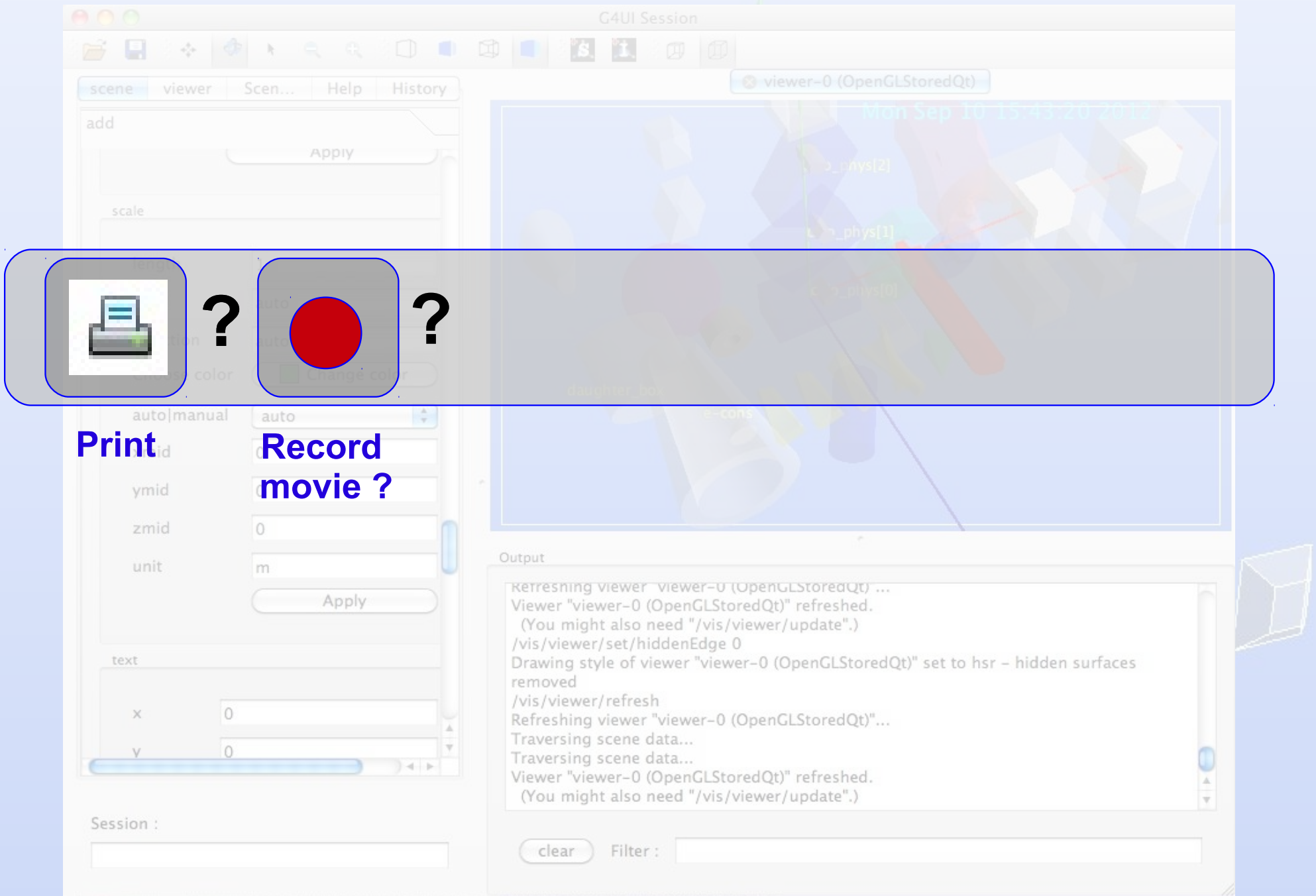
Session :

Output

```
Refreshing viewer "viewer-0 (OpenGLStoredQt)" ...
Viewer "viewer-0 (OpenGLStoredQt)" refreshed.
(You might also need "/vis/viewer/update".)
/vis/viewer/set/hiddenEdge 0
Drawing style of viewer "viewer-0 (OpenGLStoredQt)" set to hsr - hidden surfaces removed
/vis/viewer/refresh
Refreshing viewer "viewer-0 (OpenGLStoredQt)"...
Traversing scene data...
Traversing scene data...
Viewer "viewer-0 (OpenGLStoredQt)" refreshed.
(You might also need "/vis/viewer/update".)
```

clear Filter :

Qt Visualisation Driver – What next ? –



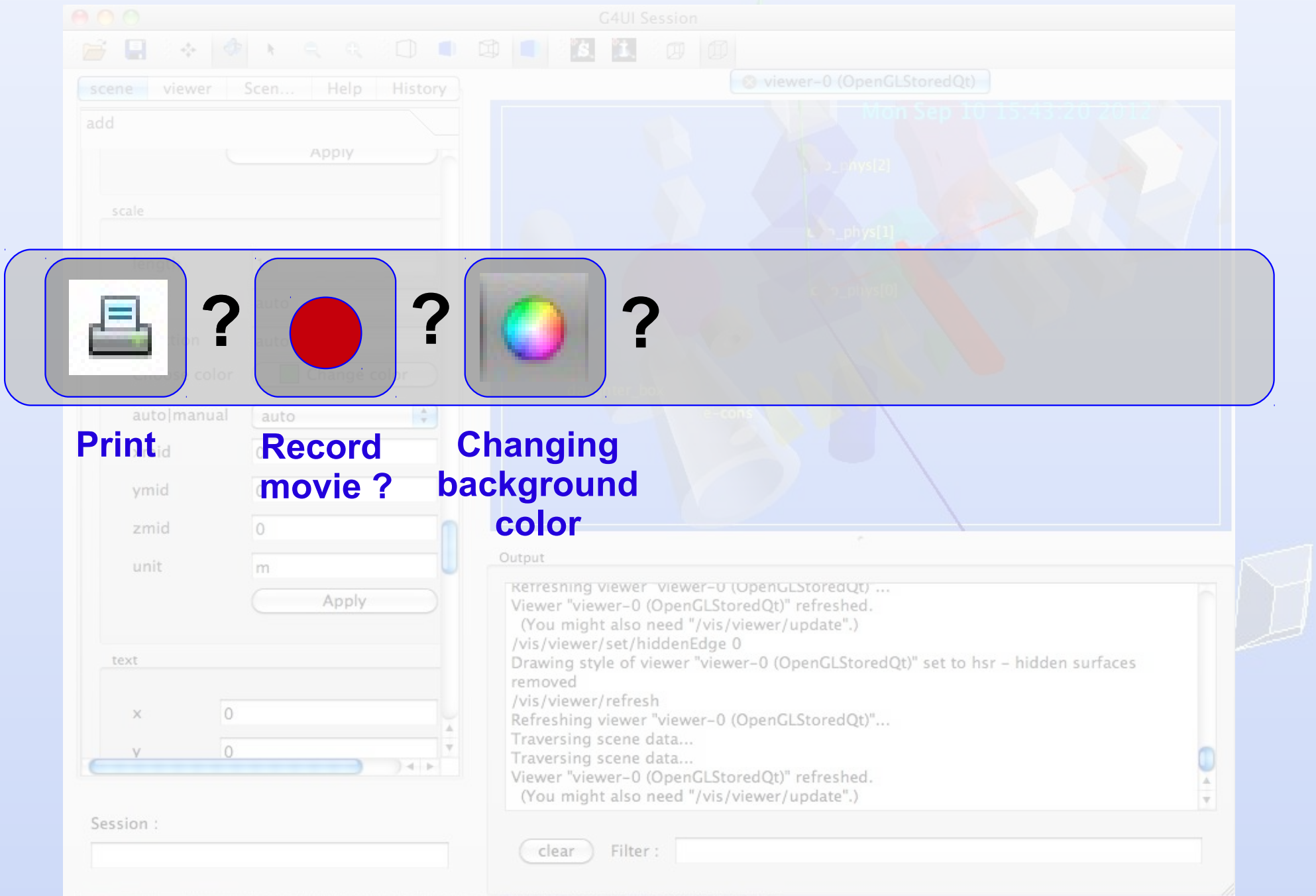
The image shows a screenshot of the G4UI Session software interface. The main window displays a 3D visualization of a particle detector geometry. A semi-transparent grey overlay is positioned in the foreground, containing two icons: a printer icon and a red circle icon, each followed by a question mark. Below the overlay, the text "Print" and "Record movie ?" is visible in blue. The background interface includes a menu bar, a toolbar, a scene viewer, and an output console.

Print ? **Record movie ?**


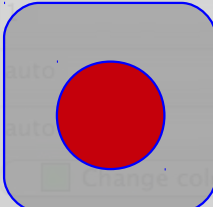

Output

```
Retreshing viewer "viewer-0 (OpenGLStoredQt)" ...  
Viewer "viewer-0 (OpenGLStoredQt)" refreshed.  
(You might also need "/vis/viewer/update".)  
/vis/viewer/set/hiddenEdge 0  
Drawing style of viewer "viewer-0 (OpenGLStoredQt)" set to hsr - hidden surfaces removed  
/vis/viewer/refresh  
Refreshing viewer "viewer-0 (OpenGLStoredQt)"...  
Traversing scene data...  
Traversing scene data...  
Viewer "viewer-0 (OpenGLStoredQt)" refreshed.  
(You might also need "/vis/viewer/update".)
```

Qt Visualisation Driver – What next ? –



The image shows a screenshot of the G4UI Session software interface. A semi-transparent grey bar with a blue border is overlaid across the middle of the screen. This bar contains three icons, each followed by a large black question mark:

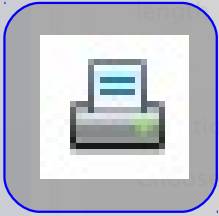
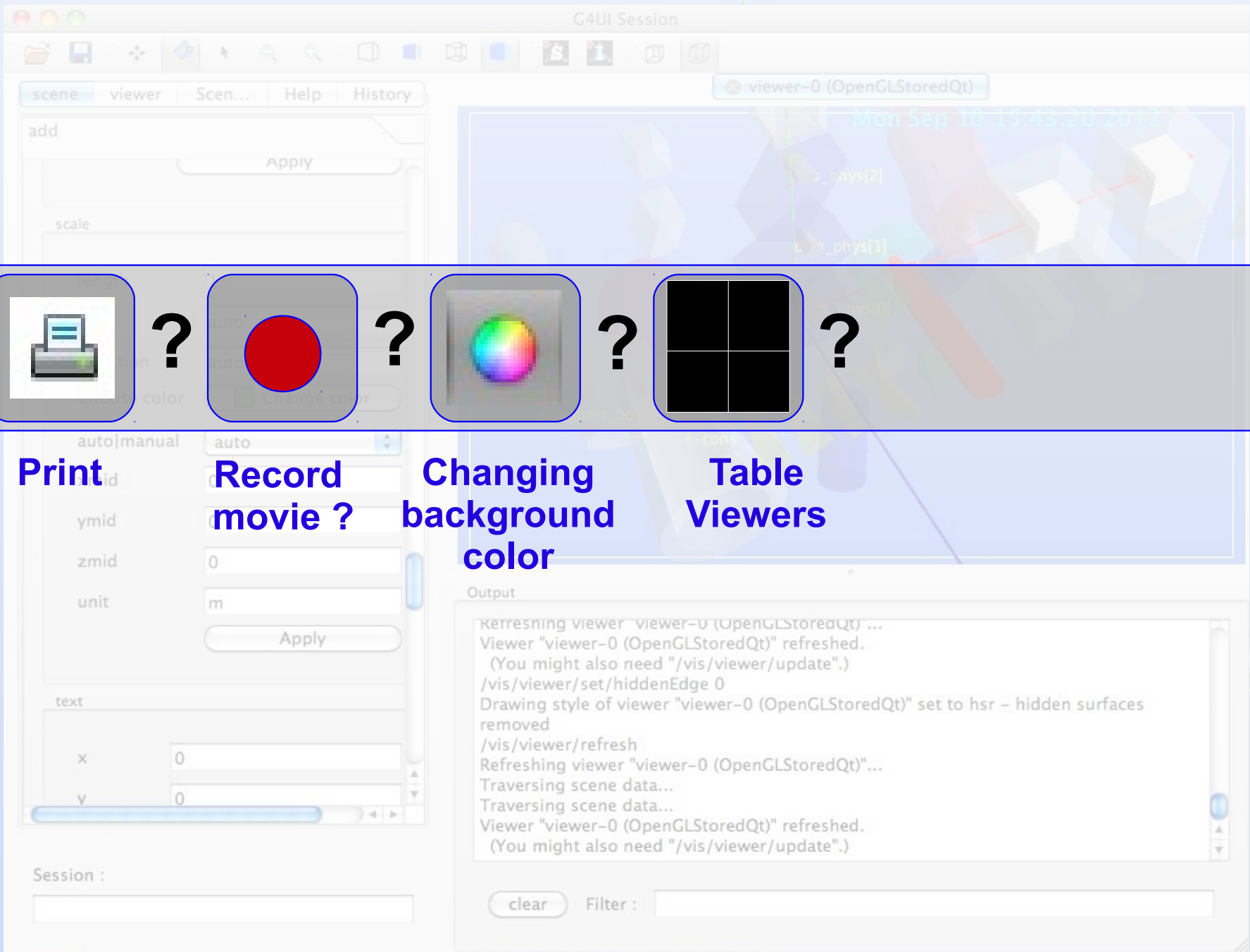
-  ?
-  ?
-  ?

Below the bar, the text labels for these functions are written in blue:

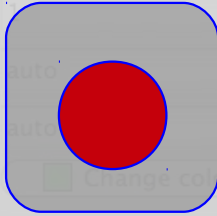
- Print**
- Record movie ?**
- Changing background color**

The background shows the G4UI Session window with a 3D visualization of a particle detector. The interface includes a menu bar (scene, viewer, Scen..., Help, History), a toolbar, and a central viewer window titled "viewer-0 (OpenGLStoredQt)". The viewer displays a 3D scene with various components labeled, including "phys[2]", "phys[1]", and "phys[0]". A timestamp "Mon Sep 10 15:43:20 2012" is visible in the top right of the viewer. Below the viewer, there are control panels for "add", "scale", and "text" (with x and y coordinates set to 0). An "Output" window at the bottom right shows a log of viewer refresh and scene traversal messages.

Qt Visualisation Driver – What next ? –



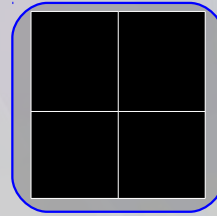
?



?



?



?

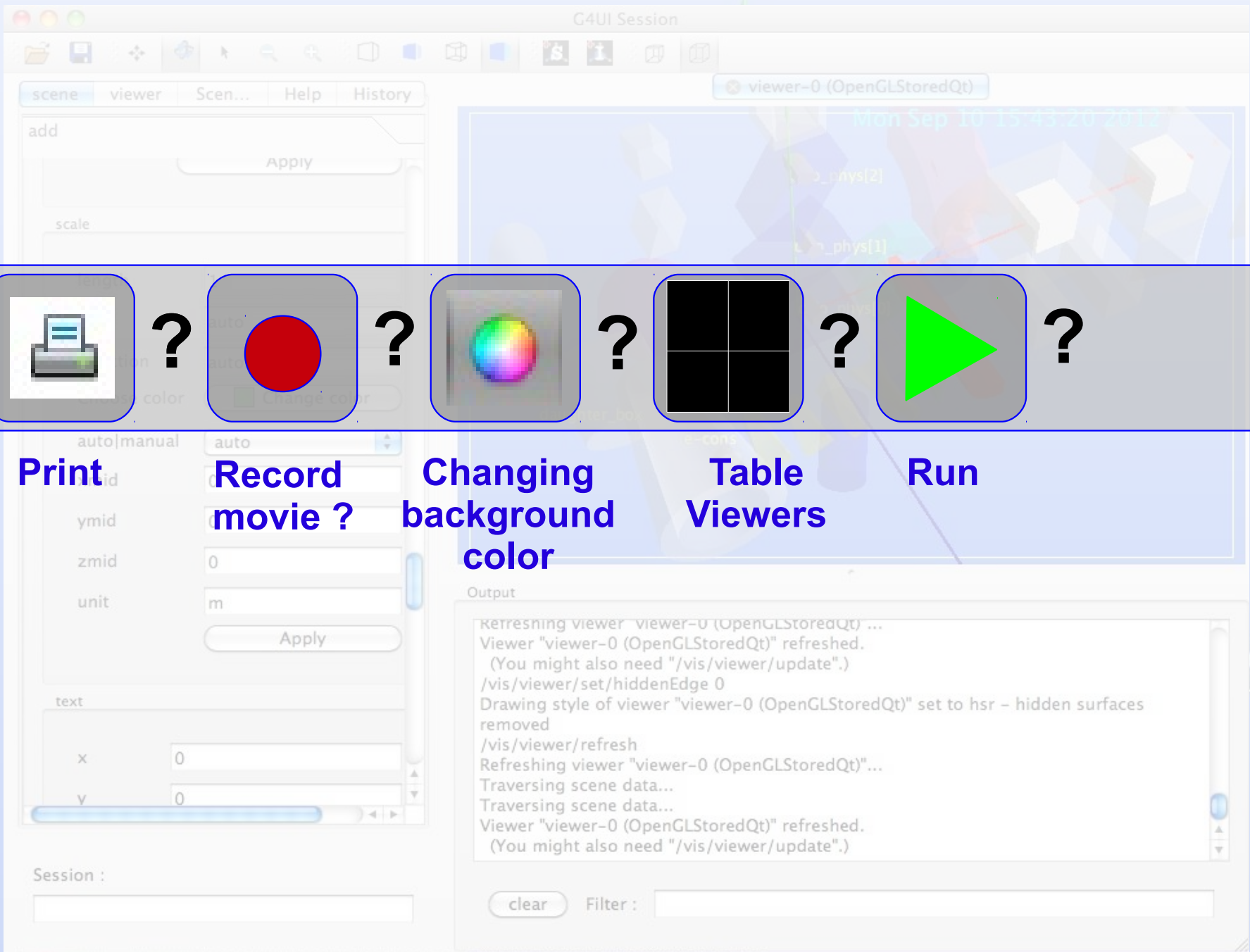
Print

**Record
movie ?**

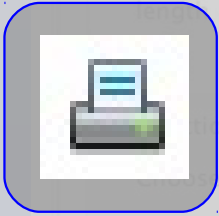
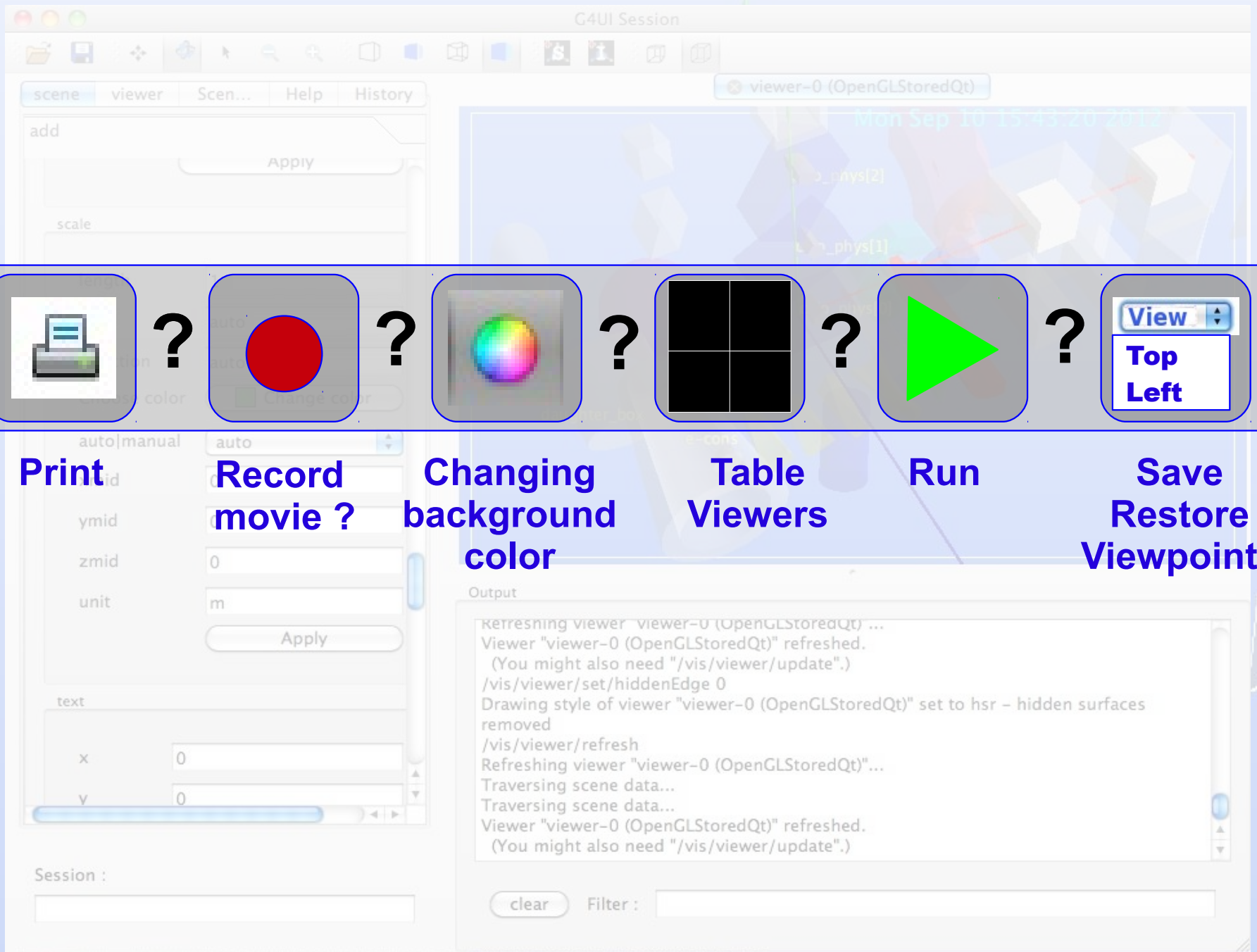
**Changing
background
color**

**Table
Viewers**

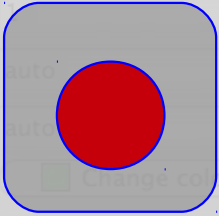
Qt Visualisation Driver – What next ? –



Qt Visualisation Driver – What next ? –



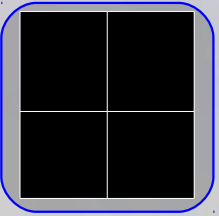
?



?



?



?



?



?

Print

**Record
movie ?**

**Changing
background
color**

**Table
Viewers**

Run

**Save
Restore
Viewpoints**

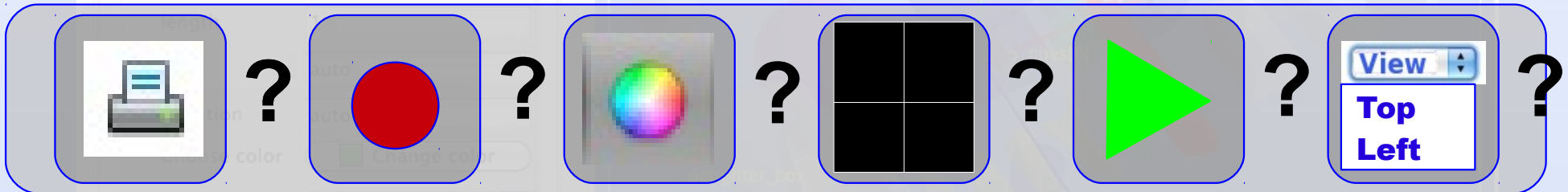
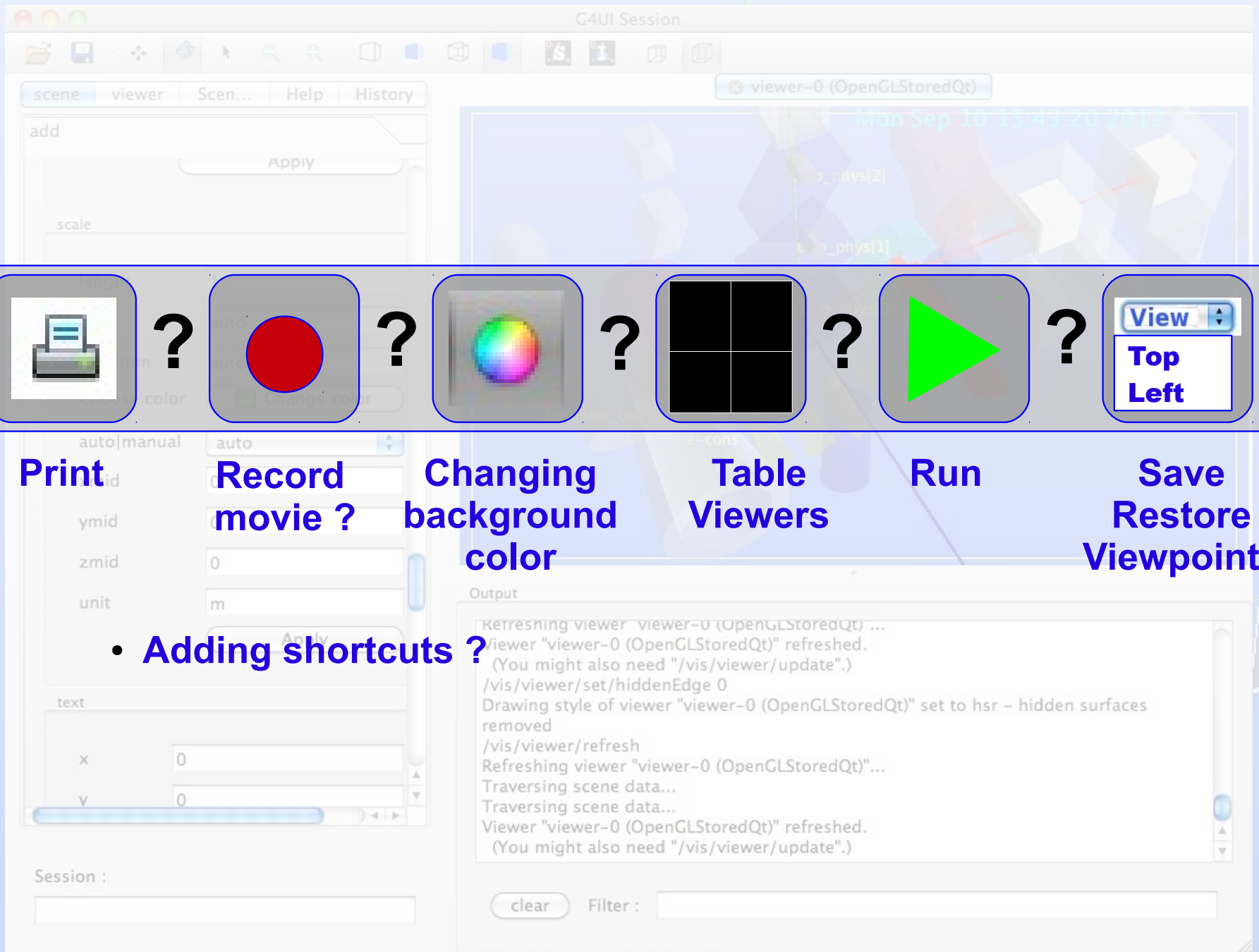
Output

```
Retreshing viewer "viewer-0 (OpenGLStoredQt)" ...  
Viewer "viewer-0 (OpenGLStoredQt)" refreshed.  
(You might also need "/vis/viewer/update".)  
/vis/viewer/set/hiddenEdge 0  
Drawing style of viewer "viewer-0 (OpenGLStoredQt)" set to hsr - hidden surfaces  
removed  
/vis/viewer/refresh  
Refreshing viewer "viewer-0 (OpenGLStoredQt)"...  
Traversing scene data...  
Traversing scene data...  
Viewer "viewer-0 (OpenGLStoredQt)" refreshed.  
(You might also need "/vis/viewer/update".)
```

clear

Filter :

Qt Visualisation Driver – What next ? –



Print

Record movie ?

Changing background color

Table Viewers

Run

Save Restore Viewpoints

- **Adding shortcuts ?**

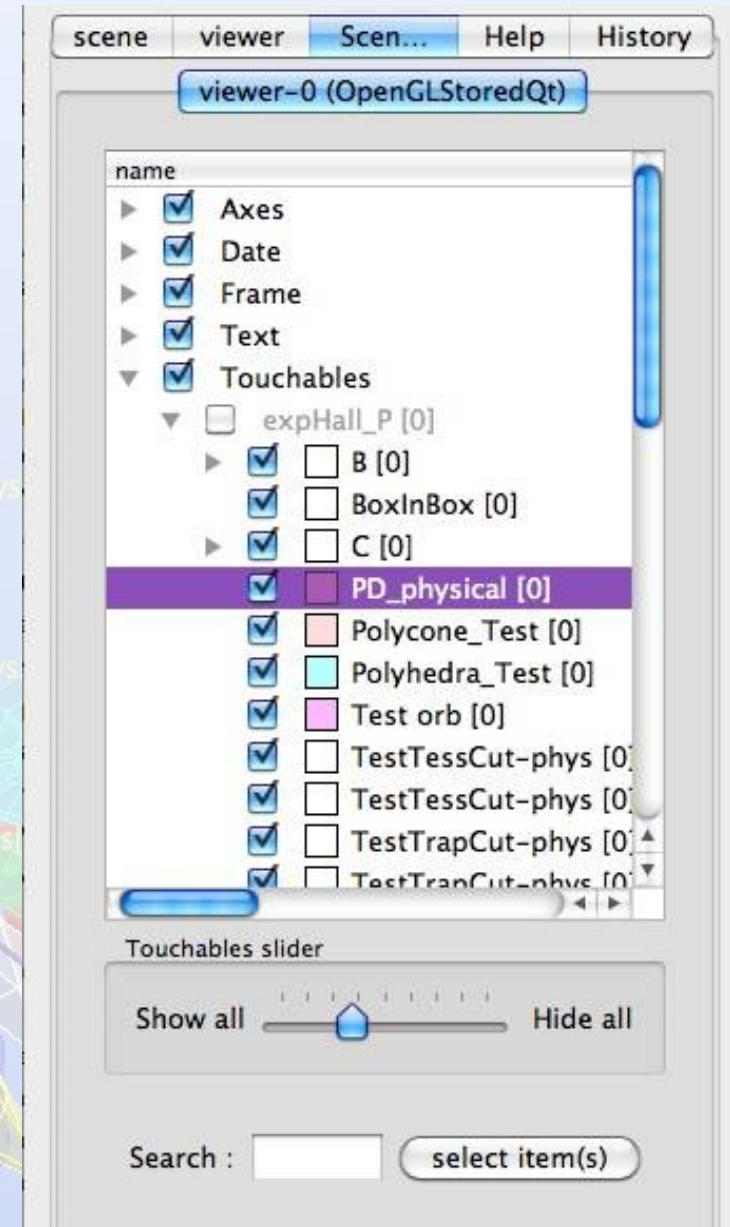
Qt Visualisation Driver – Inside scene viewer –

⇒ Scene tree only implemented in Store Mode in QT

At creation, one opengl display list is create for each « volume »

Corresponding entrie is create in scene tree

At run time, before drawing display lists, we ask the scene tree for display status (color, hide, visible)



Qt Visualisation Driver – Problems –

- Many ways to interact with Geant4 :

- Command line
- Shortcuts
- Context menu
- Icons

⇒ We have to combine all of them to be consistent.

Changing perspective/orthographic with command line should change toolbar icon status

- Some icons should belong to viewers instead of UI.

⇒ wireframe/solid icons are properties of viewer, we should be able to have several viewers with different properties...have to deal with that...

- A good point will be to be able to store/save the scene tree state in a macro file

- Several viewers will display several scene tree, not a problem, but have to think about it.

