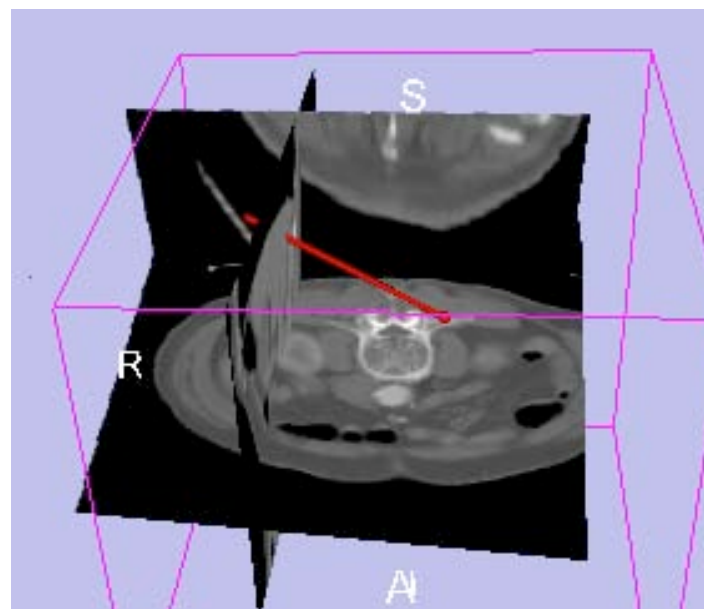




# Image Guided Therapy in Slicer3

Introduction to Navigation  
using OpenIGTLink

Danielle Pace, B.CmpH





# Acknowledgements

---



**Surgical Planning Lab, Harvard Medical School**  
Junichi Tokuda, Haiying Liu, Nobuhiko Hata, Steve Pieper, Ron Kikinis



**National Alliance for Medical Image Computing**



**National Center for Image-Guided Therapy**



**Robarts Research Institute**  
Chris Wedlake



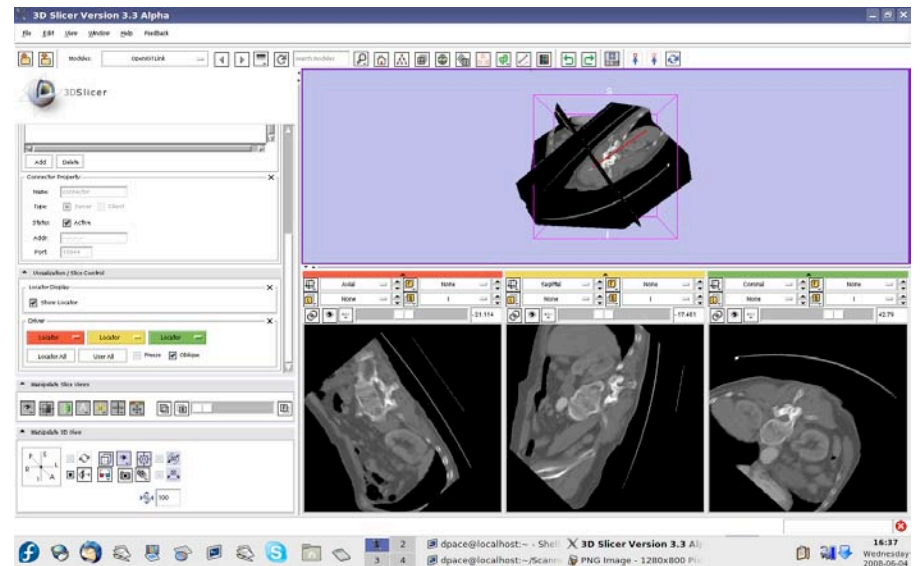
**NEDO Intelligent Surgical Instruments Project**  
Kiyo Chinzei



# Learning objective

Following this tutorial,  
you will:

- Understand how to use tracking devices with Slicer3 using the OpenIGTLink module
- OpenIGTLink can also be used to interface with other devices, such as imaging devices and medical robots

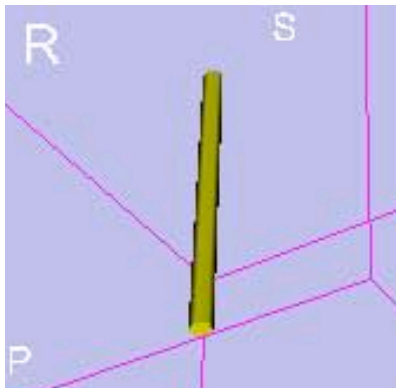




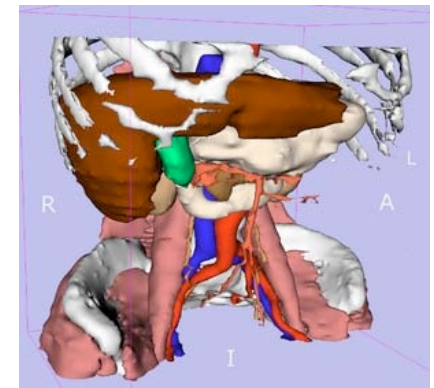
# Material

---

- This course requires a simple VTK tool model plus *either* the SPL-PNL brain atlas or the SPL abdominal atlas:



**VTK model:**



**Brain and abdominal atlases:**

<http://wiki.na-mic.org/Wiki/index.php/IGT:ToolKit/Navigation-tutorial>



# Required software

---

This tutorial requires the [OpenIGTLink Slicer3 module](#) and a [tracker simulator](#):

- For both of these, you have the choice of either downloading a precompiled version (binary) **OR** building it yourself from the source code

For installation instructions, see the wiki page at <http://wiki.na-mic.org/Wiki/index.php/IGT:ToolKit/Navigation-tutorial>

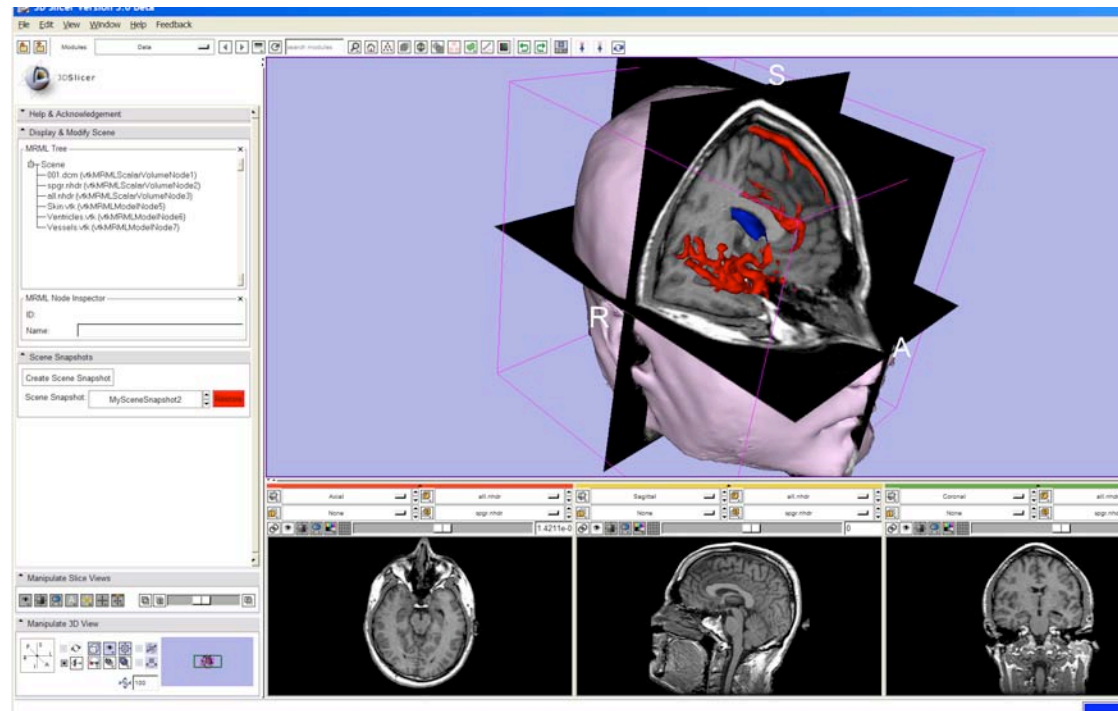
*Disclaimer: It is the responsibility of the user of 3D Slicer to comply with both the terms of the license and with the applicable laws, regulations and rules.*



# Prerequisites

- Data Loading and Visualization in Slicer3:

[http://wiki.na-mic.org/Wiki/index.php/Slicer:Workshops:Slicer3\\_Training](http://wiki.na-mic.org/Wiki/index.php/Slicer:Workshops:Slicer3_Training)





# Tutorial outline

---

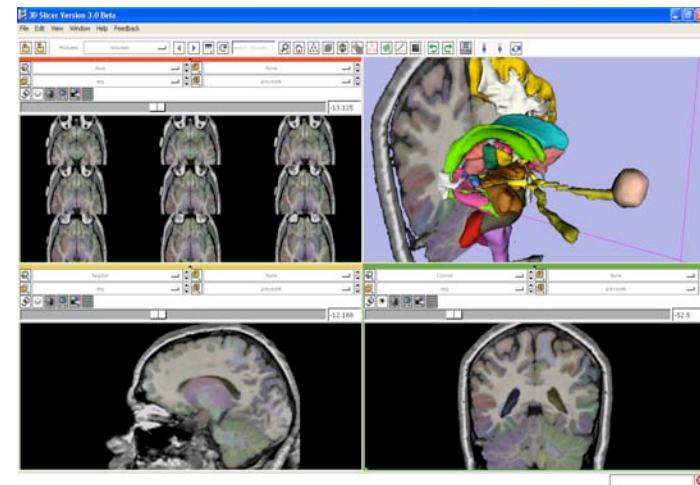
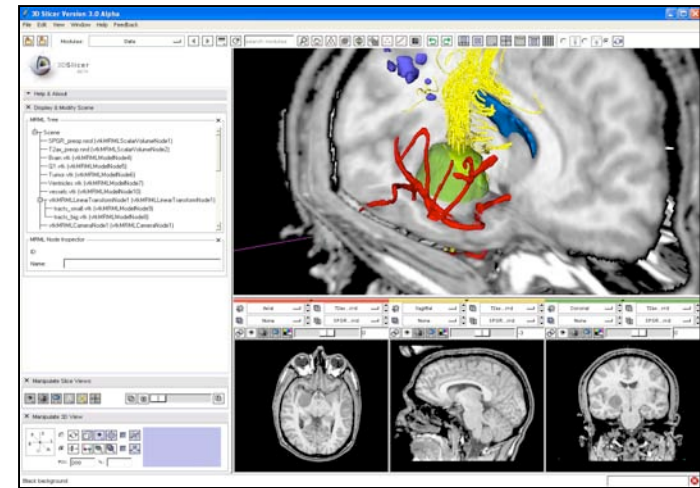
1. **Introduction to surgical navigation**
2. Interfacing Slicer3 with external devices using OpenIGTLink
3. Hands-on navigation using a tracking simulator
4. Examples of OpenIGTLink in use





# 3D Slicer

- Integrates algorithms and utilities for medical image computing research and Image Guided Therapy into a single framework
- Is both an end-user application and a platform for research
- The precompiled program and the source code are both freely downloadable







# Image Guided Therapy (IGT) in Slicer3

---

Slicer3 has extensive support for IGT, including:

- Visualization
- Registration
- Segmentation
- Model making
- Diffusion Tensor Imaging
- Quantification
- Filtering
- Interfacing to imaging devices, trackers and medical robots

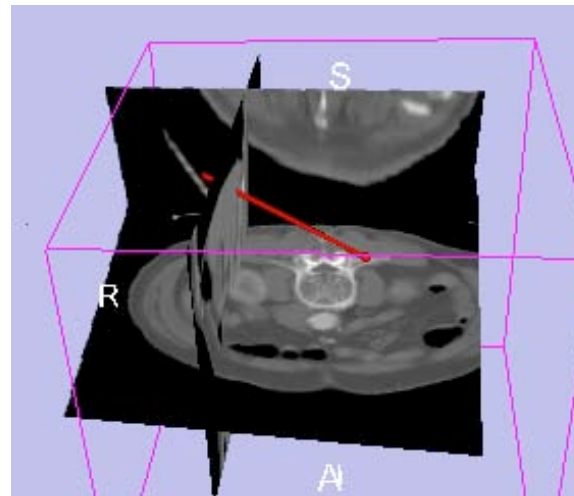
} **Focus of  
this tutorial**



# Navigation in IGT

---

- Determining the **positions and orientations** of surgical tools using a tracking system
- **Displaying virtual representations** of those tools on the screen for the surgeon





# Navigation in IGT

---

- **Selected clinical uses:**
  - Real-time update of tool position and orientation in augmented reality environments (ex. for minimally-invasive cardiac surgery)
  - Image-to-patient registration using tracked pointer tools (ex. for total hip replacement surgery)
  - Image-to-patient registration using tracked intraoperative imaging devices (ex. ultrasound)

In order to perform navigation, software must be able to receive position and orientation data from tracking devices!



# Tutorial outline

---

1. Introduction to surgical navigation
- 2. Interfacing Slicer3 with external devices using OpenIGTLink**
3. Hands-on navigation using a tracking simulator
4. Examples of OpenIGTLink in use



# What is OpenIGTLink?

---

- OpenIGTLink is a **communication protocol** that allows Slicer3 to communicate with external devices



# What is OpenIGTLink?



**Slicer3  
OpenIGTLink  
module**

**OpenIGTLink**



**Imaging devices**  
(ex MRI, US)



**OpenIGTLink**



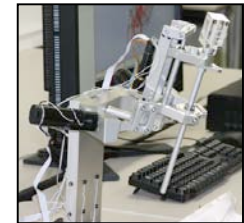
**Tracking devices**



**OpenIGTLink**



**Medical robots**

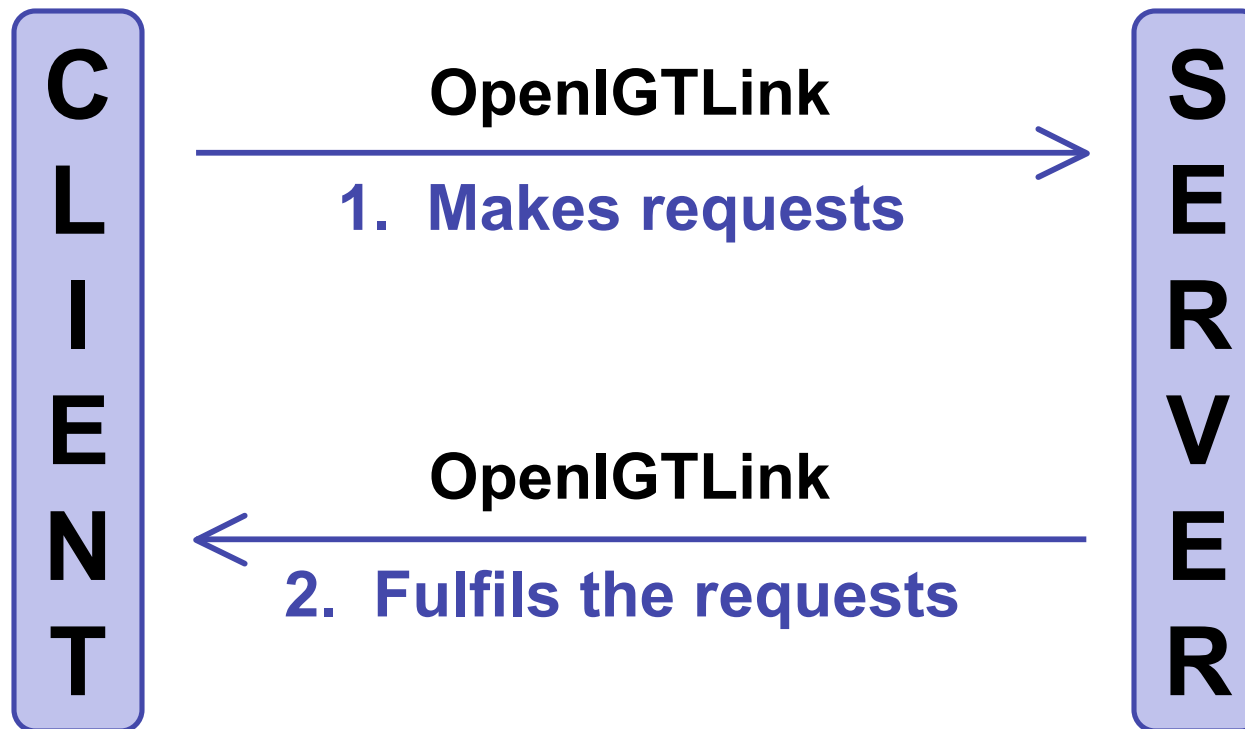




# OpenIGTLink

---

- OpenIGTLink uses a “Client-Server” architecture.

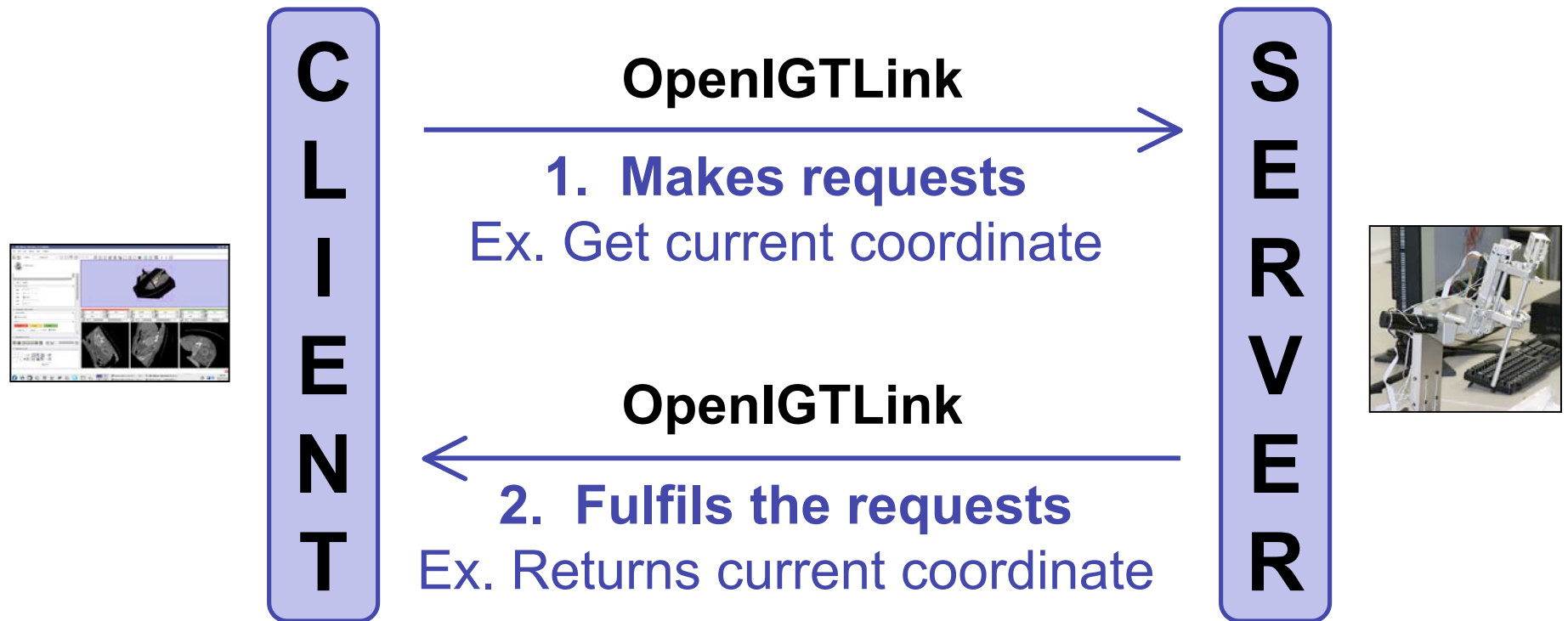






# OpenIGTLink

- Surgical robot example:

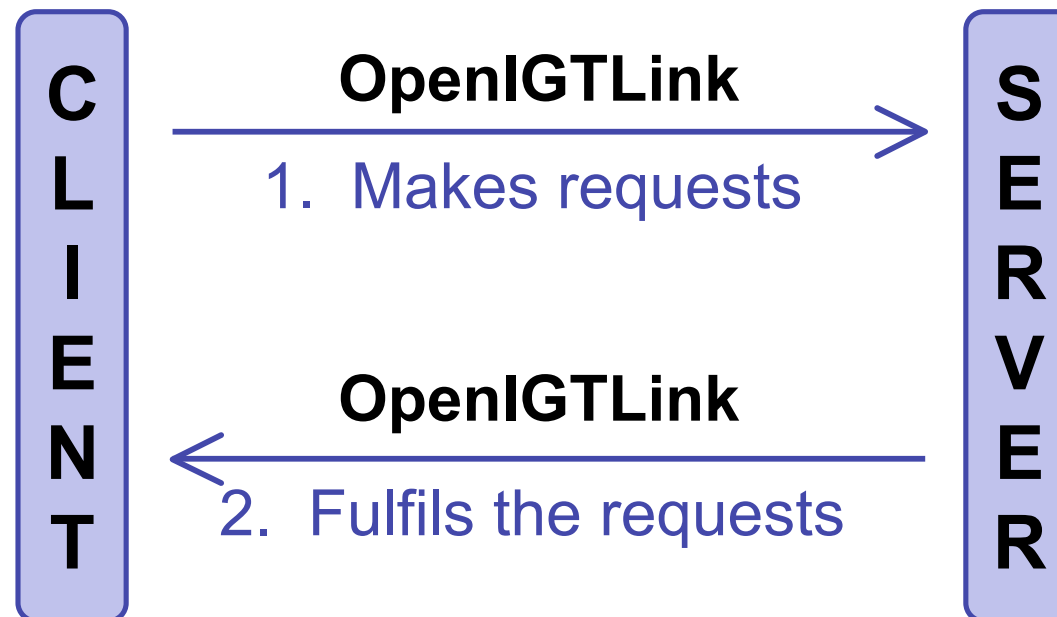




# OpenIGTLink

---

- The OpenIGTLink protocol specifies the structure of the messages sent between the client and the server
- Slicer3 can be either the client or the server, depending on the application





# The OpenIGTLink module in Slicer3

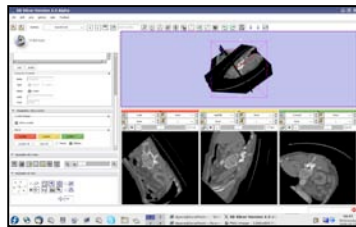
---

- OpenIGTLink is a **protocol**
- There is an **OpenIGTLink module** in Slicer3 that implements the protocol so that Slicer3 can communicate with external devices



# OpenIGTLink and IGSTK

- IGSTK = Image-Guided Surgery Tool Kit
- OpenIGTLink functionality has been added to IGSTK: you can now use IGSTK to write programs that interact with both Slicer3 and the physical device

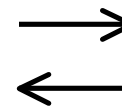


**Slicer3**  
(client or server)

**OpenIGTLink**



**IGSTK**  
program  
(client or server)



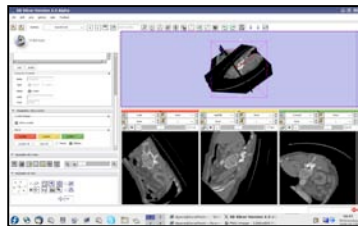
**Device**  
**ex.**  
**tracker**



# The tracking simulator

---

- In this tutorial, a **tracking simulator** is used instead of using an actual tracking device
- The tracking simulator acts as the **client** to send simulated data to Slicer3 (the server) over OpenIGTLink



**Slicer3**  
(server)

**OpenIGTLink**  
↔

**TRACKING  
SIMULATOR**  
(client)



# Tutorial outline

---

1. Introduction to surgical navigation
2. Interfacing Slicer3 with external devices using OpenIGTLink
- 3. Hands-on navigation using a tracking simulator**
4. Examples of OpenIGTLink in use



# Hands-on navigation

---

- Using a tracking simulator, you will learn how to:
  - Set up an OpenIGTLink connection in Slicer3
  - Show the resulting transforms using both the Slicer3 “locator” and a vtk model
  - Add a calibration matrix
  - Reslice image volumes using the tracker transform





# Note

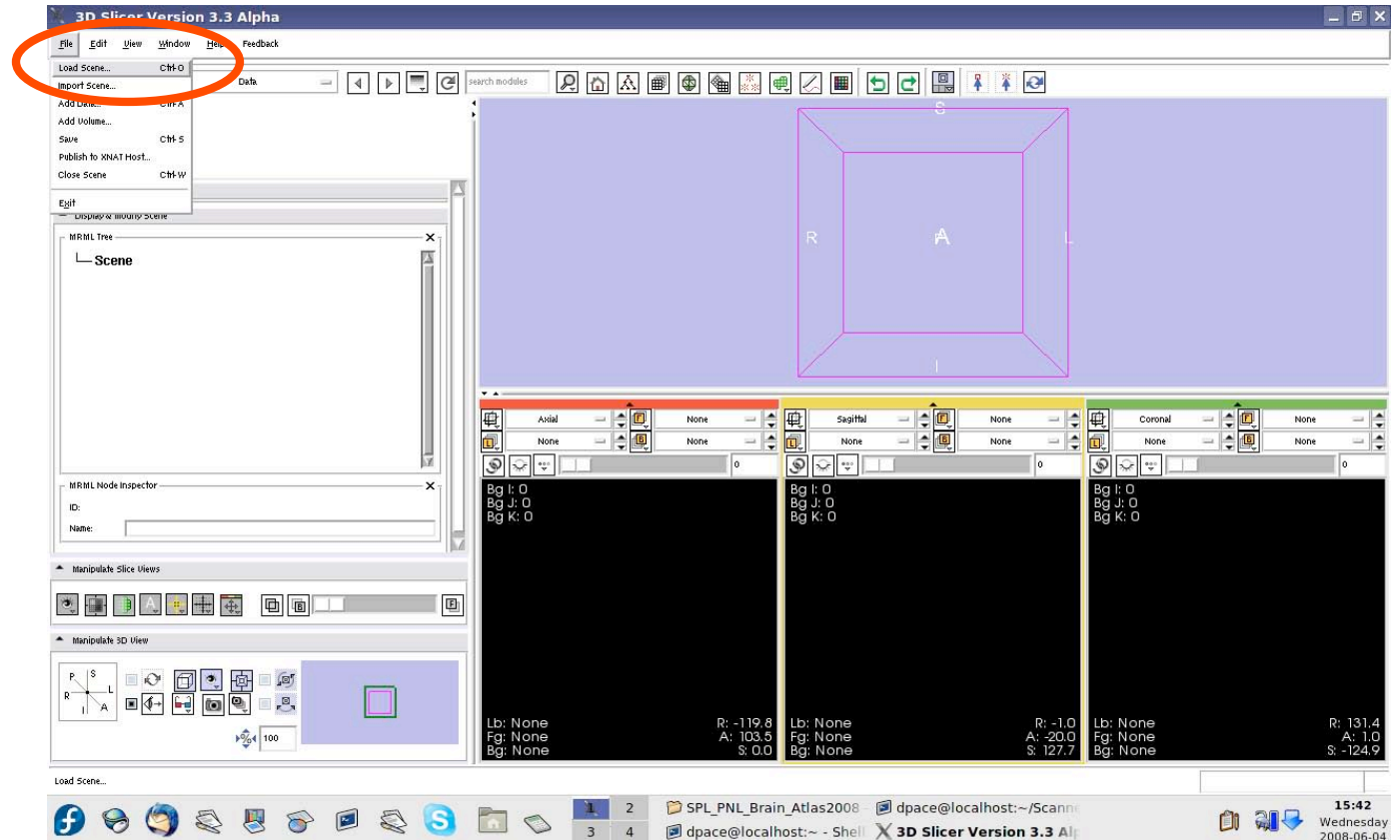
---

- Although the screenshots used in this tutorial use the SPL abdominal atlas, the SPL-PNL brain atlas can also be used



# Load the atlas

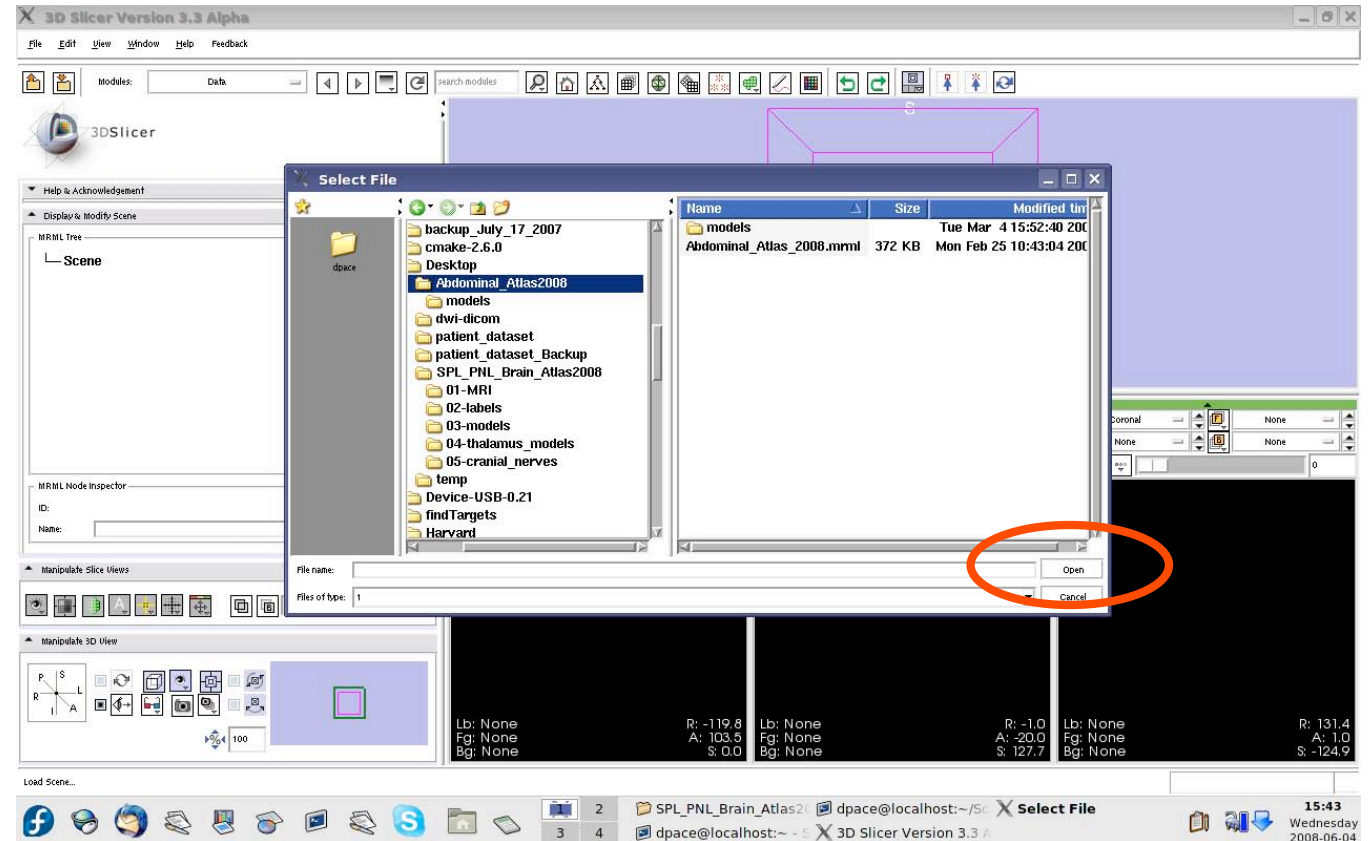
Click on File  
-> Load  
Scene





# Load the atlas

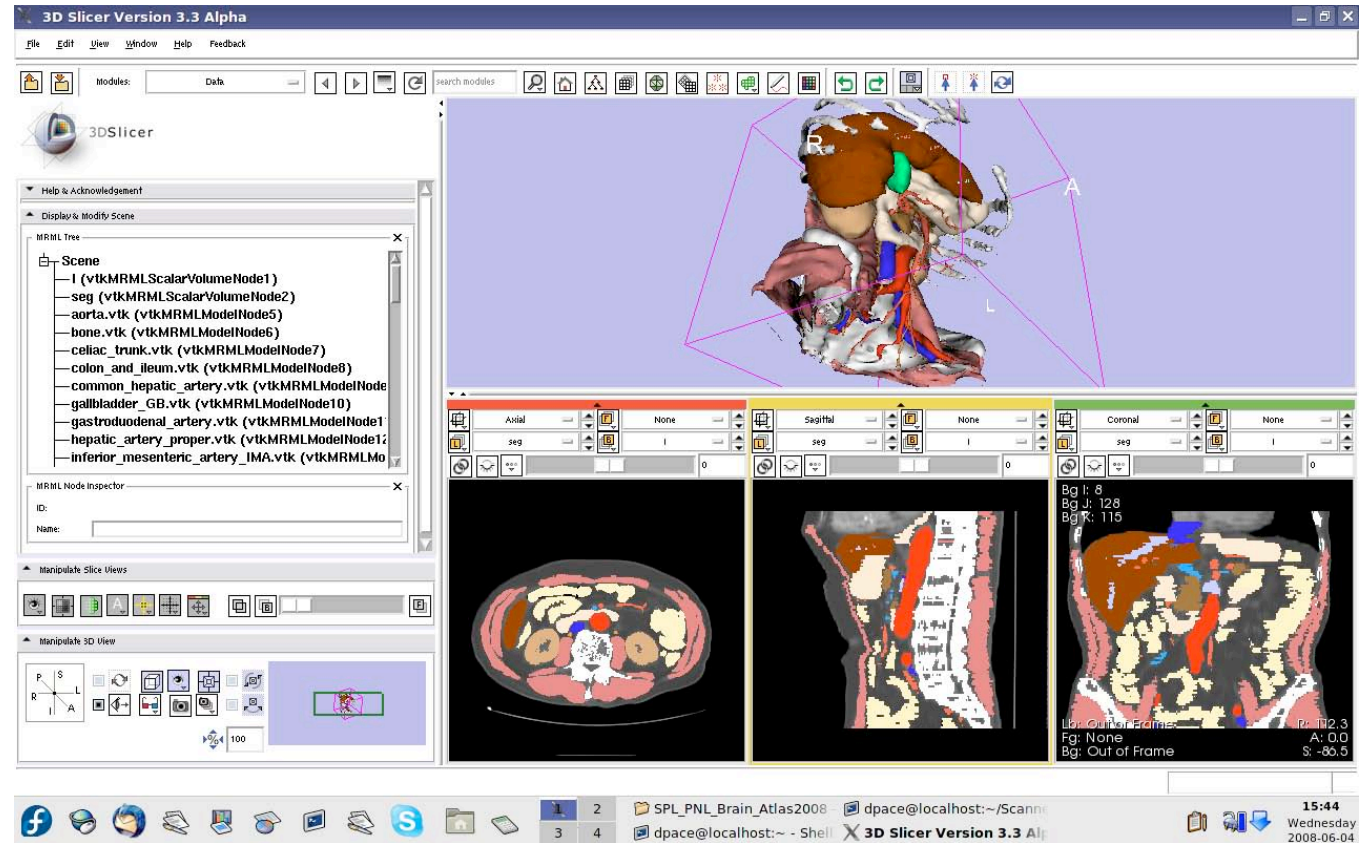
Select the scene file for the atlas (brain\_atlas\_2008.mrml or Abdominal\_Atlas\_2008.mrml) and click “Open”





# Load the atlas

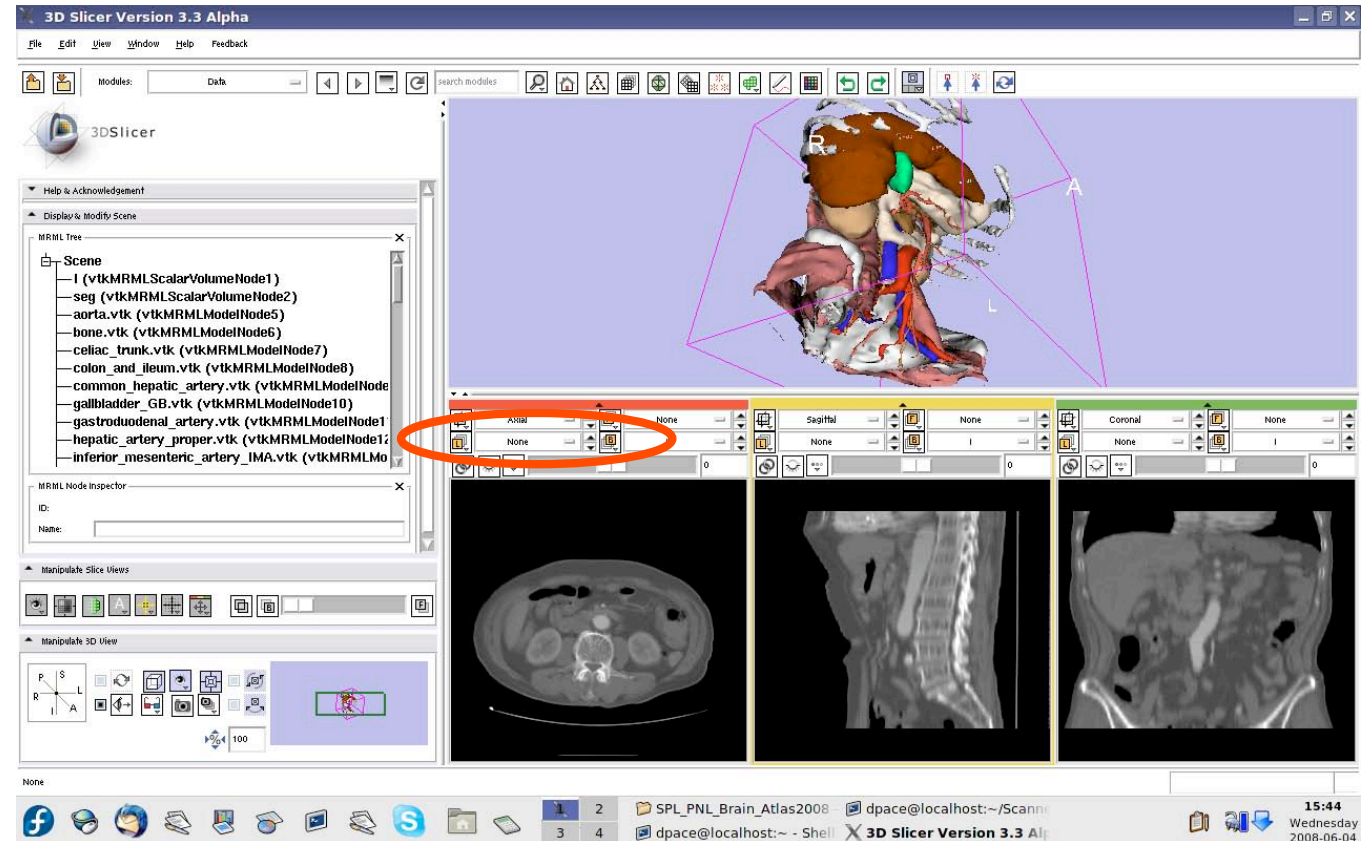
All of the atlas components are shown in the MRML scene within the Data module





# Load the atlas

If you are using the abdominal atlas, change the label map to “None”



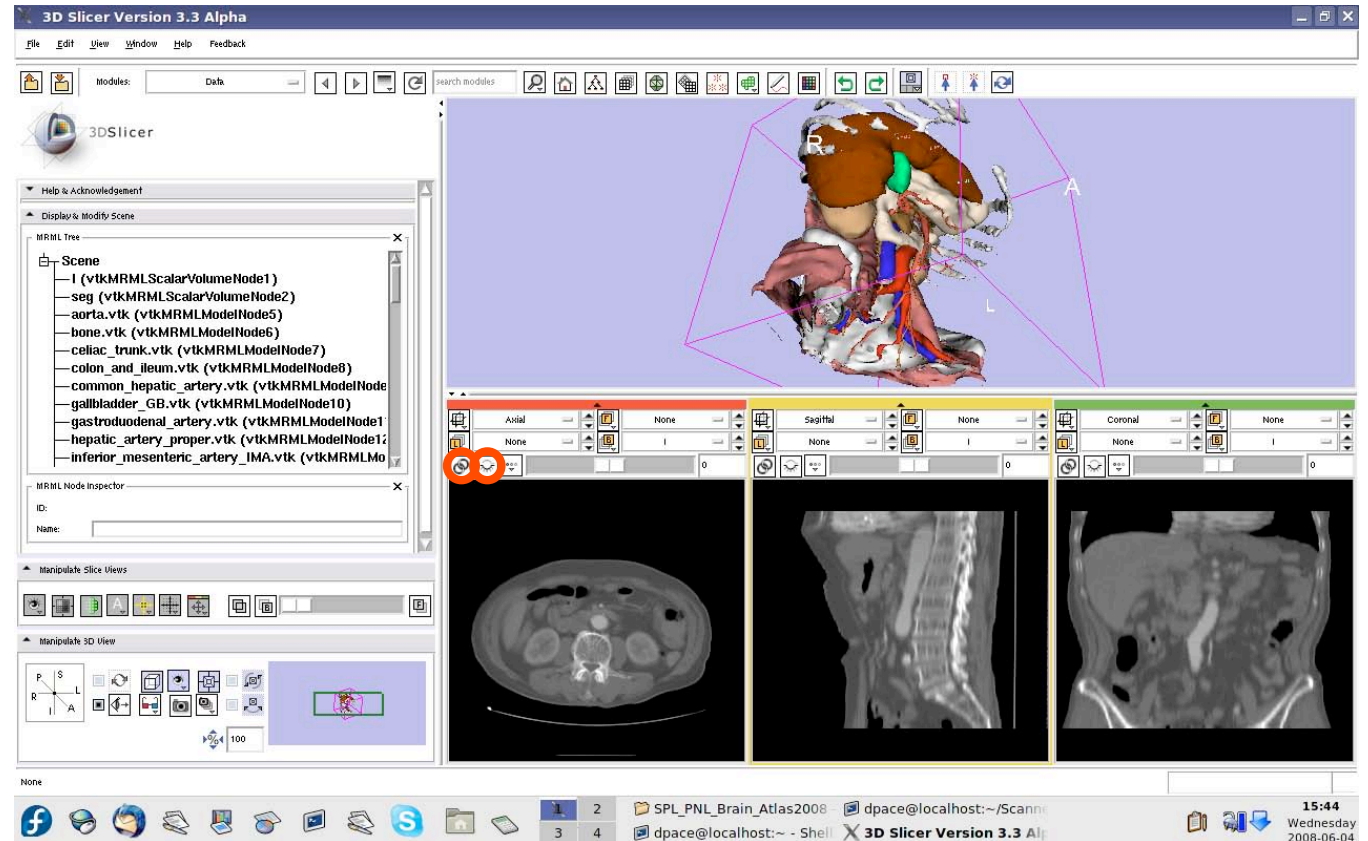


# Load the atlas

If you are using the brain atlas, turn off the visibility of the images:

Click the “Link” button

Click the “Visibility” button

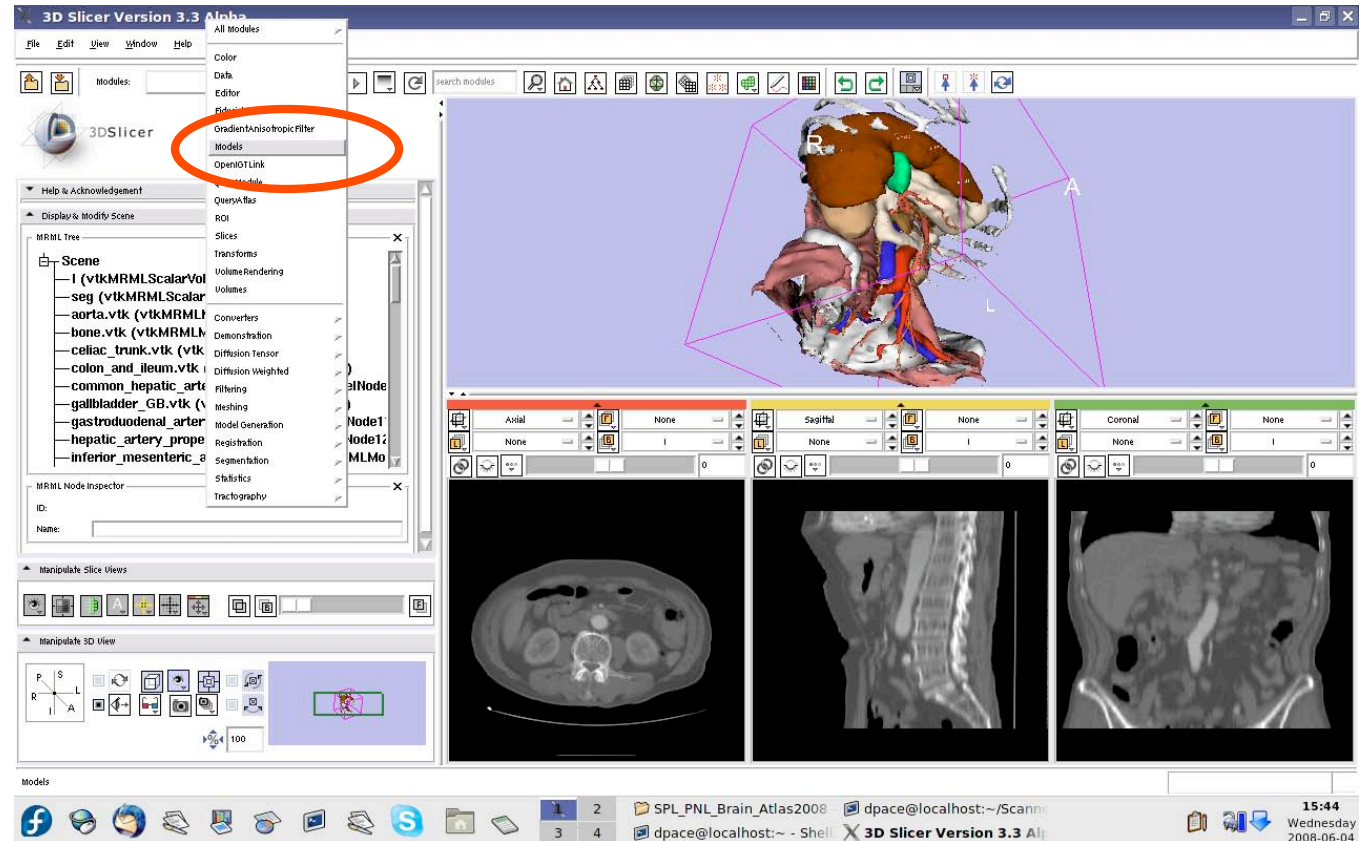






# Make the models invisible

Open the  
Models  
module

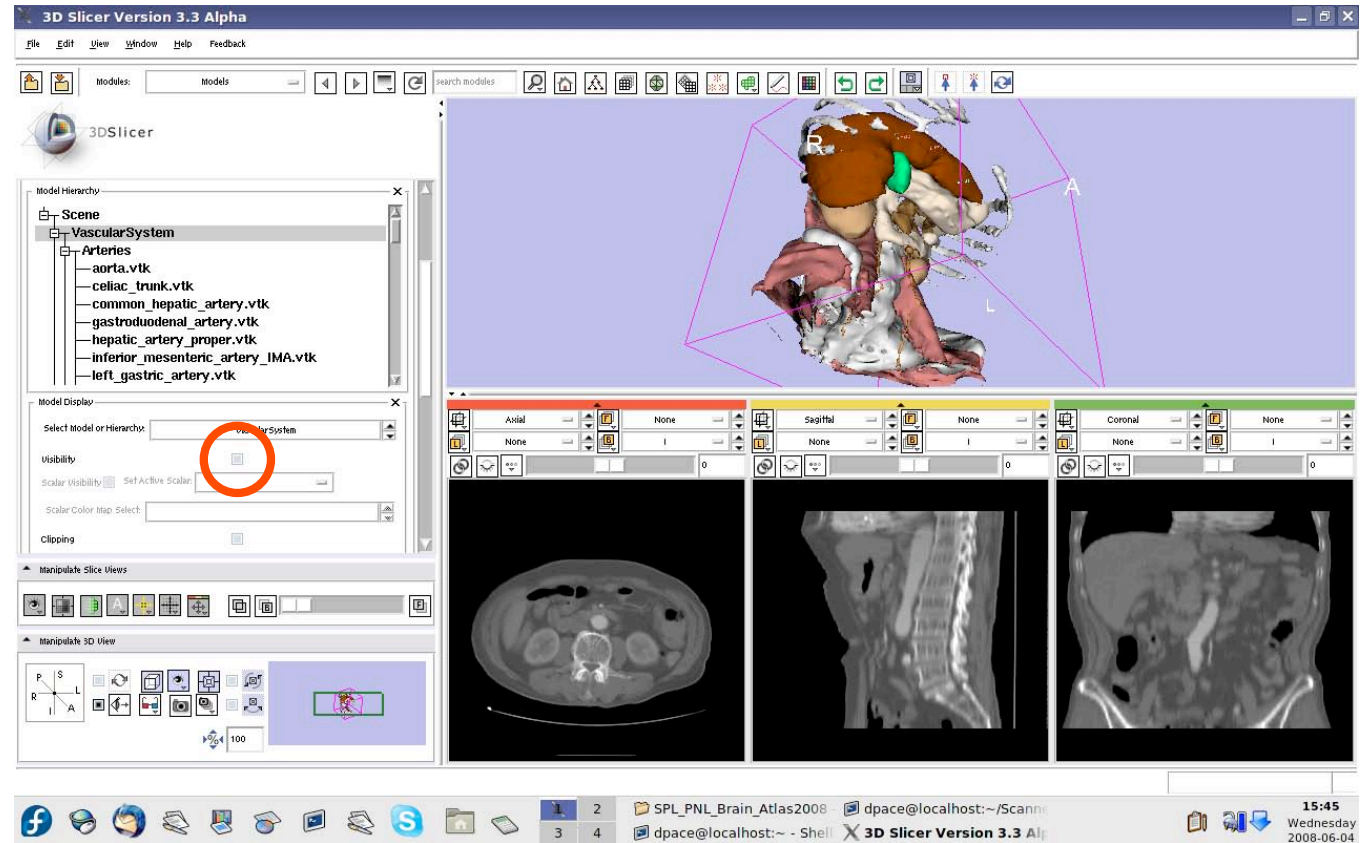






# Make the models invisible

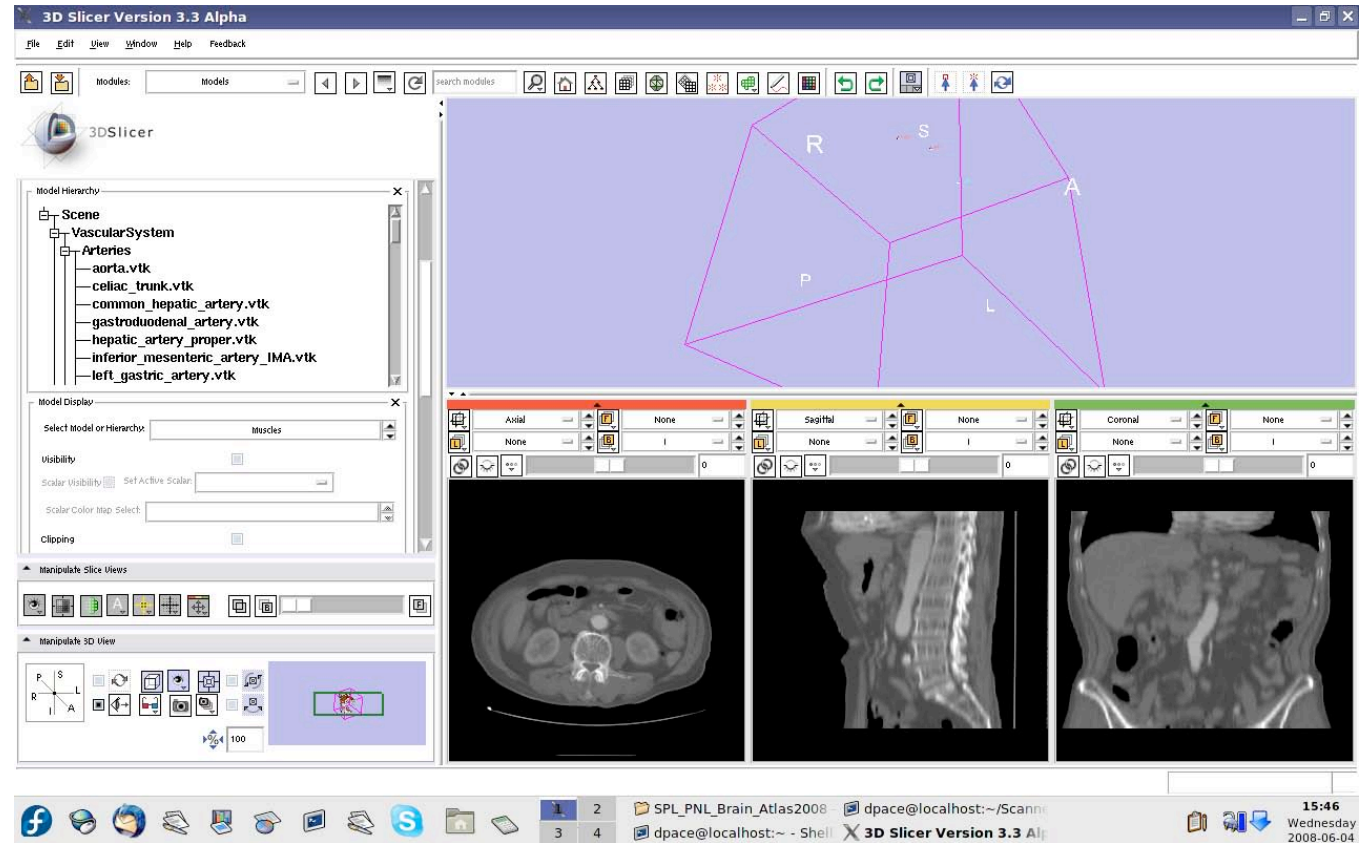
For each of the major headings in the model hierarchy, turn the visibility off





# Make the models invisible

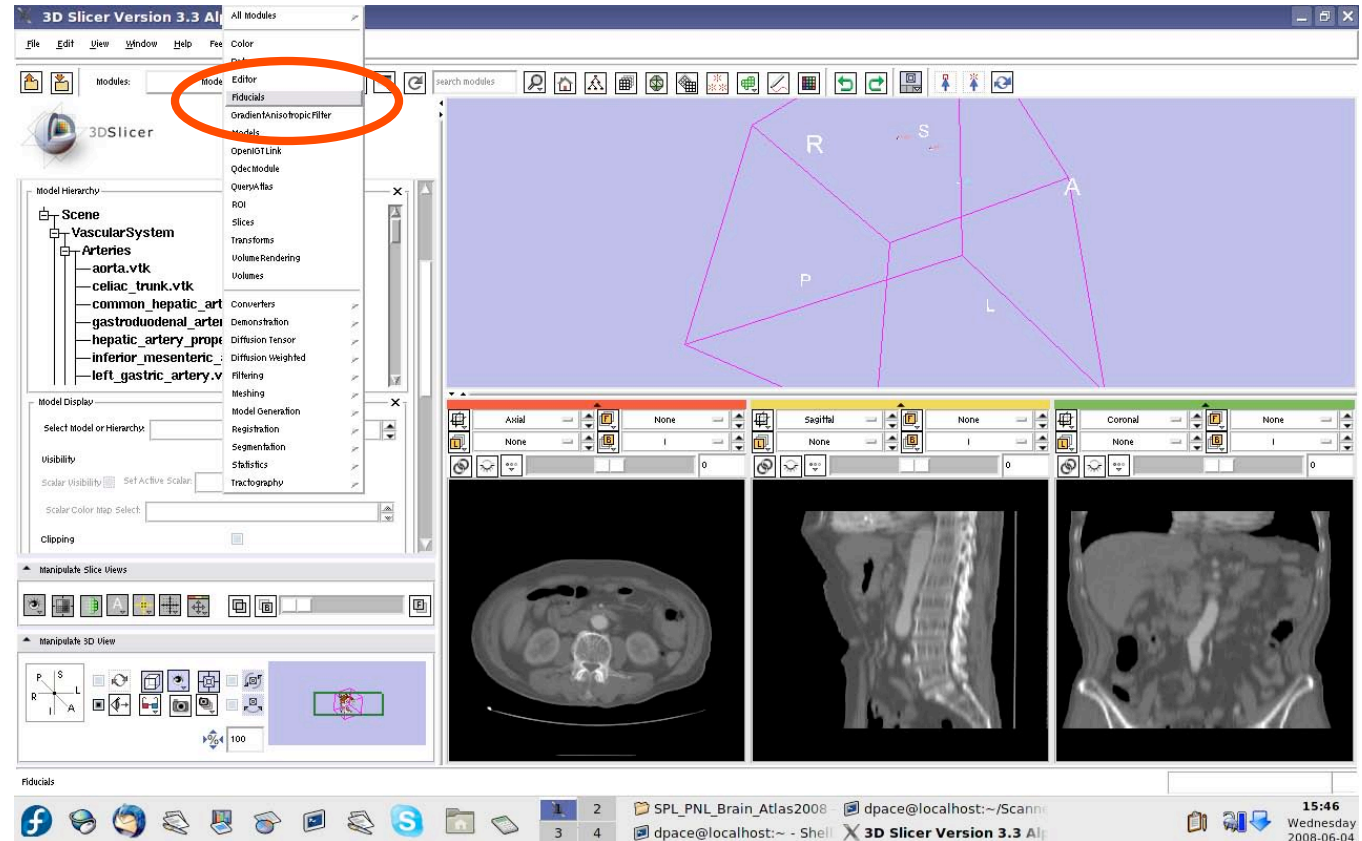
When you are finished, no models will be shown





# Make the fiducials invisible

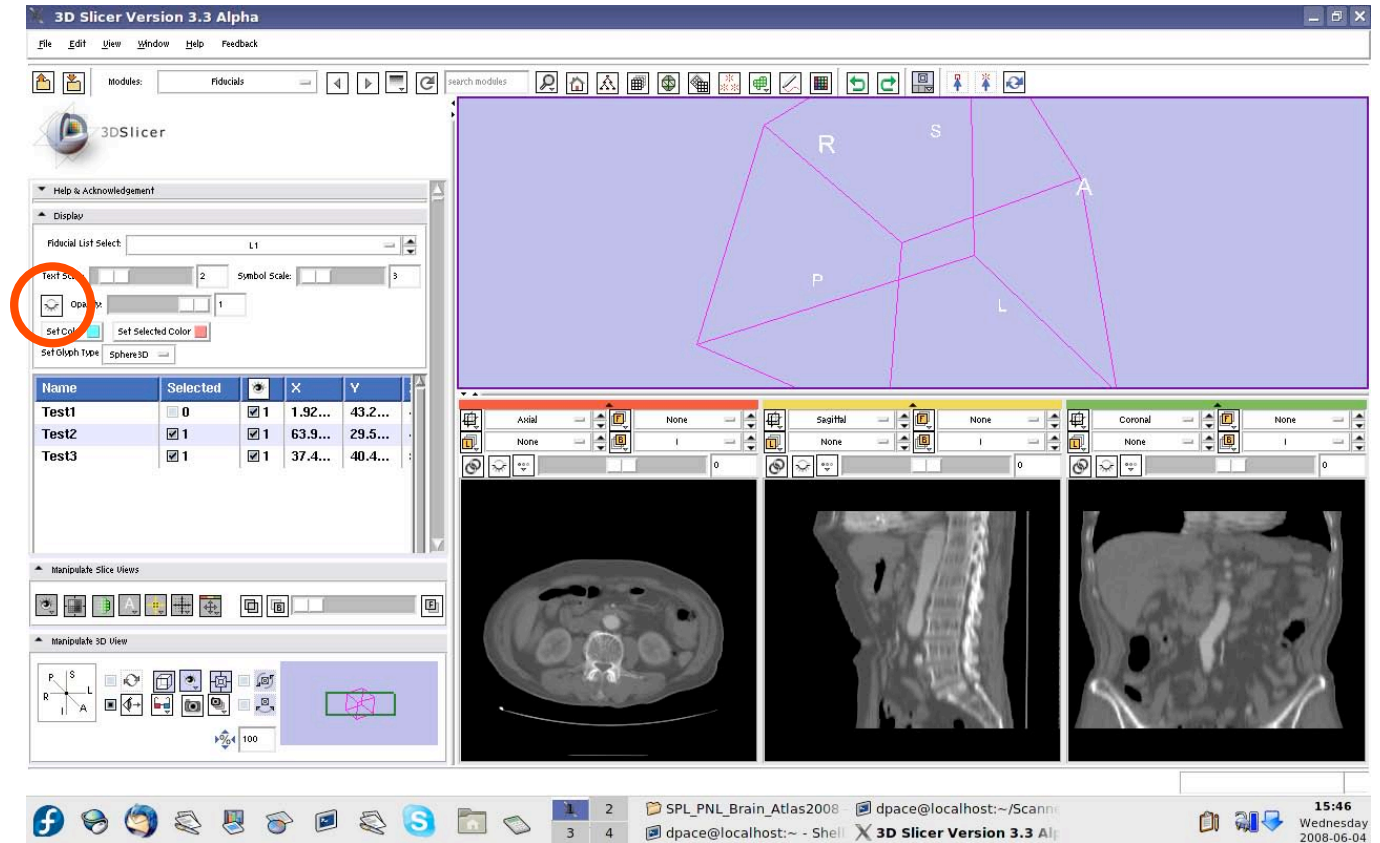
If you are using the abdominal atlas, open the Fiducials module





# Make the fiducials invisible

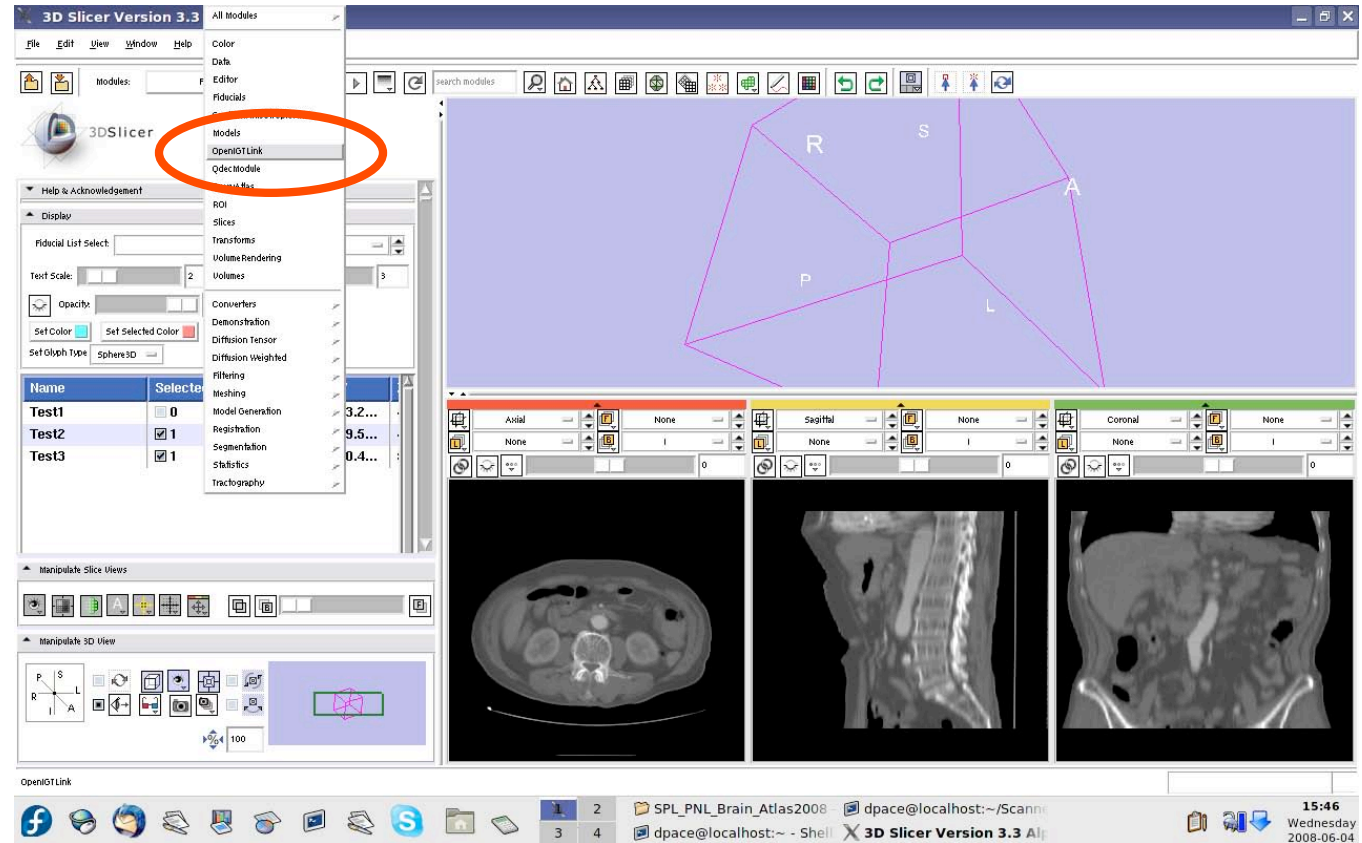
If you are using the abdominal atlas, turn off the visibility of the fiducials





# Set up the OpenIGTLink connection

Open the OpenIGTLink module



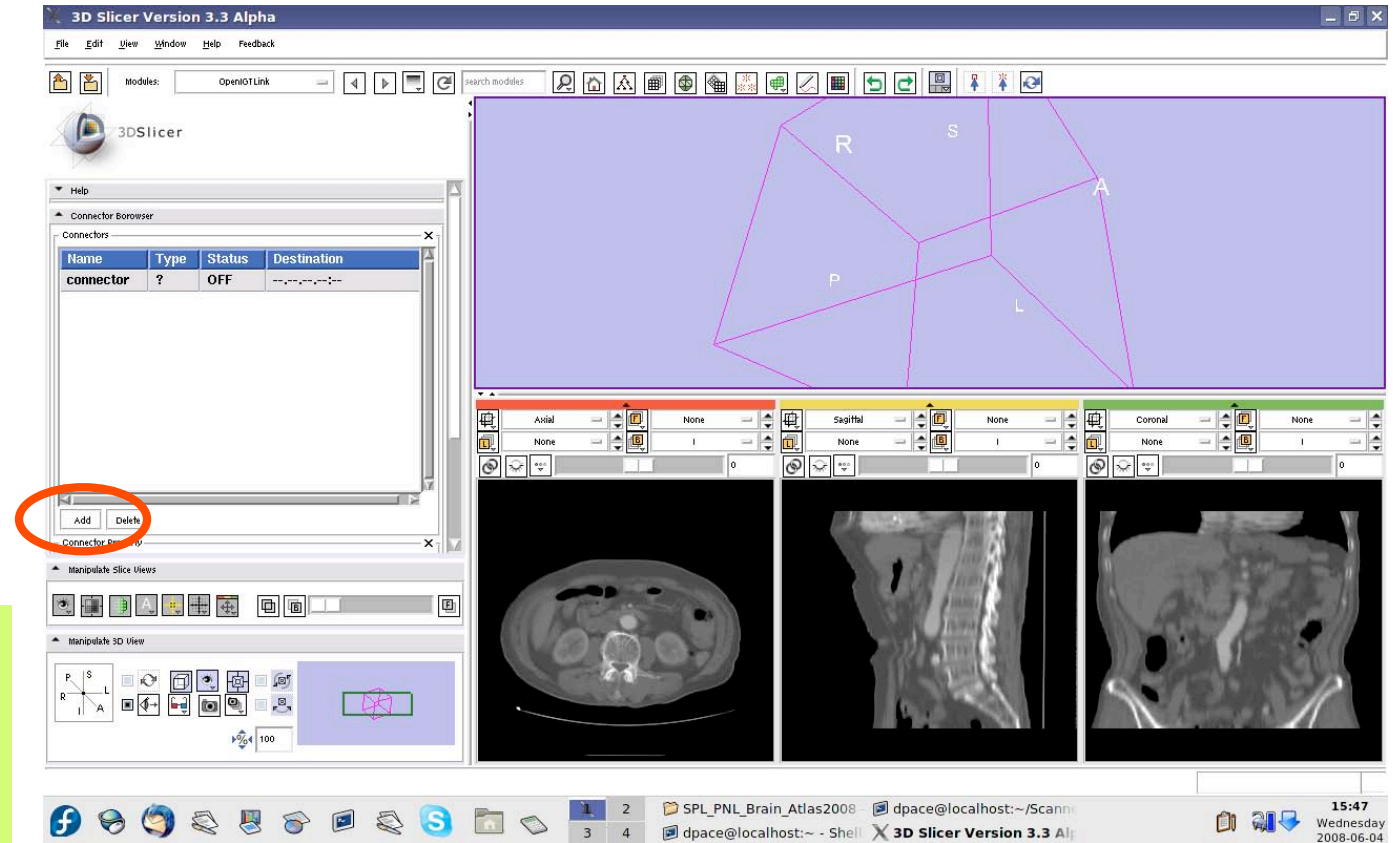




# Set up the OpenIGTLink connection

The Connectors pane shows the OpenIGTLink connections that Slicer3 is connected to

Add a new connection by clicking the “Add” button

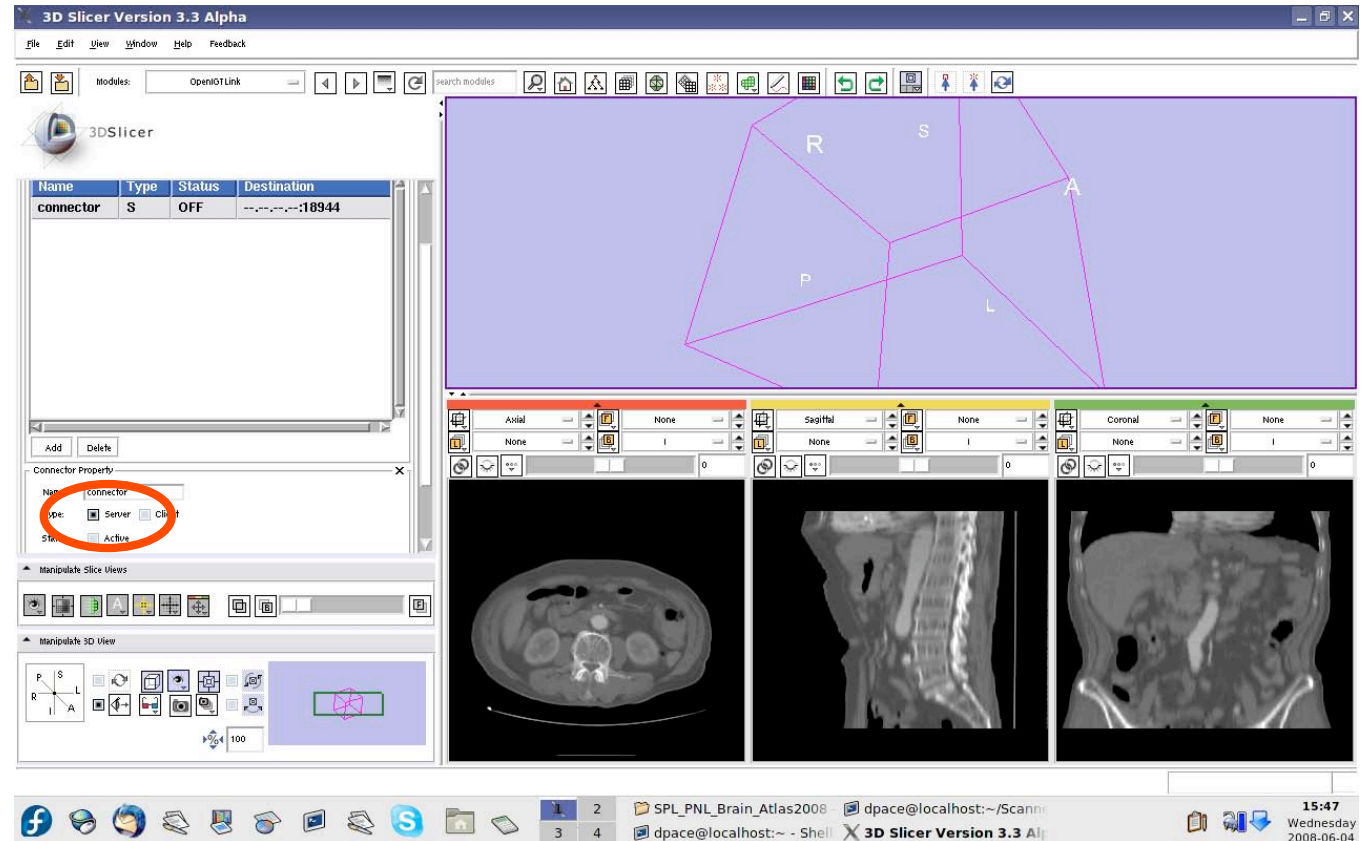




# Set up the OpenIGTLink connection

Set Slicer3 to be the server by clicking on the Server box

Note that the connector type is now set to "S" instead of "?"



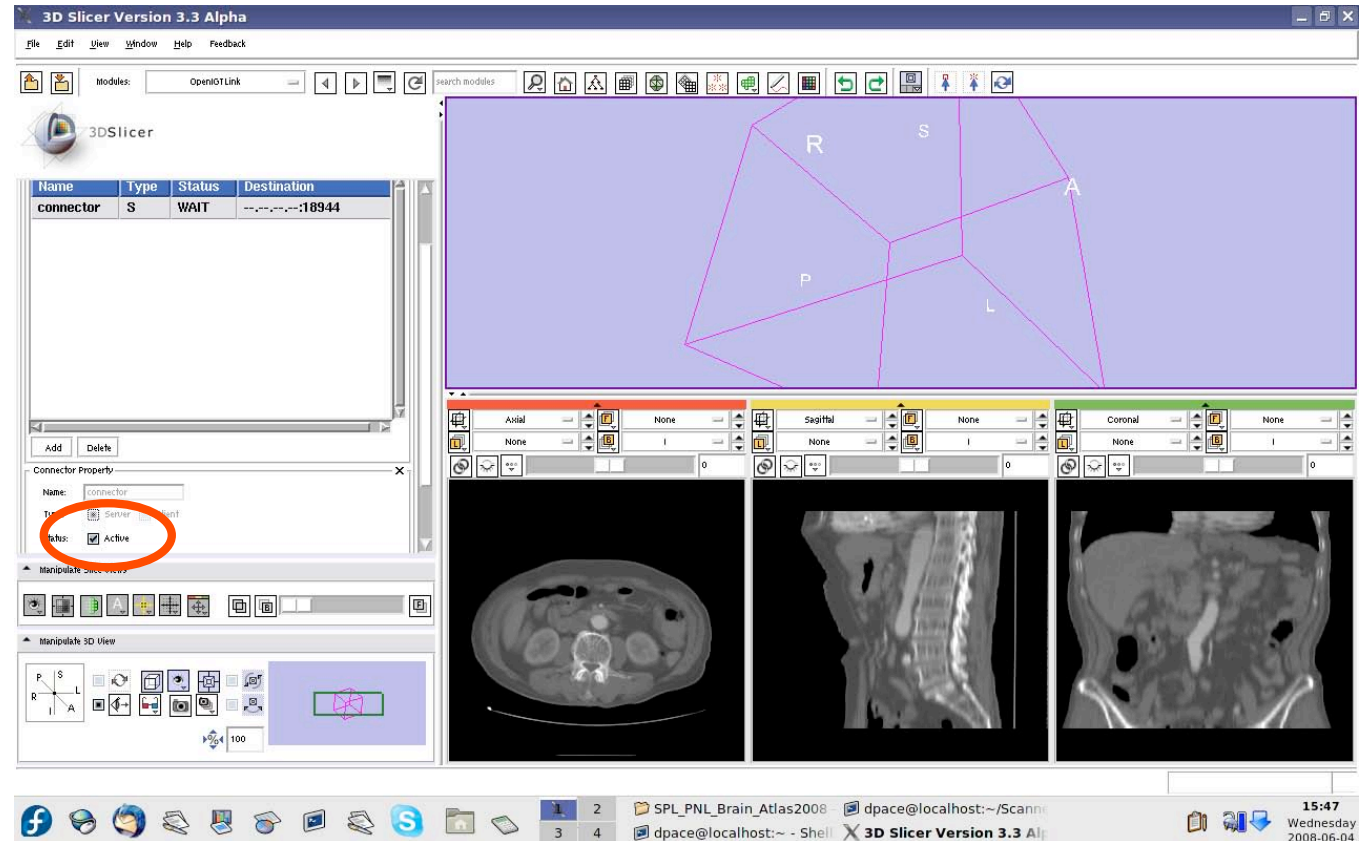




# Set up the OpenIGTLink connection

Make the connection active by clicking on the “Active” button

Note that the connector status is now set to “WAIT” instead of “OFF”



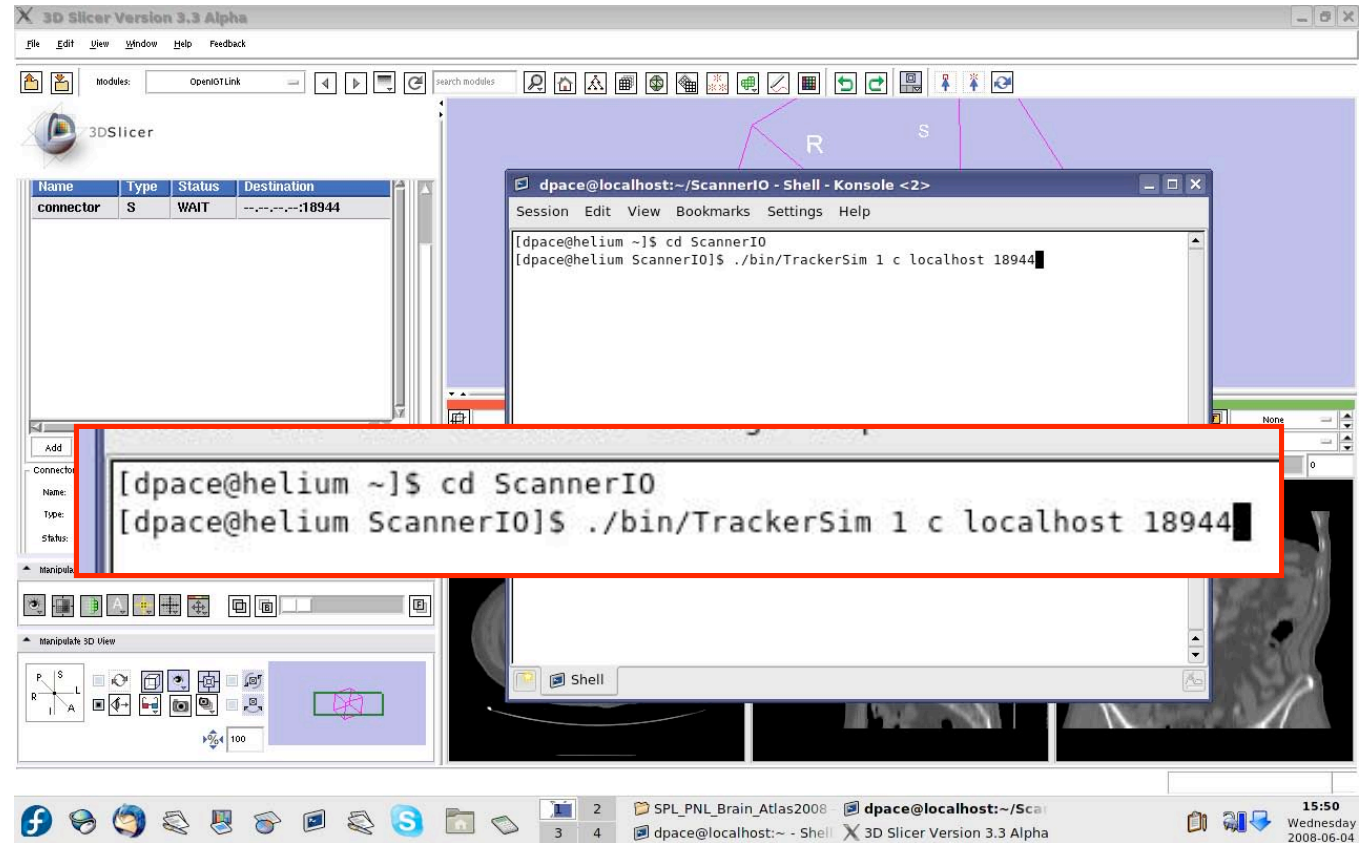


# Start the tracker simulator

LINUX and MAC:

Run the TrackerSim program:

- 1 = number of frames per second
- c = TrackerSim is the client
- localhost = the host name
- 18944 = the port number

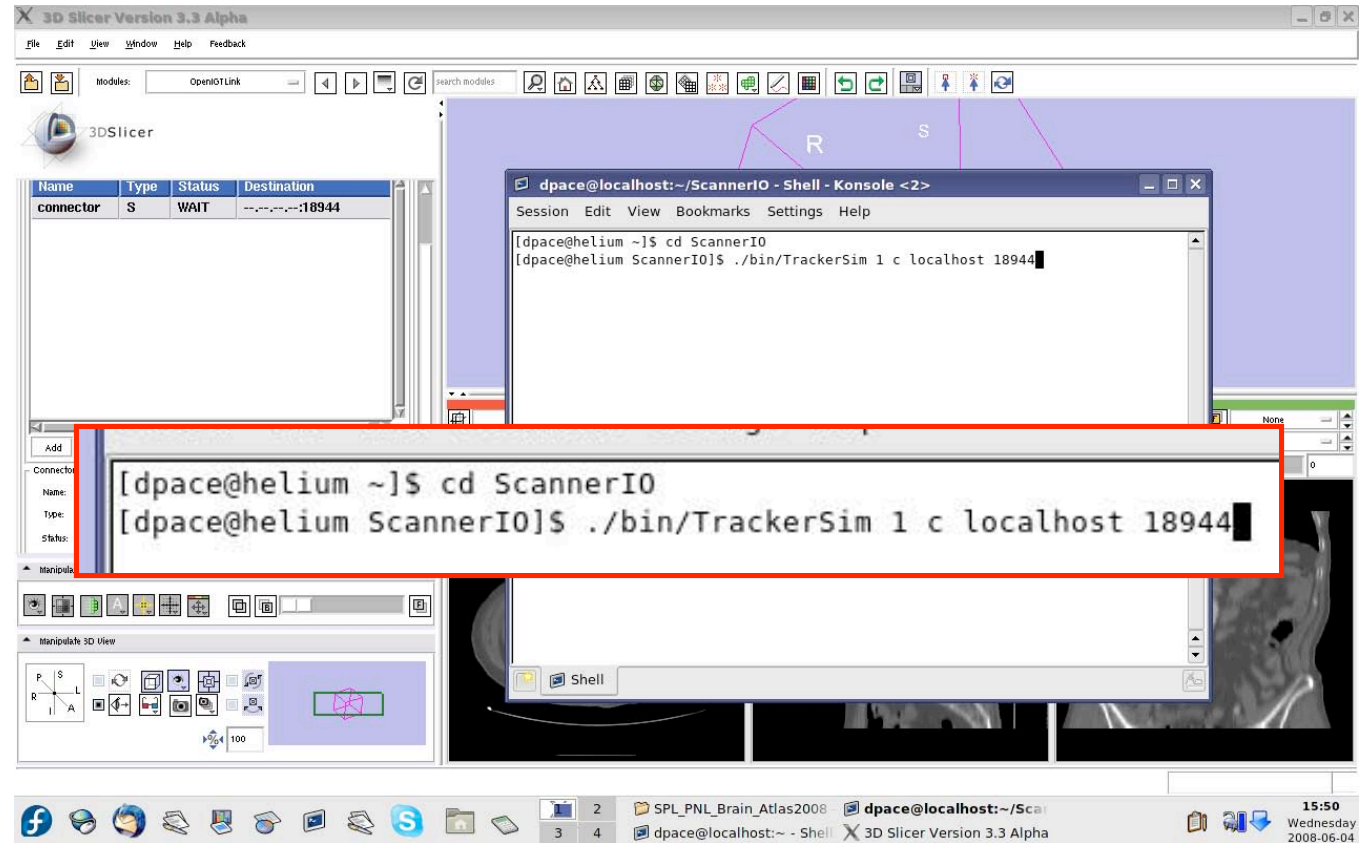




# Start the tracker simulator

## WINDOWS:

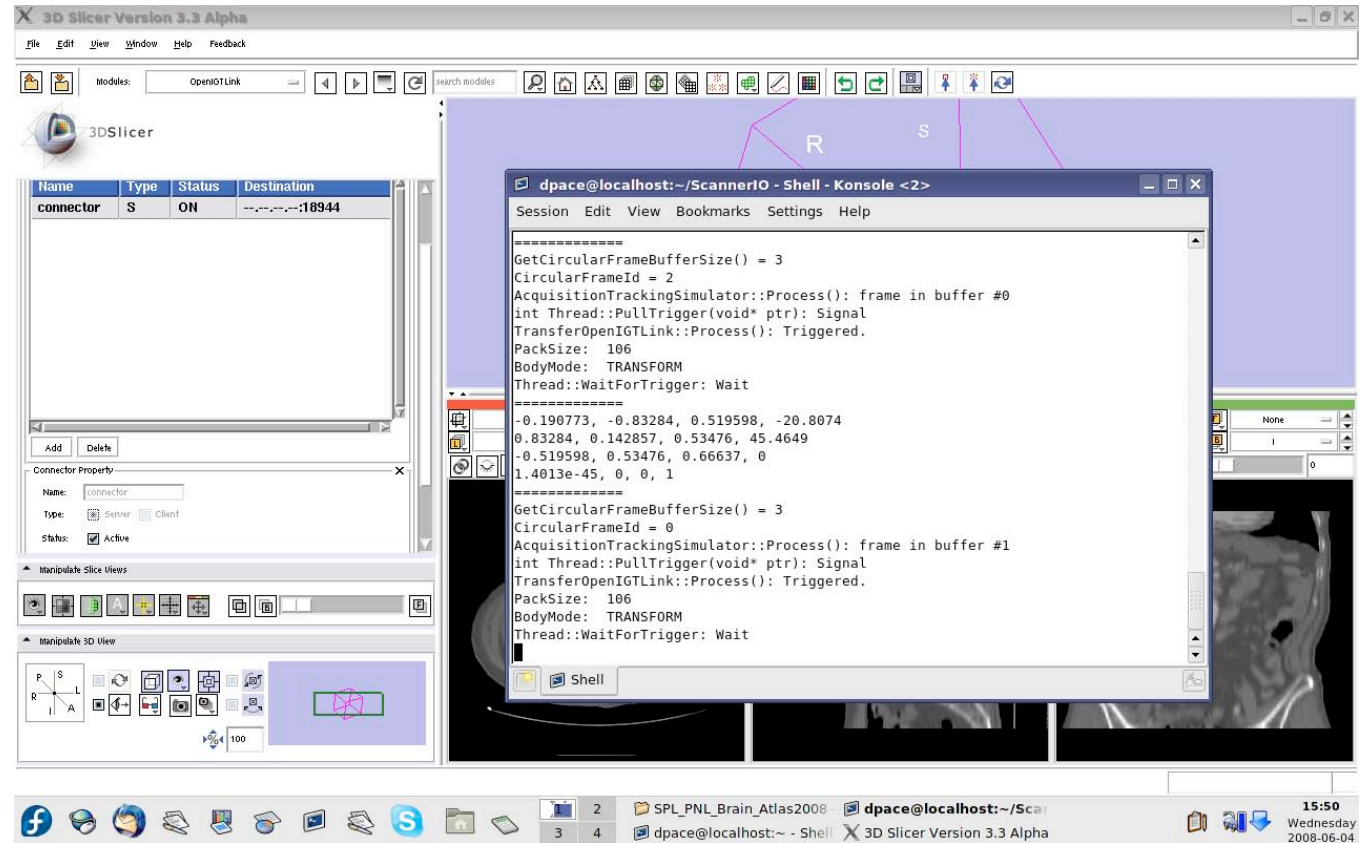
Run the RunTrackerSim program by double clicking on it





# Start the tracker simulator

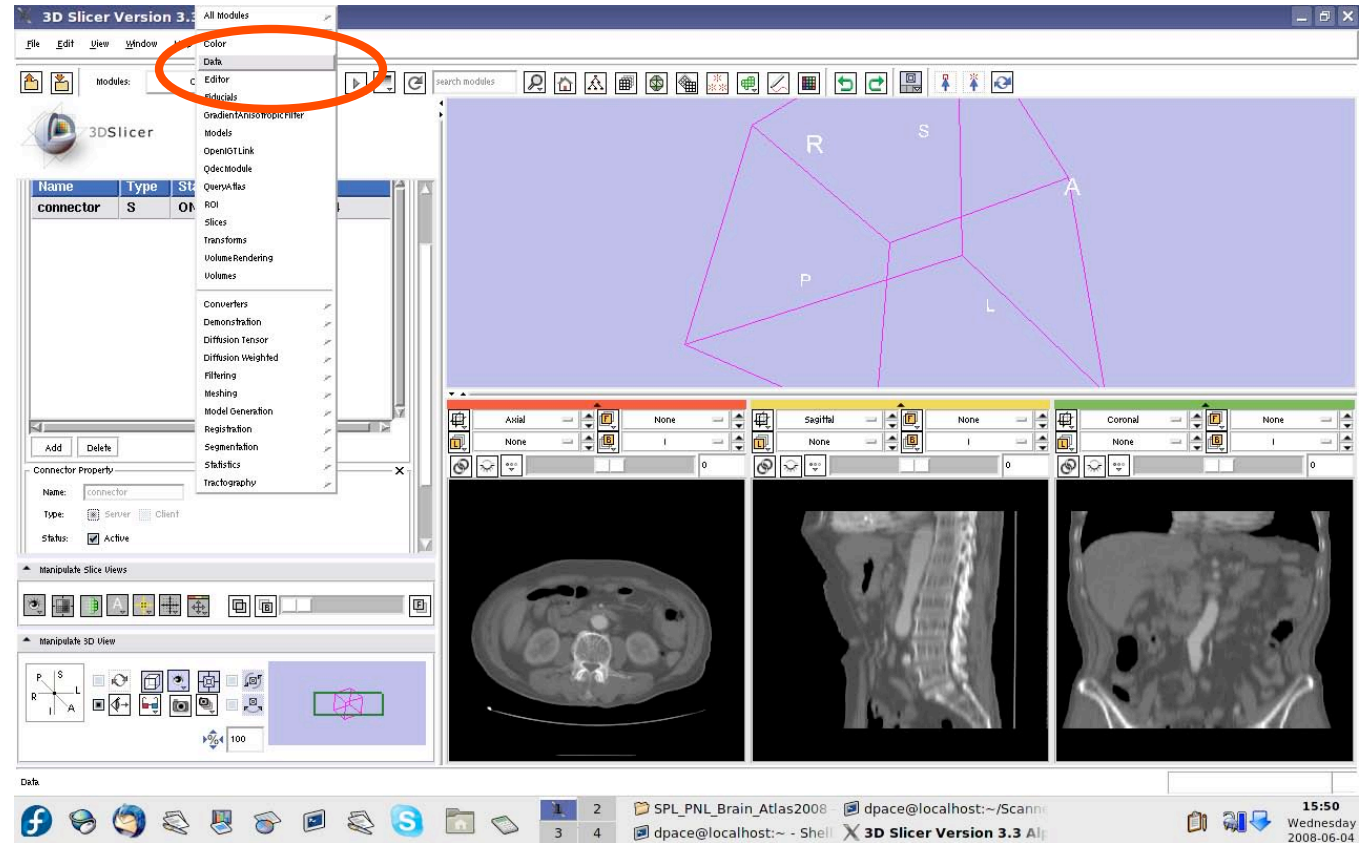
The transforms being sent are written to the terminal





# Start the tracker simulator

Open the  
Data module

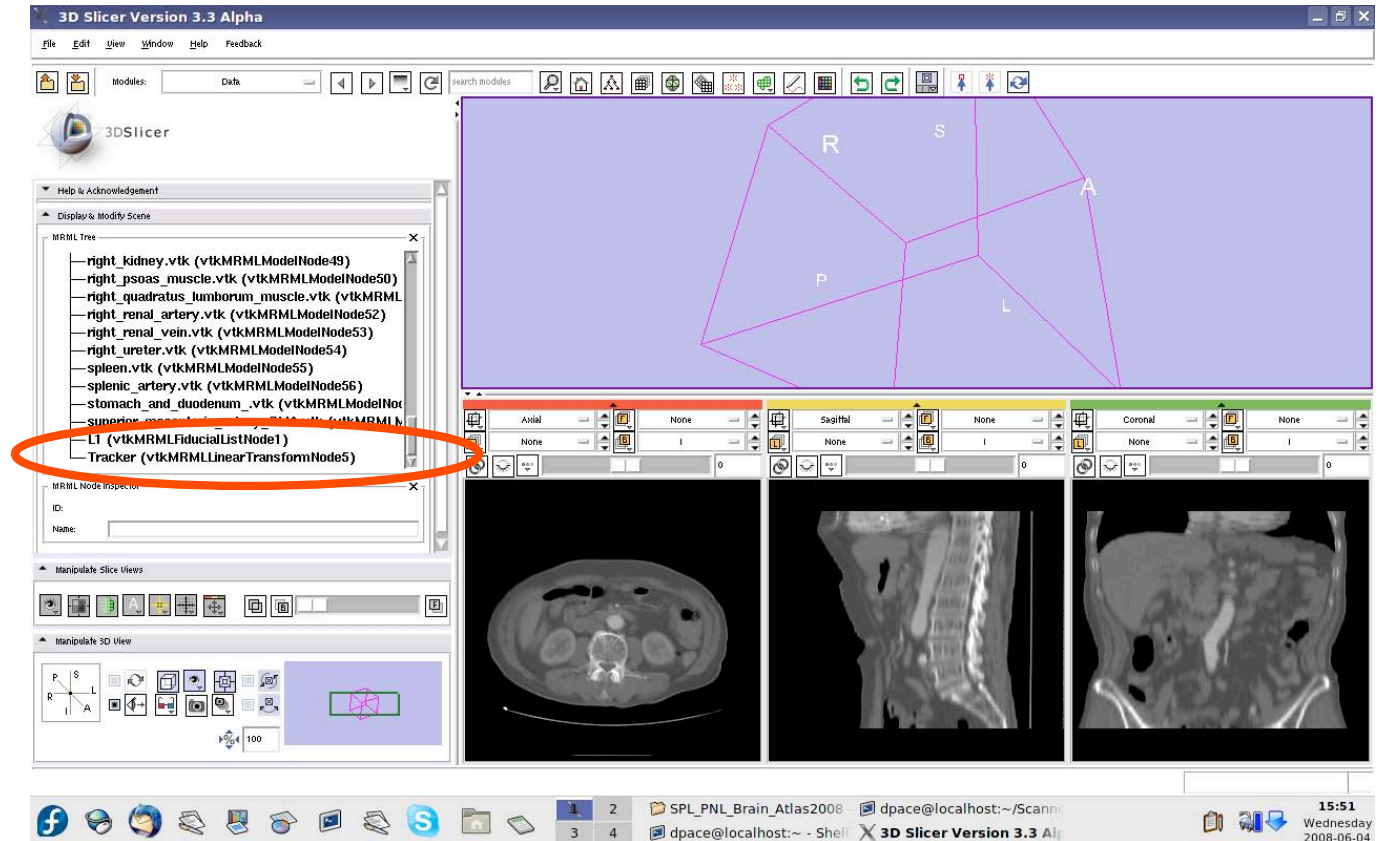






# Start the tracker simulator

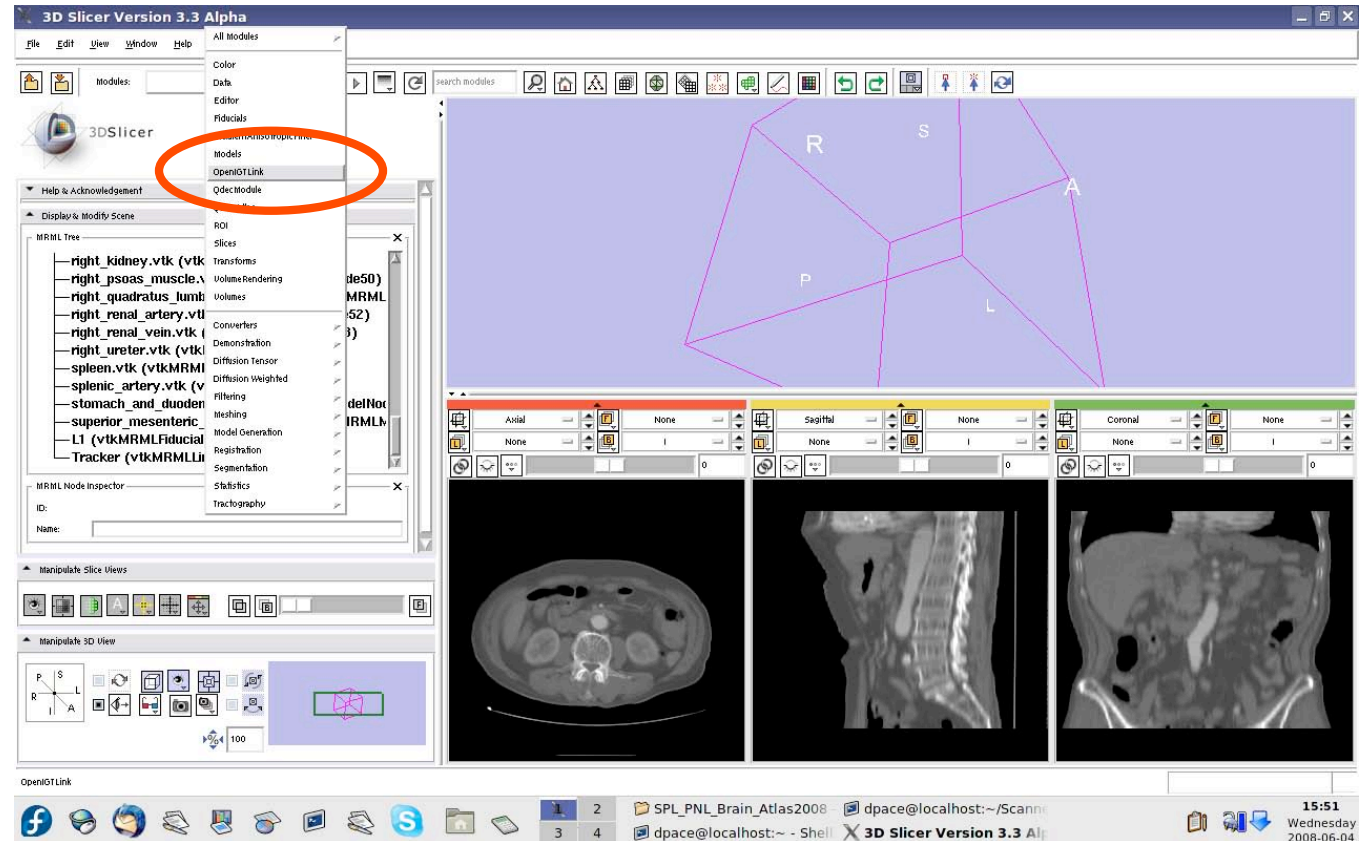
The new tracker node is a transform node - you can see it at the bottom of the MRML tree





# Start the tracker simulator

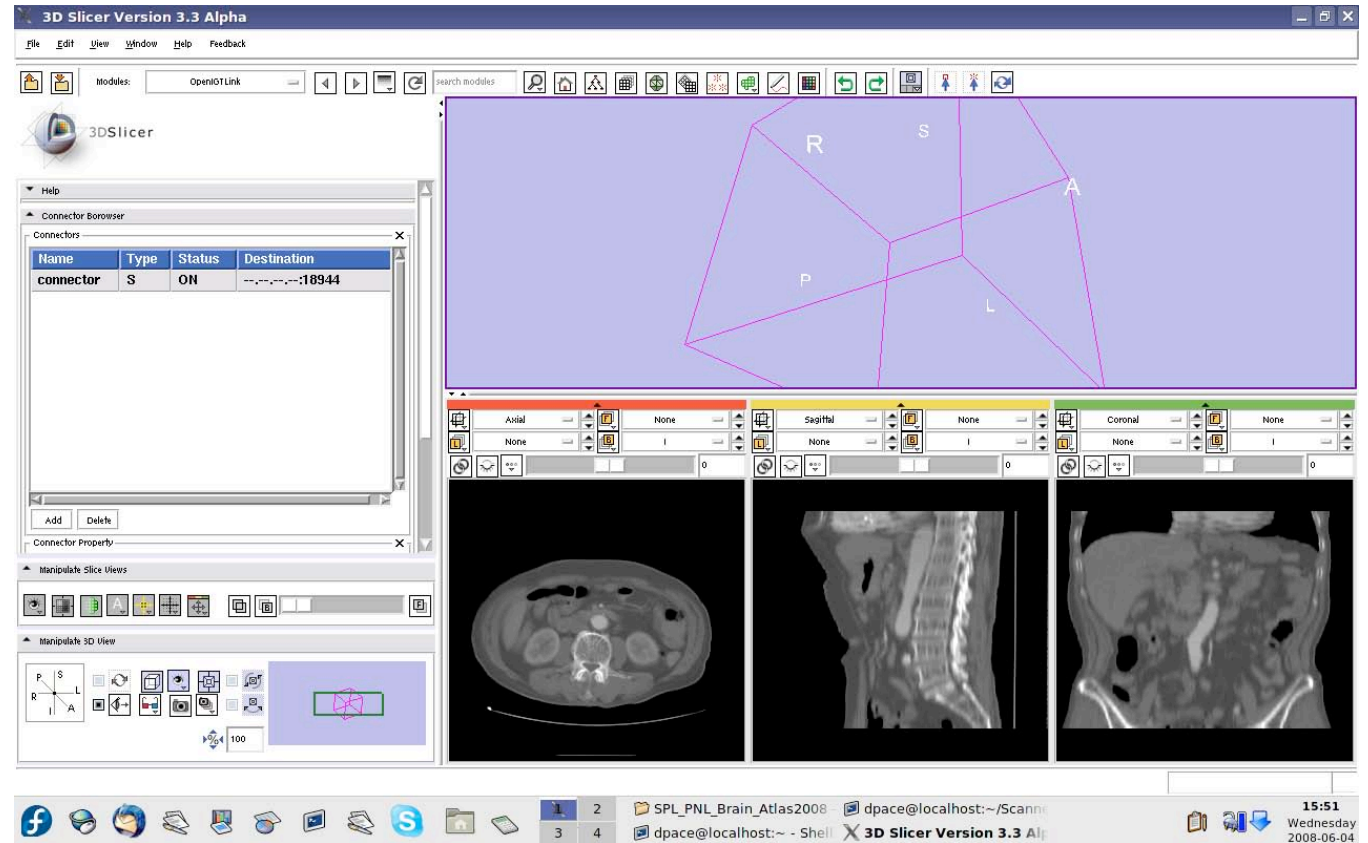
Open the  
OpenIGTLink  
module





# Start the tracker simulator

Note that the connector status is now set to “ON” instead of “WAIT”



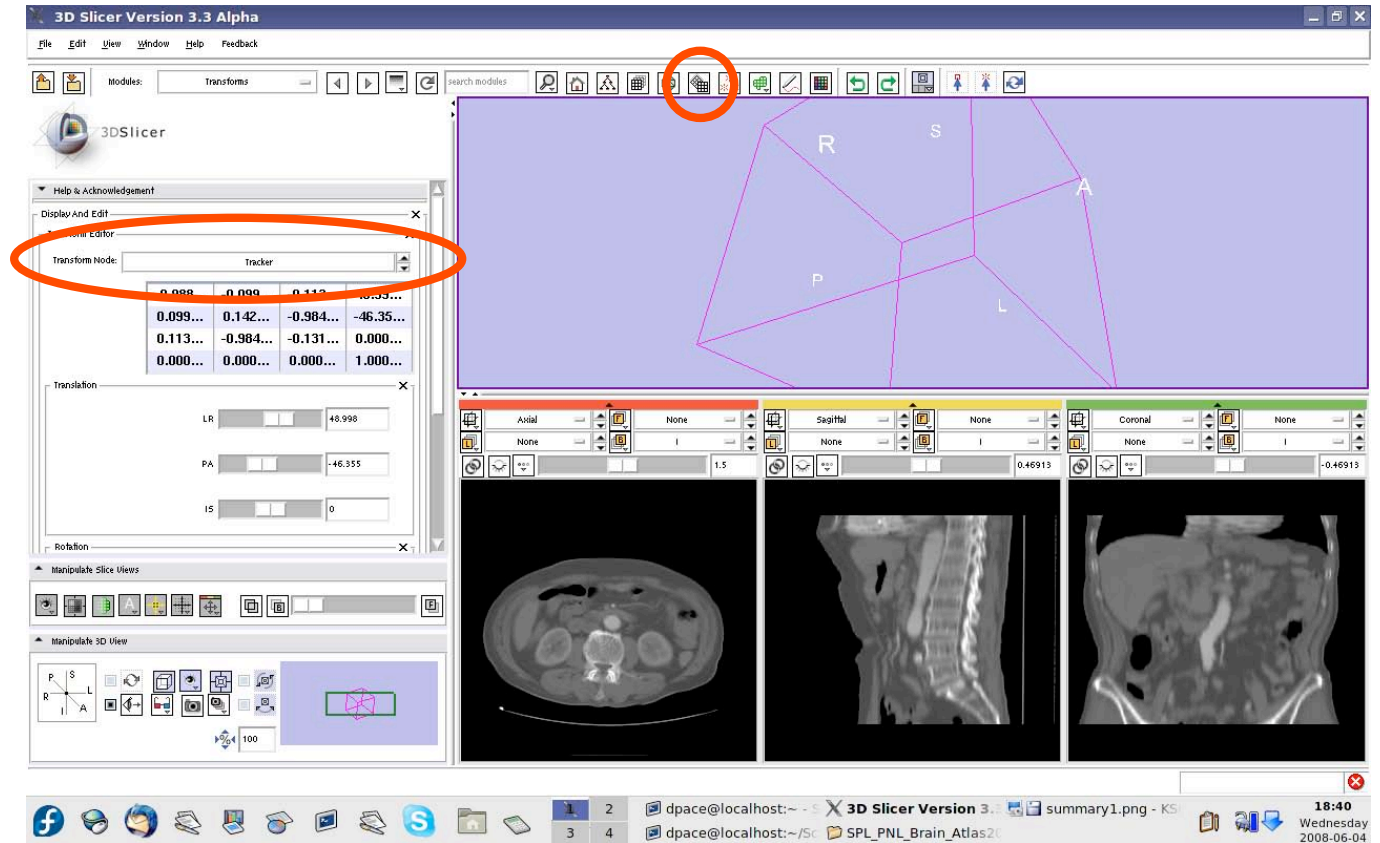




# Start the tracker simulator

Open the Transforms module

Click on the new Tracker transform to see the changing transformation matrix



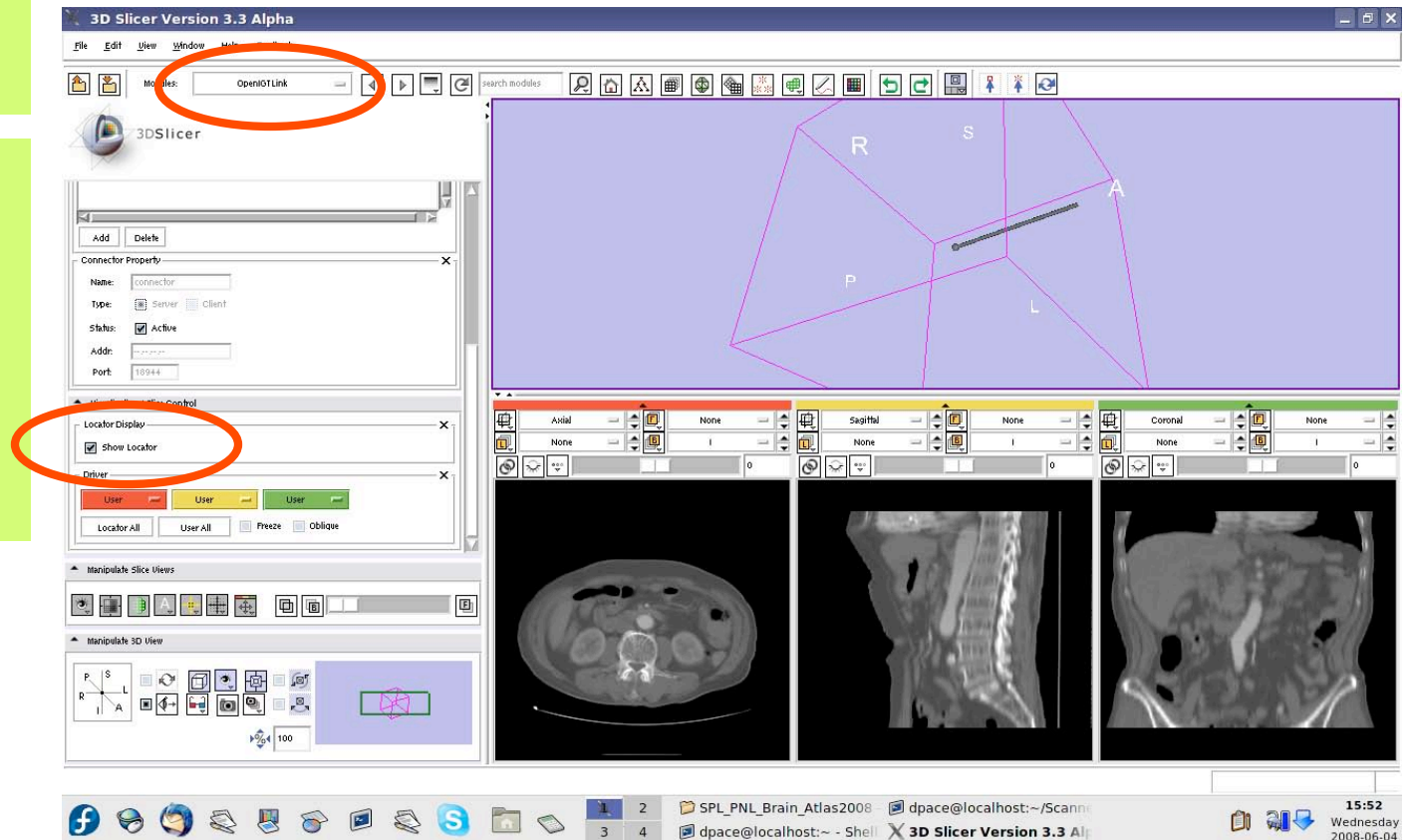


# Show the transform using the locator

Open the  
OpenIGTLink  
module

In the  
Visualization/  
Slice Control  
pane, click the  
“Show Locator”  
button

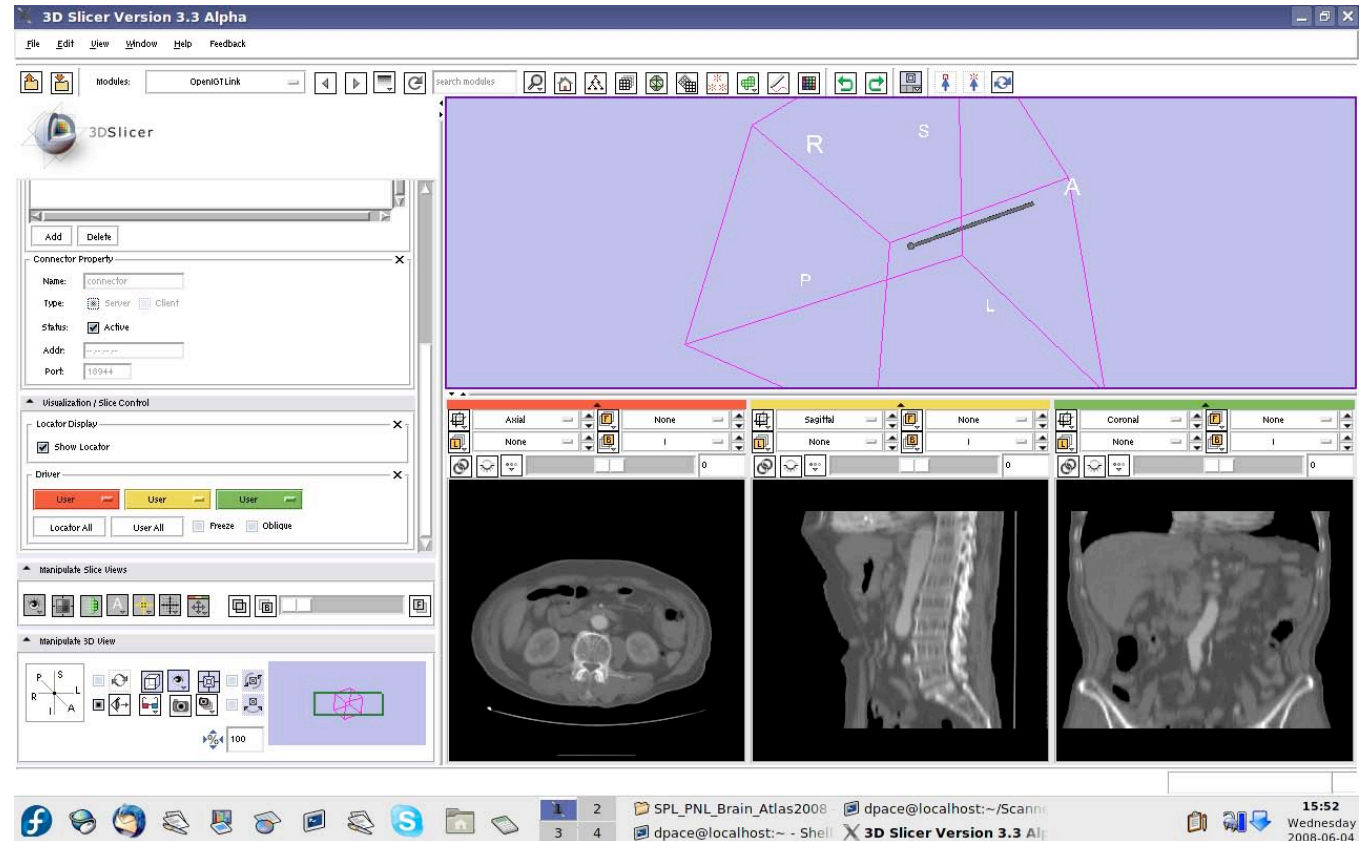
If the locator does  
not appear, make  
sure that the  
IGTLocator model  
is set to “visible” in  
the Models module





# Show the transform using the locator

The round end shows the simulated tool's position, and the cylinder shows the simulated tool's orientation

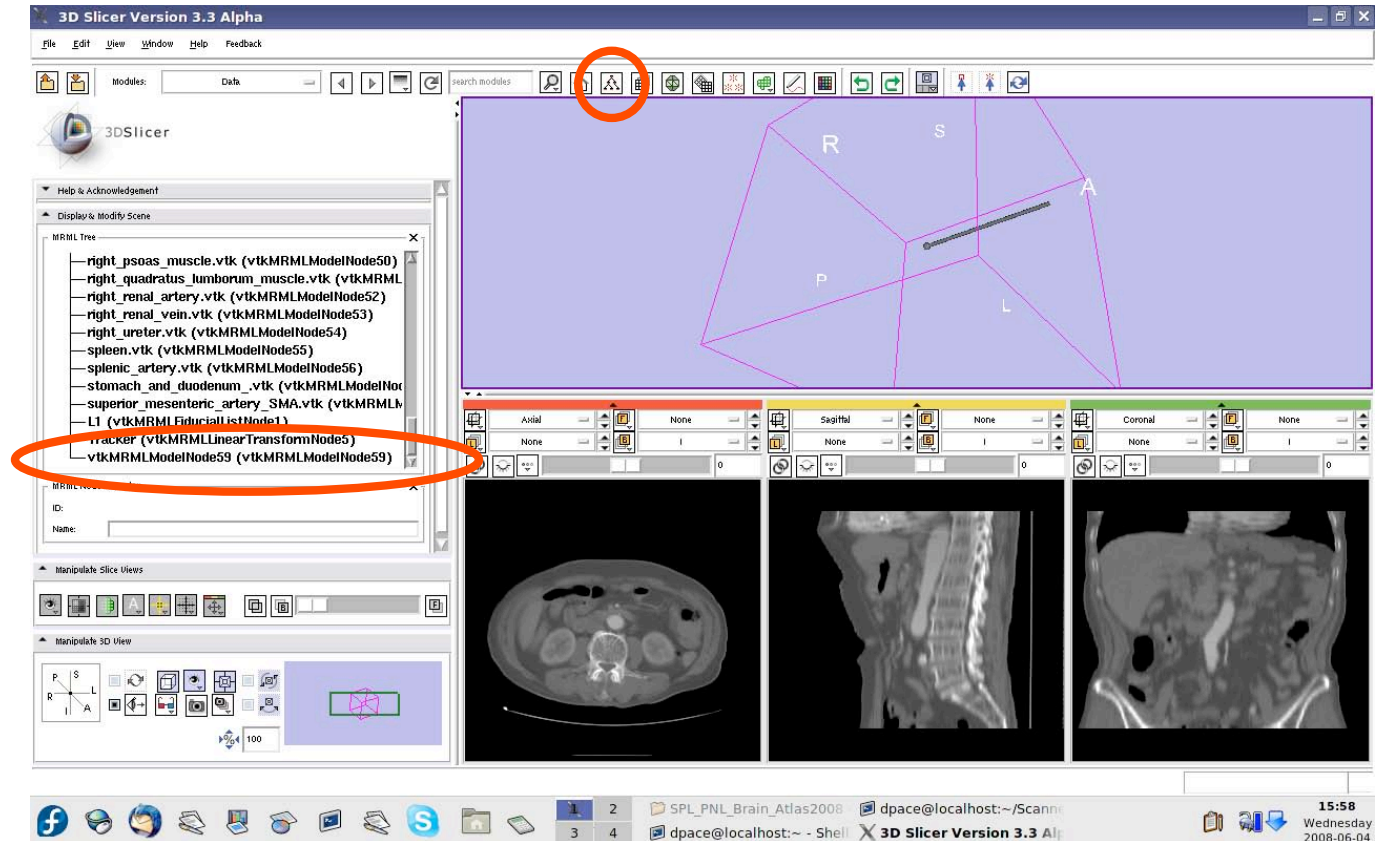




# Show the transform using the locator

Open the Data module

The new locator node is a model node at the bottom of the MRML tree

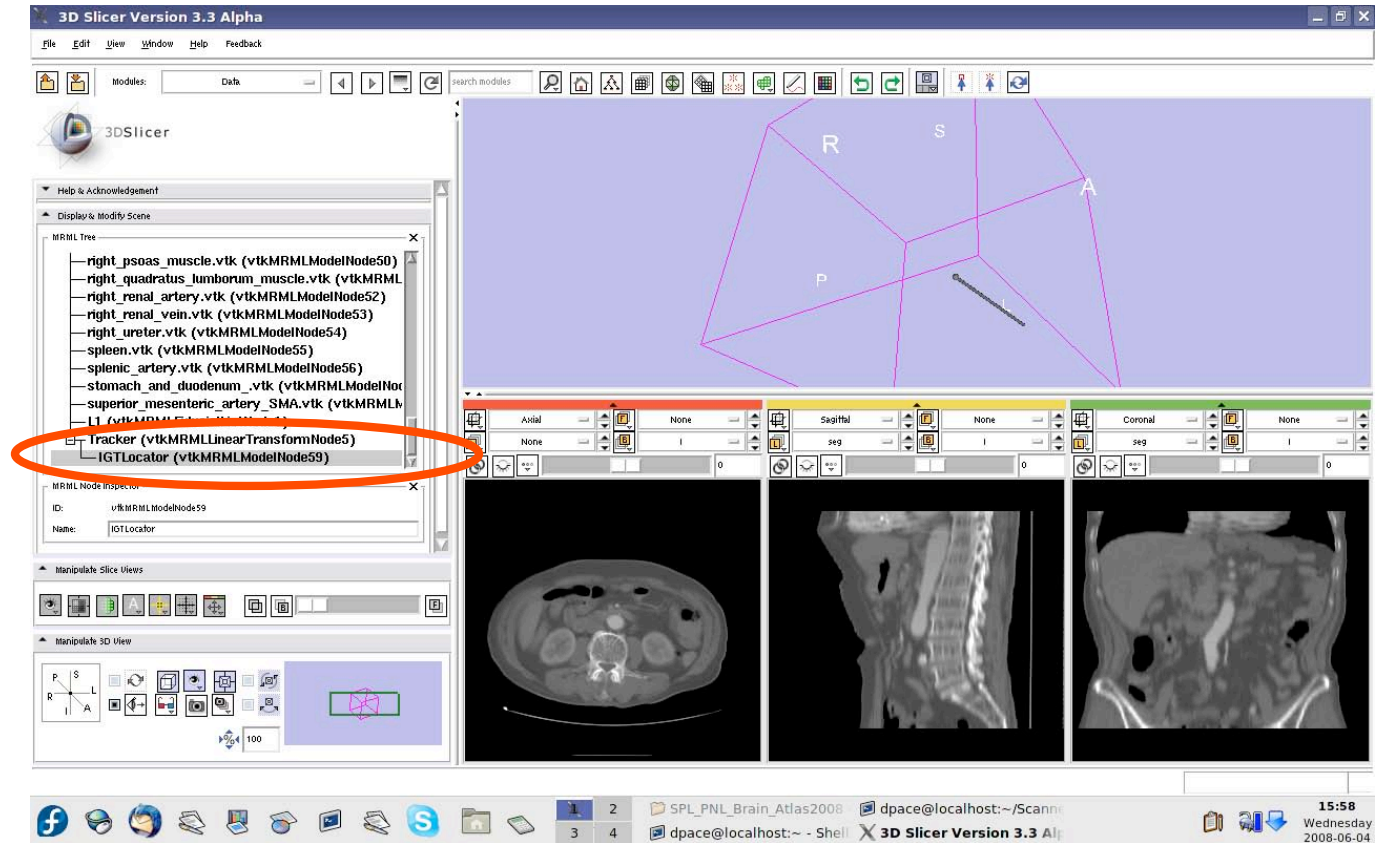




# Show the transform using the locator

Drag the locator node under the Tracker node

The Tracker transform is now applied to the locator model - it will move according to the transforms from the tracker simulator



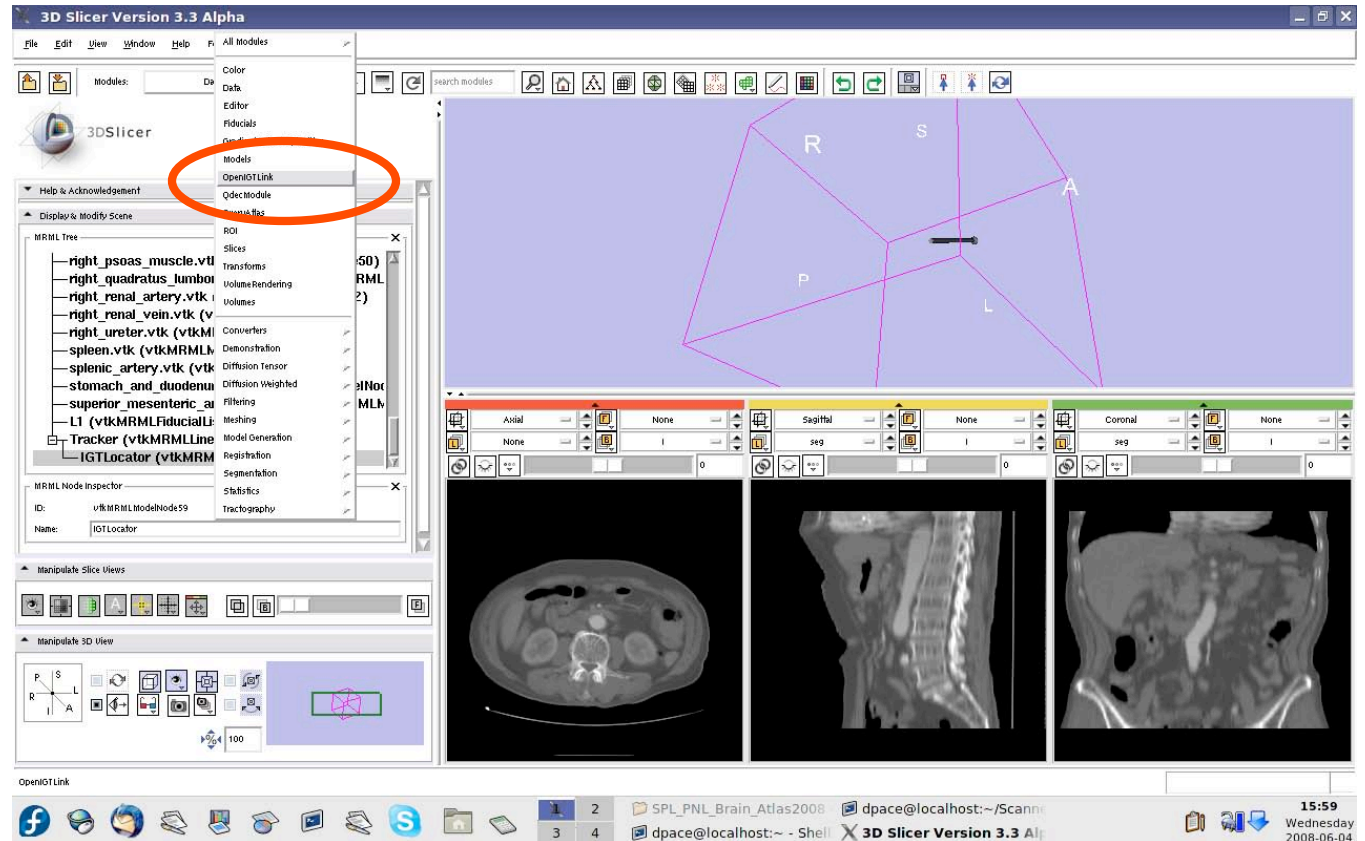




# Show the transform using a model

Other objects, such as models or images, can be moved according to the tracking transforms

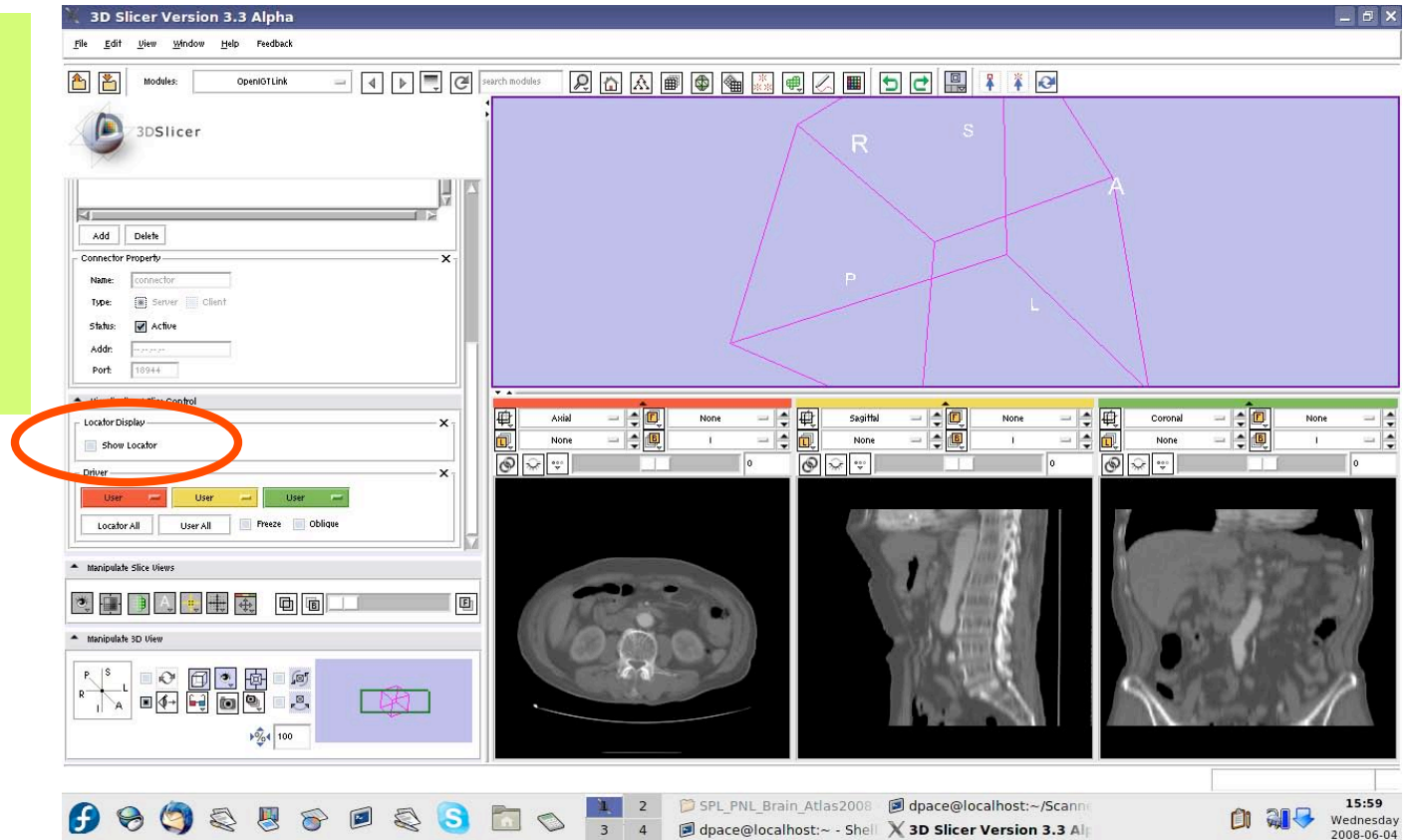
Open the OpenIGTLink module





# Show the transform using a model

In the Visualization/  
Slice Control  
pane, turn off  
the locator

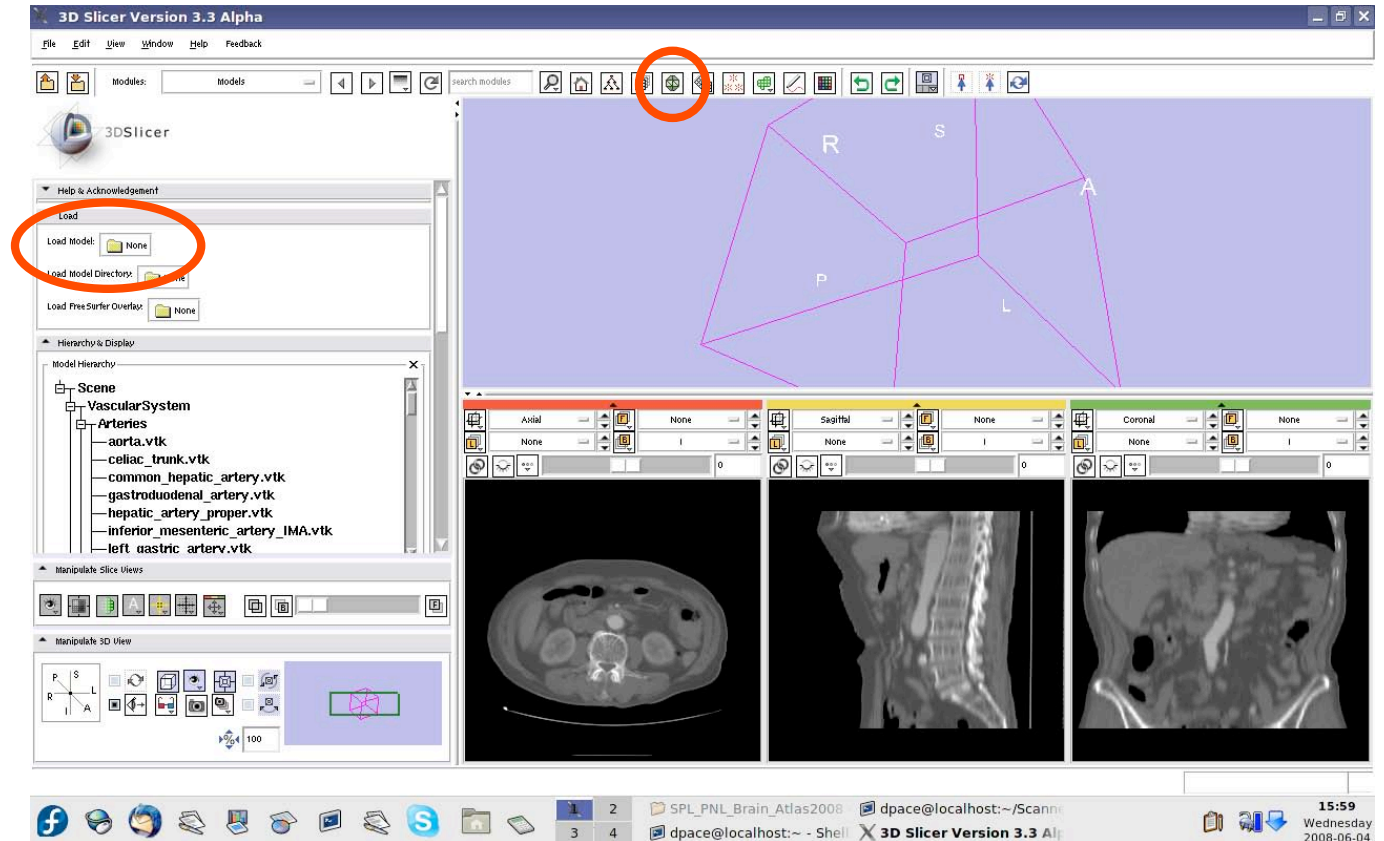




# Show the transform using a model

Open the Models module

Click on the folder icon to load a model

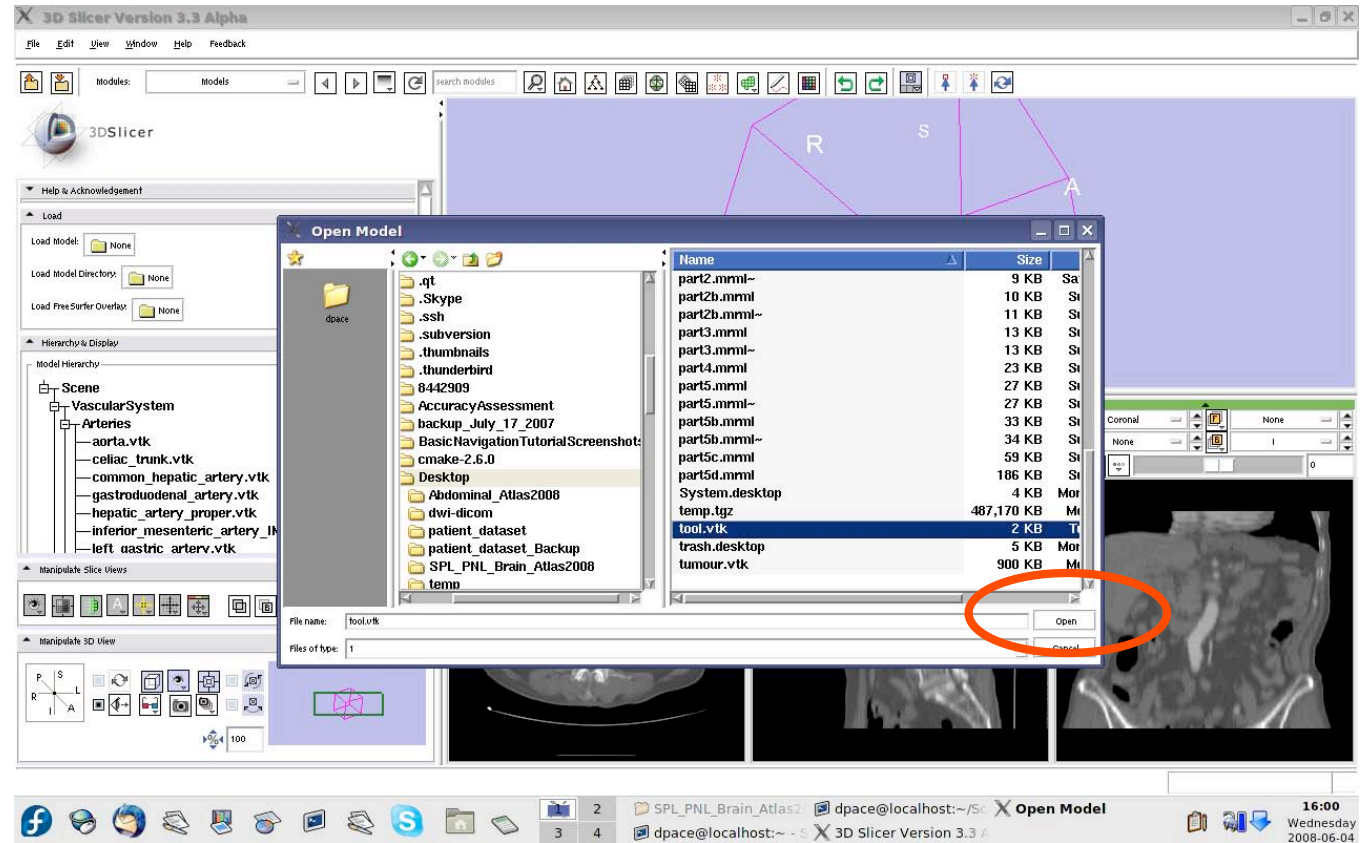






# Show the transform using a model

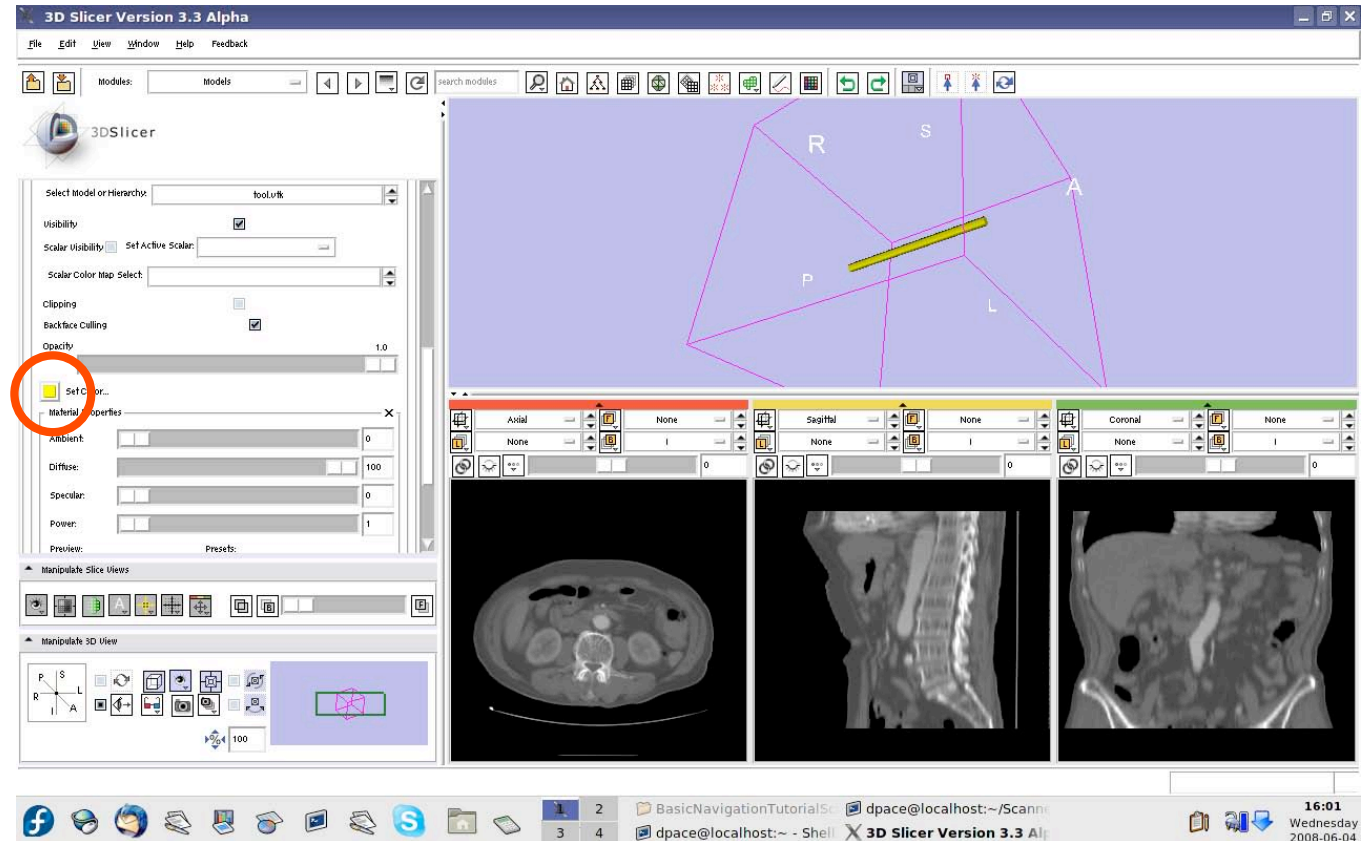
Click on  
tool.vtk and  
then click  
“Open”





# Show the transform using a model

In the Models module, change the colour of the model to yellow



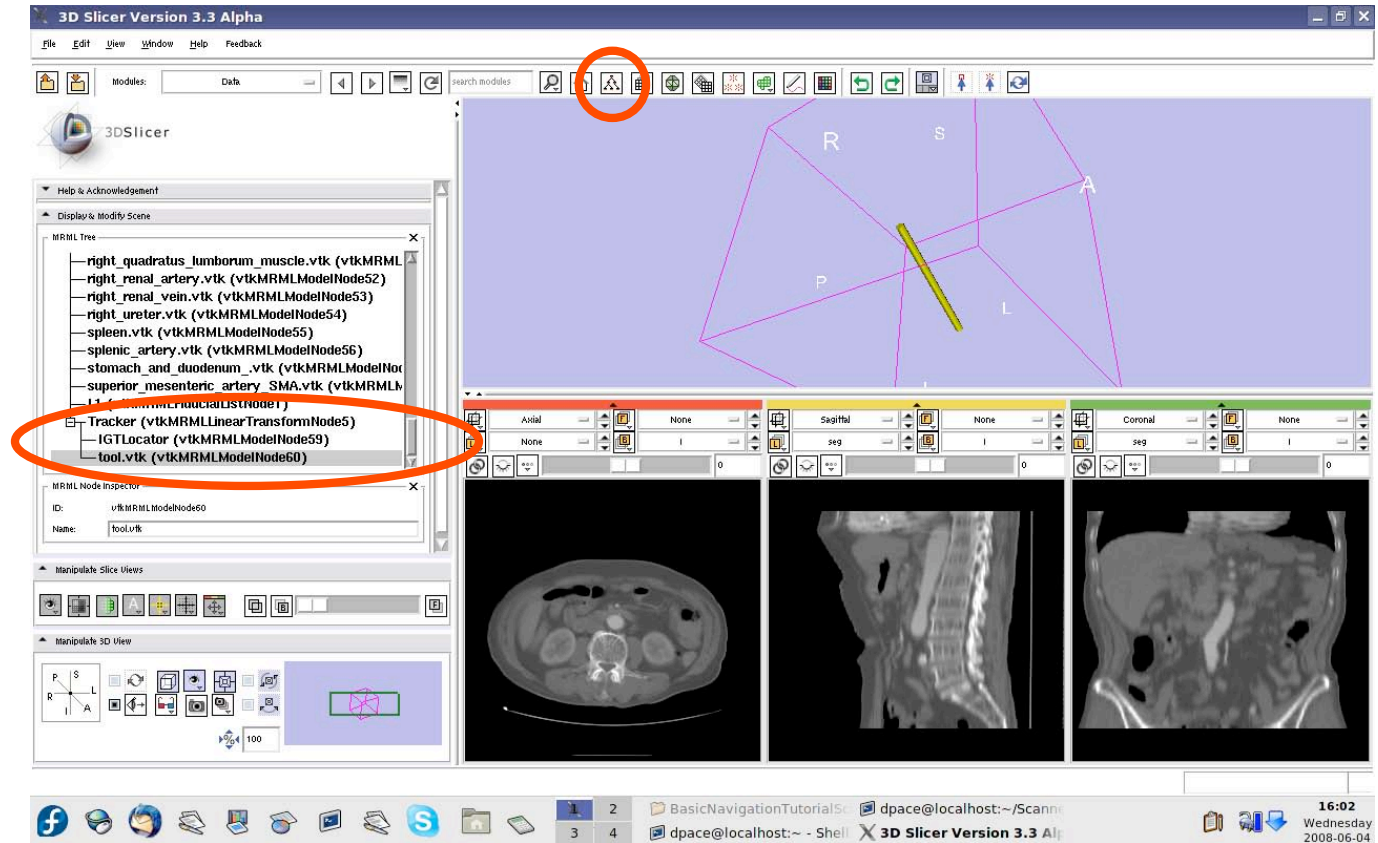


# Show the transform using a model

Open the Data module

Drag the tool.vtk node under the Tracker node

The Tracker transform is now applied to the tool model - it will move according to the transforms from the tracker simulator

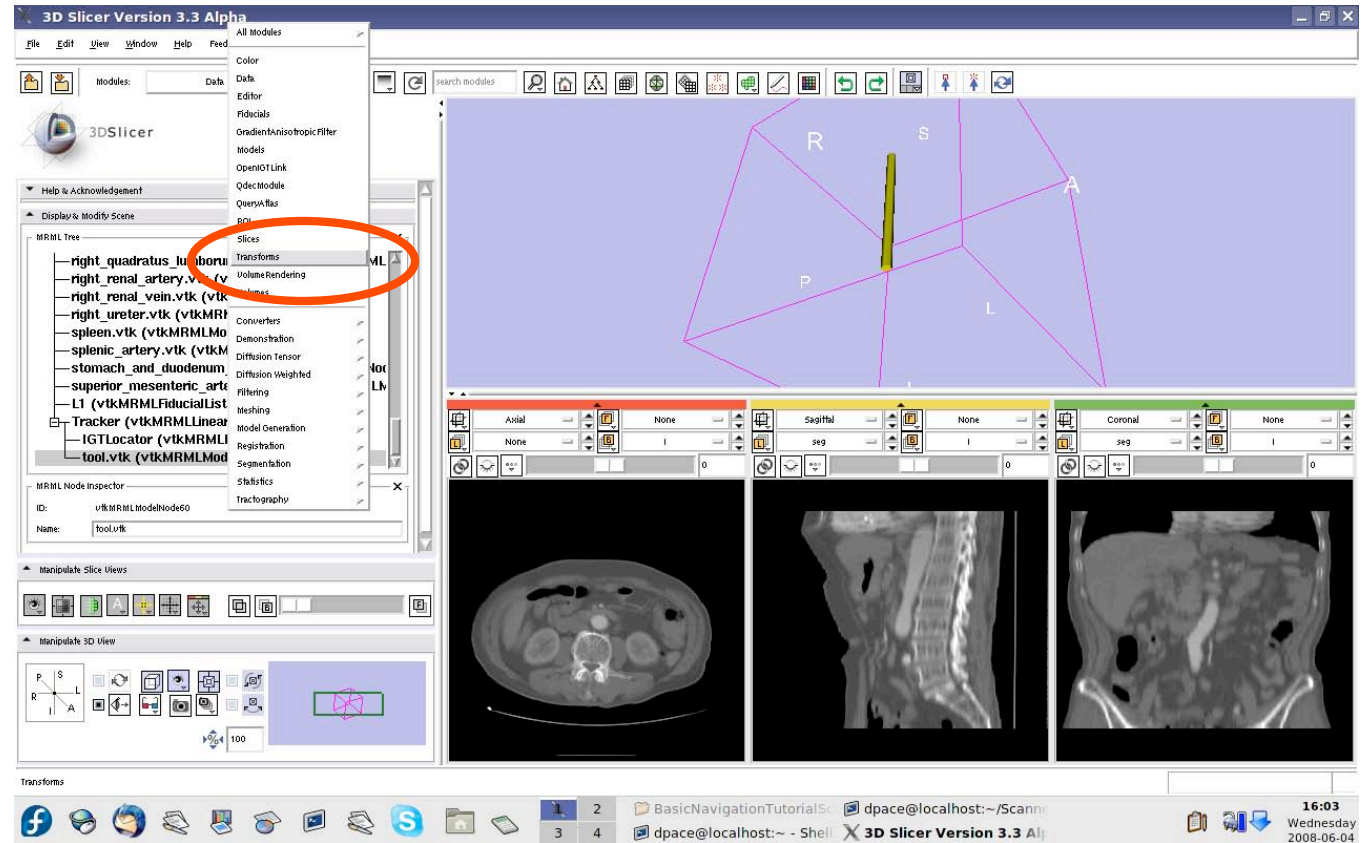




# Add a calibration matrix

Transforms can be multiplied together - we will incorporate an additional translation

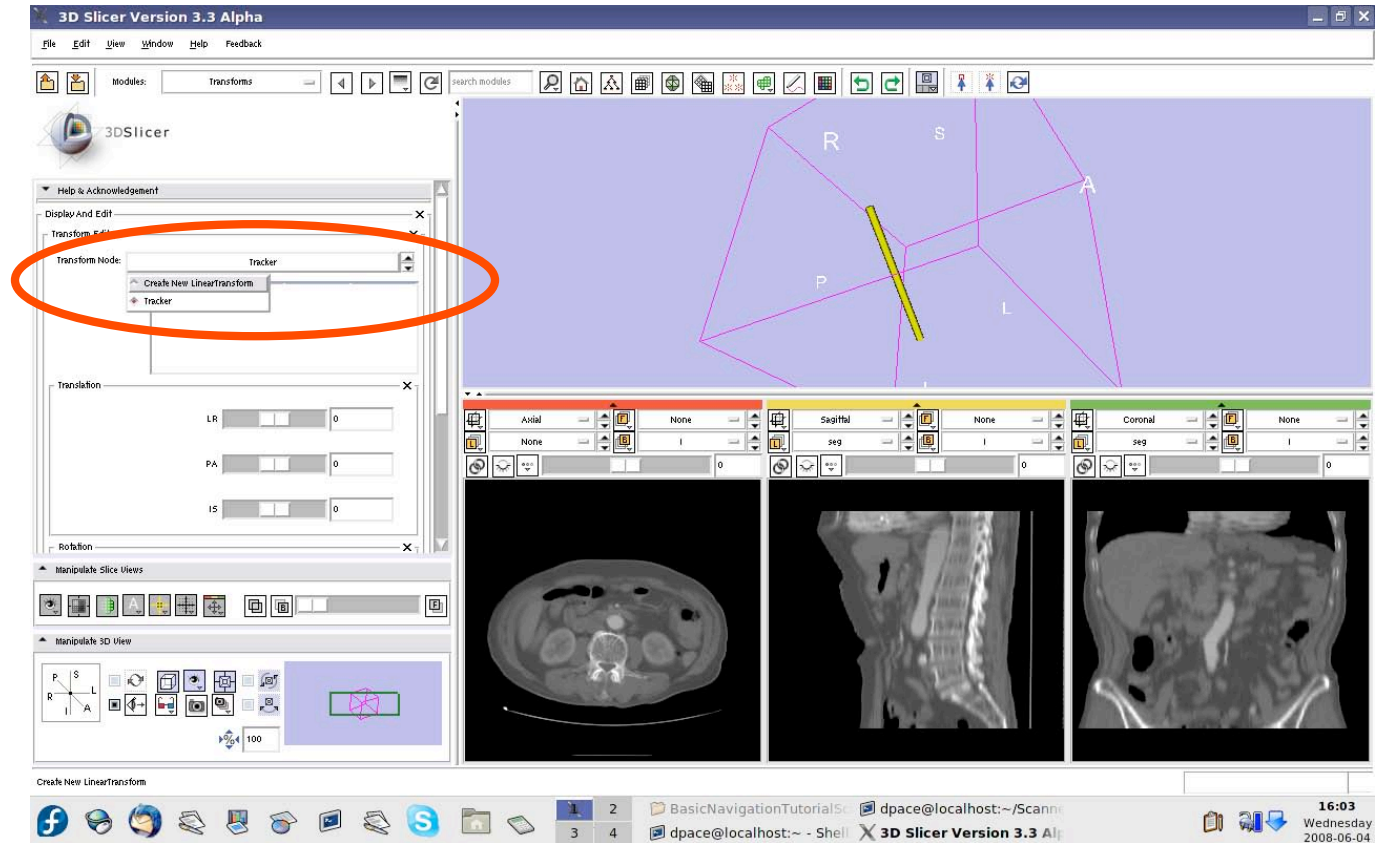
Open the Transforms module





# Add a calibration matrix

Add a new transform node



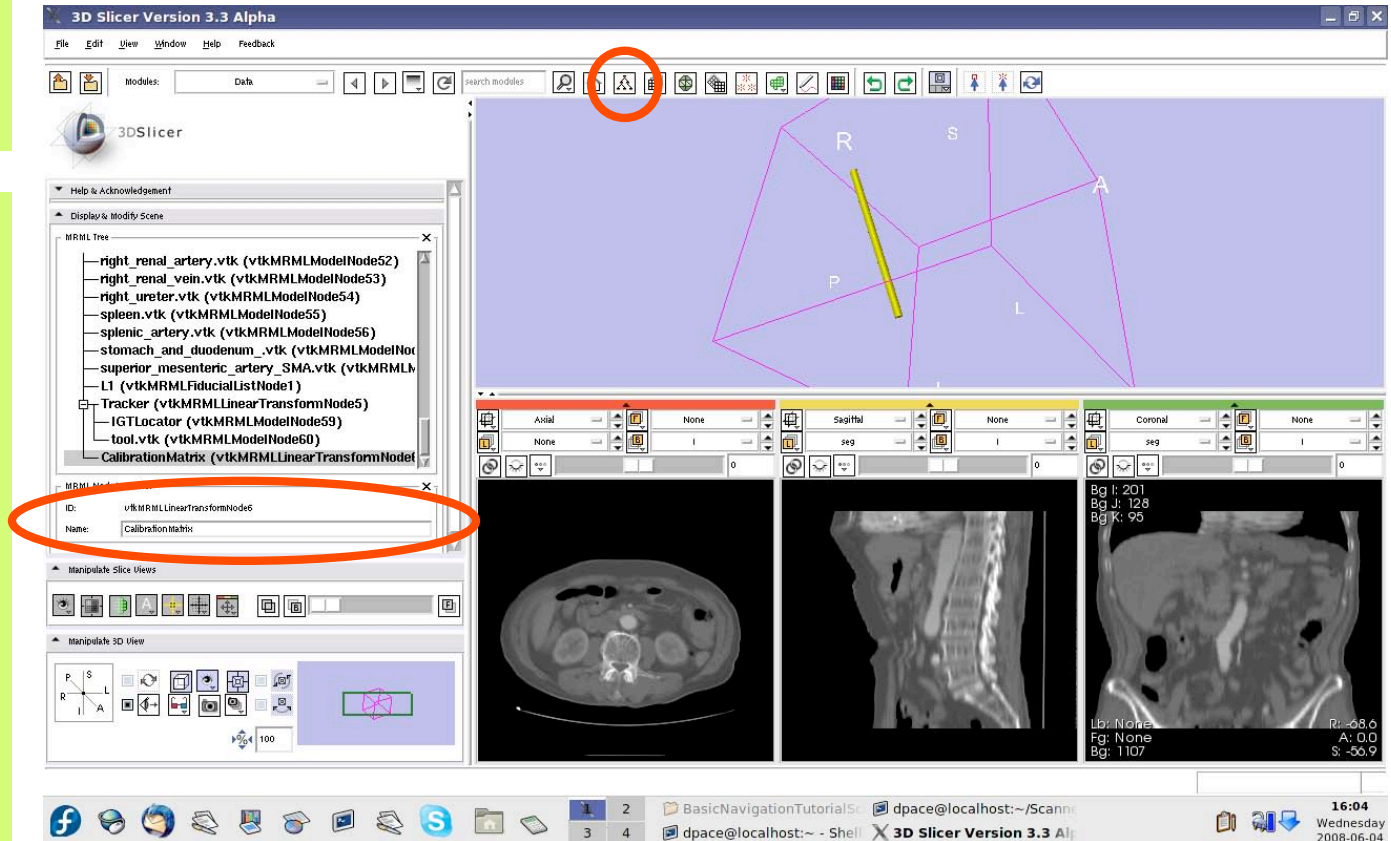




# Add a calibration matrix

Open the Data module

Rename the new transform to “Calibration Matrix” by selecting it and then changing the name in the MRML node inspector

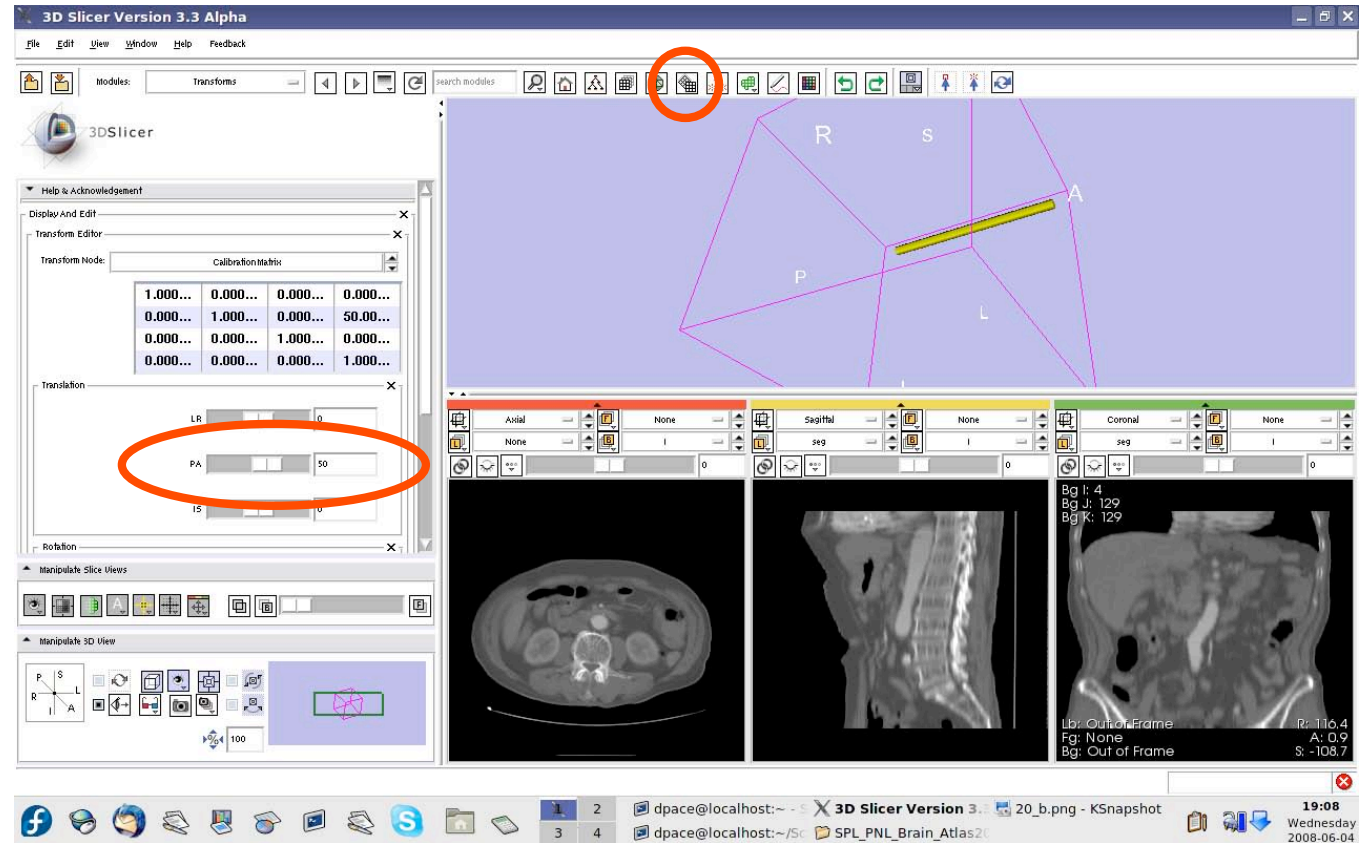




# Add a calibration matrix

Open the Transforms module

Set the PA (posterior-anterior) translation to 50



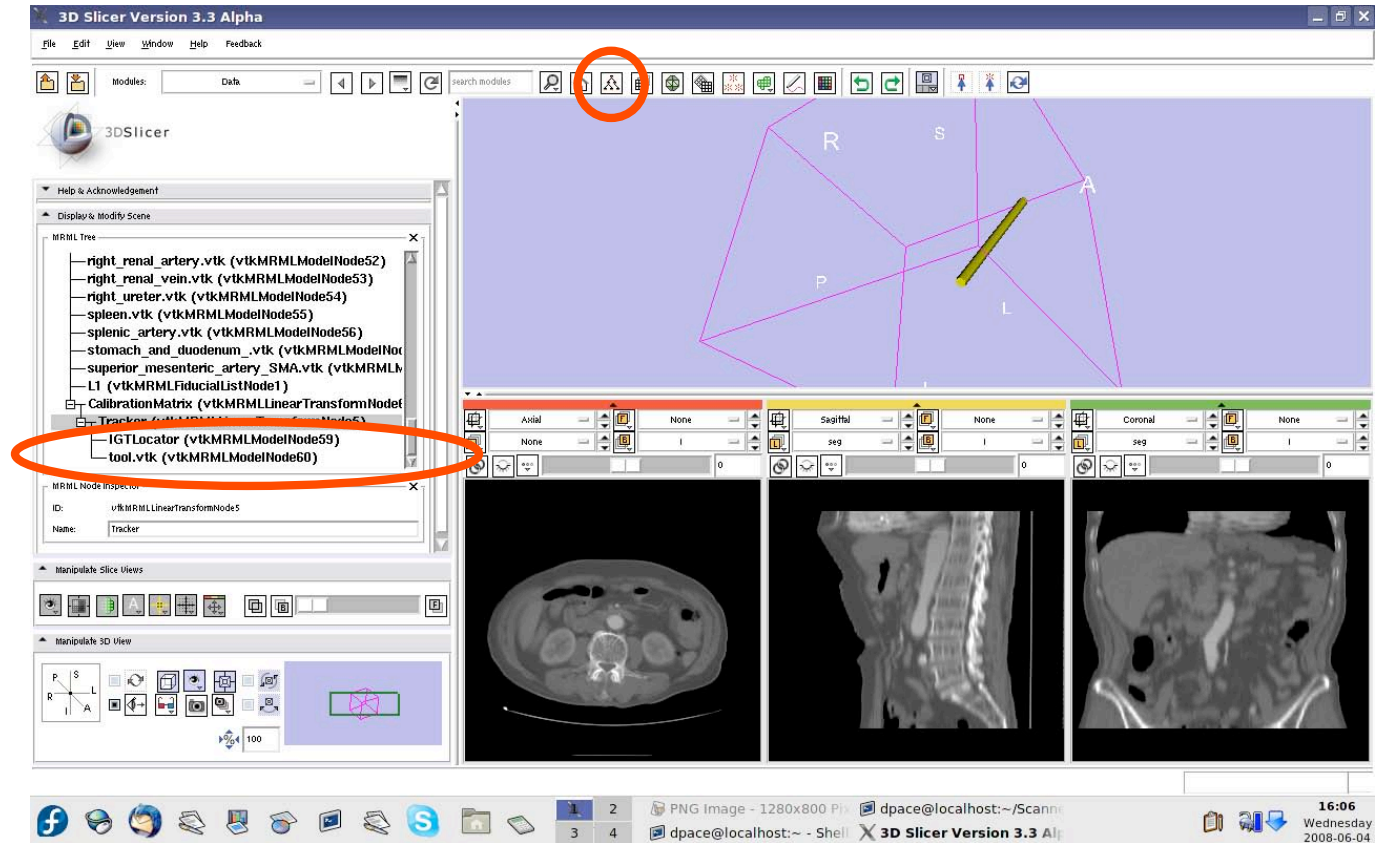


# Add a calibration matrix

Open the Data module

Drag the Tracker node under the Calibration Matrix node

The model will be translated along its length axis

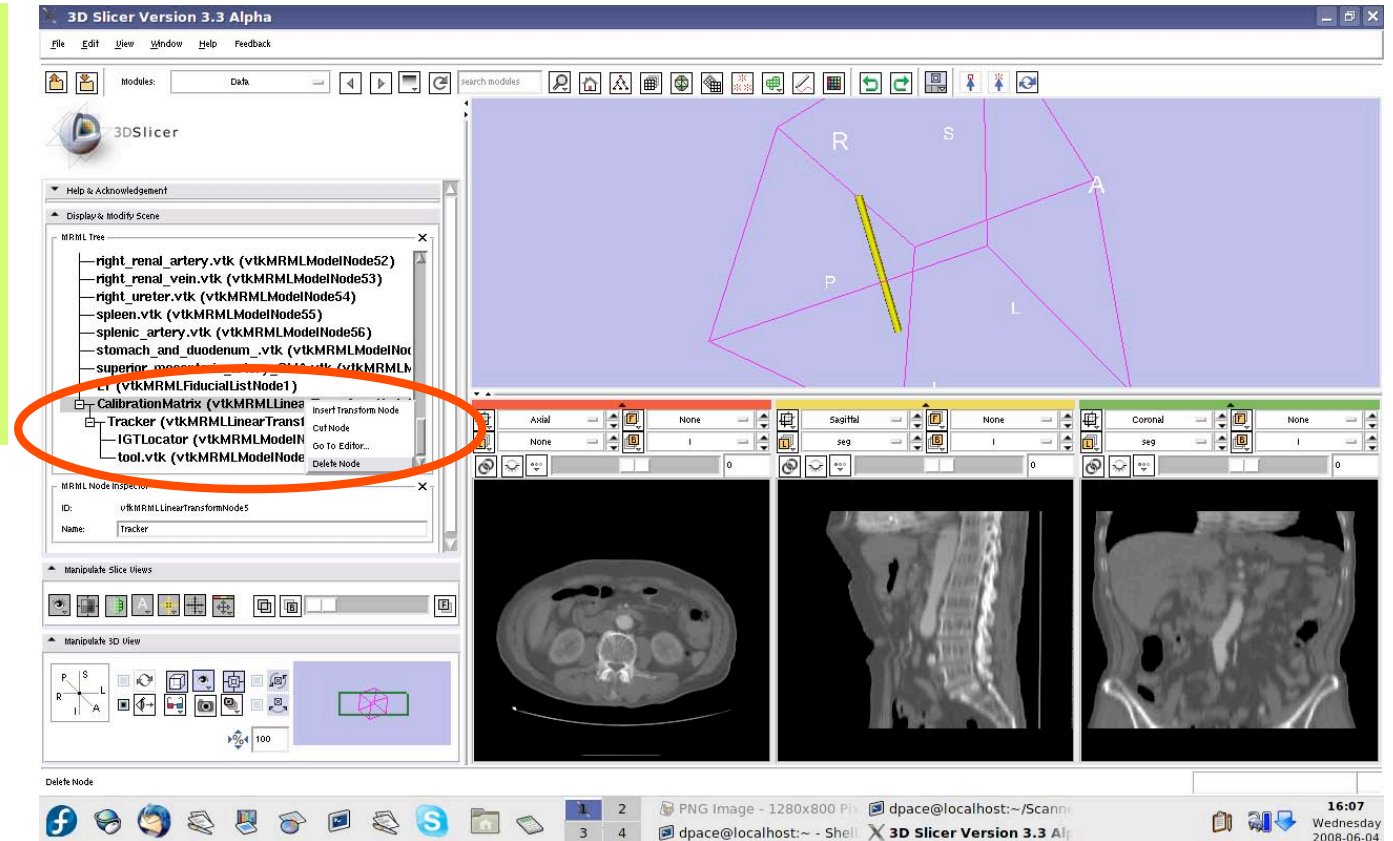






# Reslice the images using the tracker transform

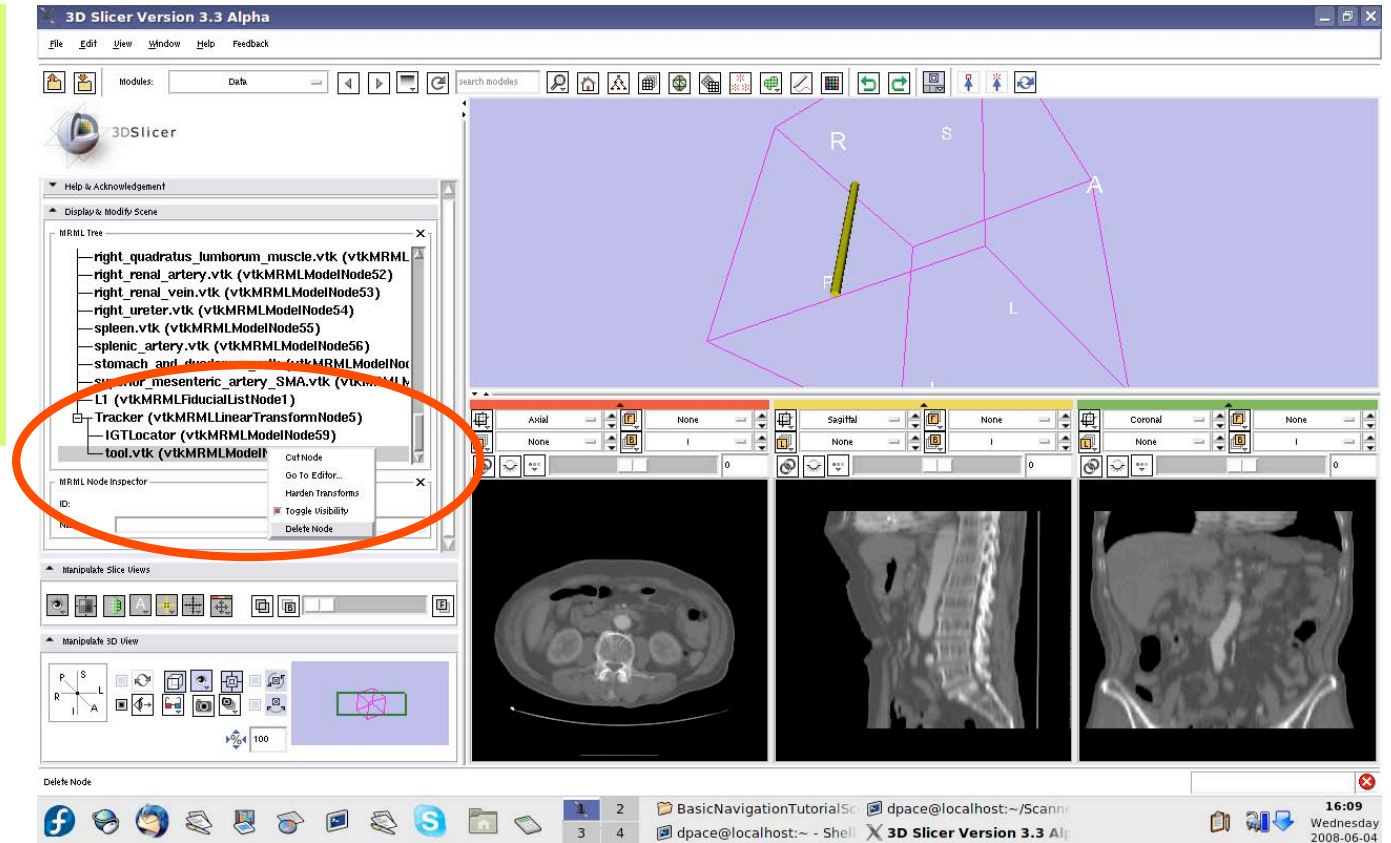
Delete the Calibration Matrix by right-clicking and selecting "Delete Node"





# Reslice the images using the tracker transform

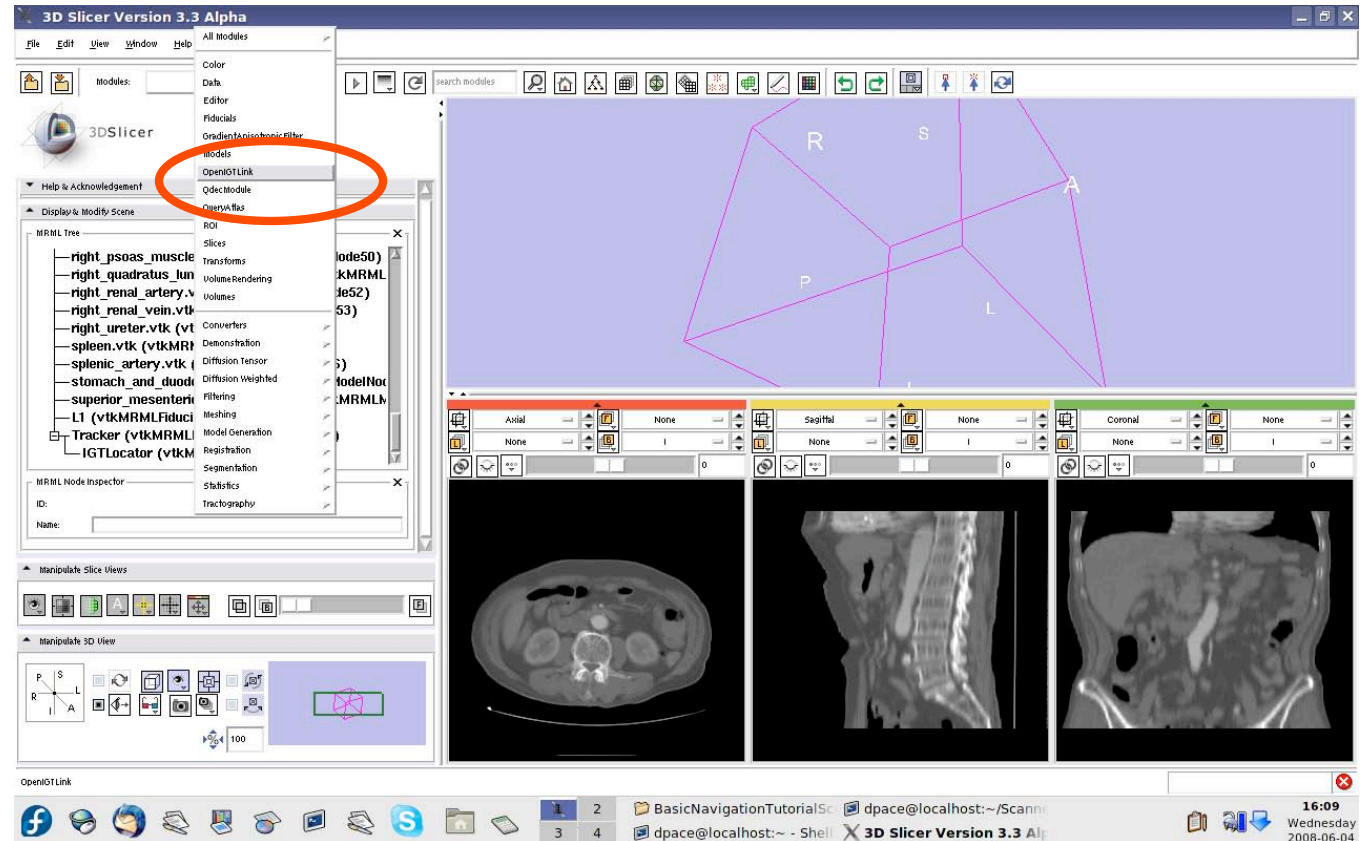
Delete the tool model by right-clicking on tool.vtk and selecting "Delete Node"





# Reslice the images using the tracker transform

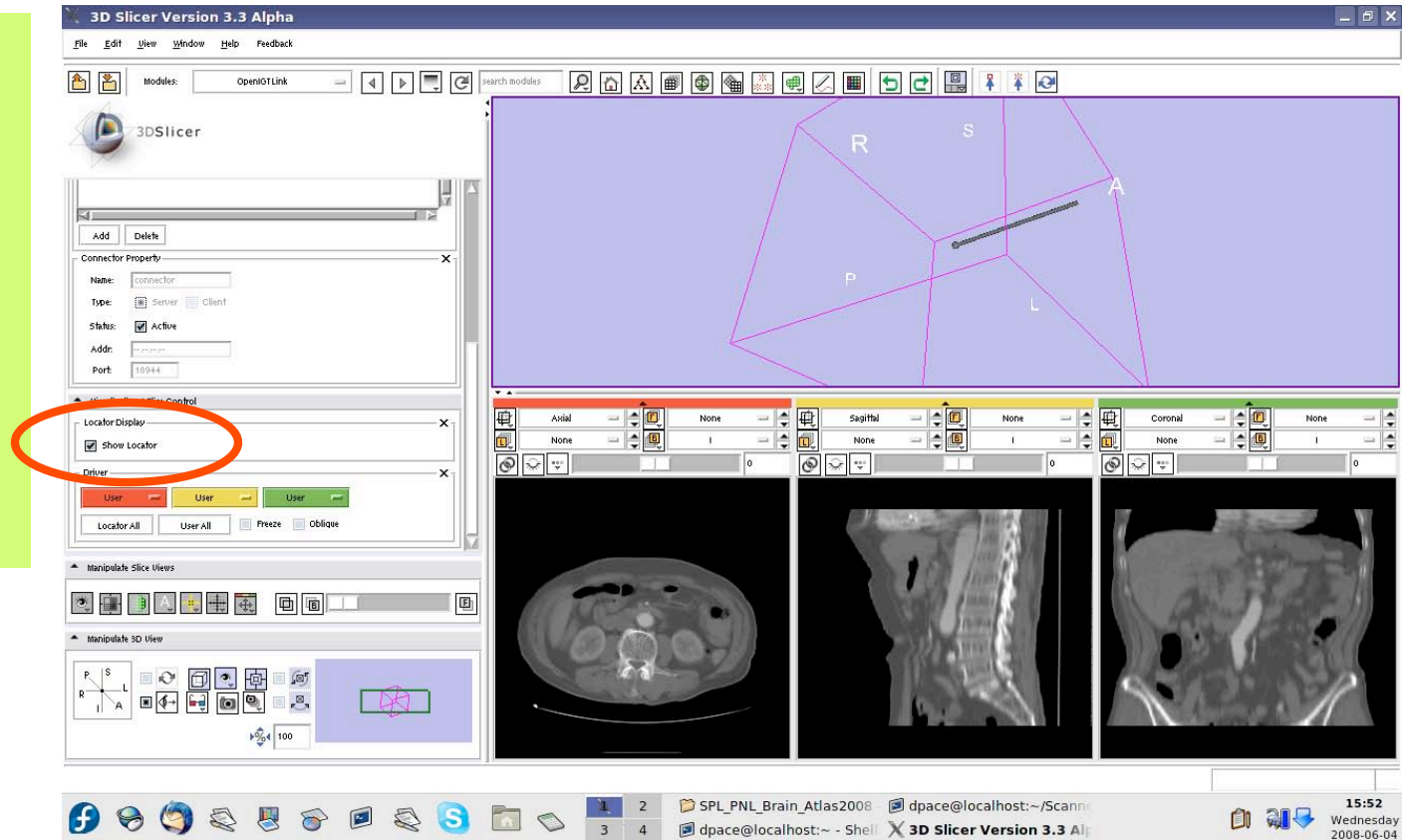
Open the OpenIGTLink module





# Reslice the images using the tracker transform

In the Visualization/  
Slice Control  
pane, click  
the “Show  
Locator”  
button

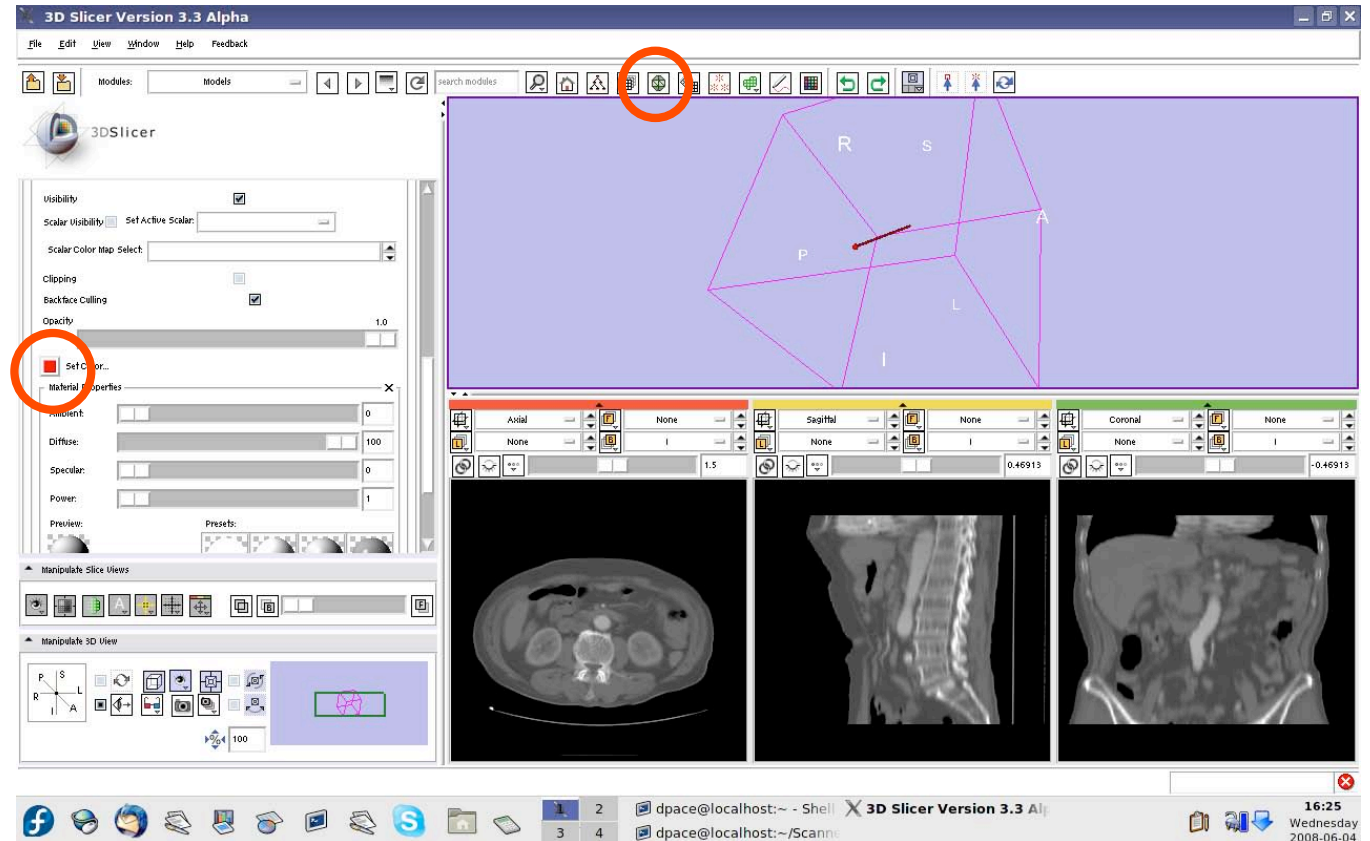




# Reslice the images using the tracker transform

Open the Models module

Select the IGTLocator model as the selected model and change its colour to red

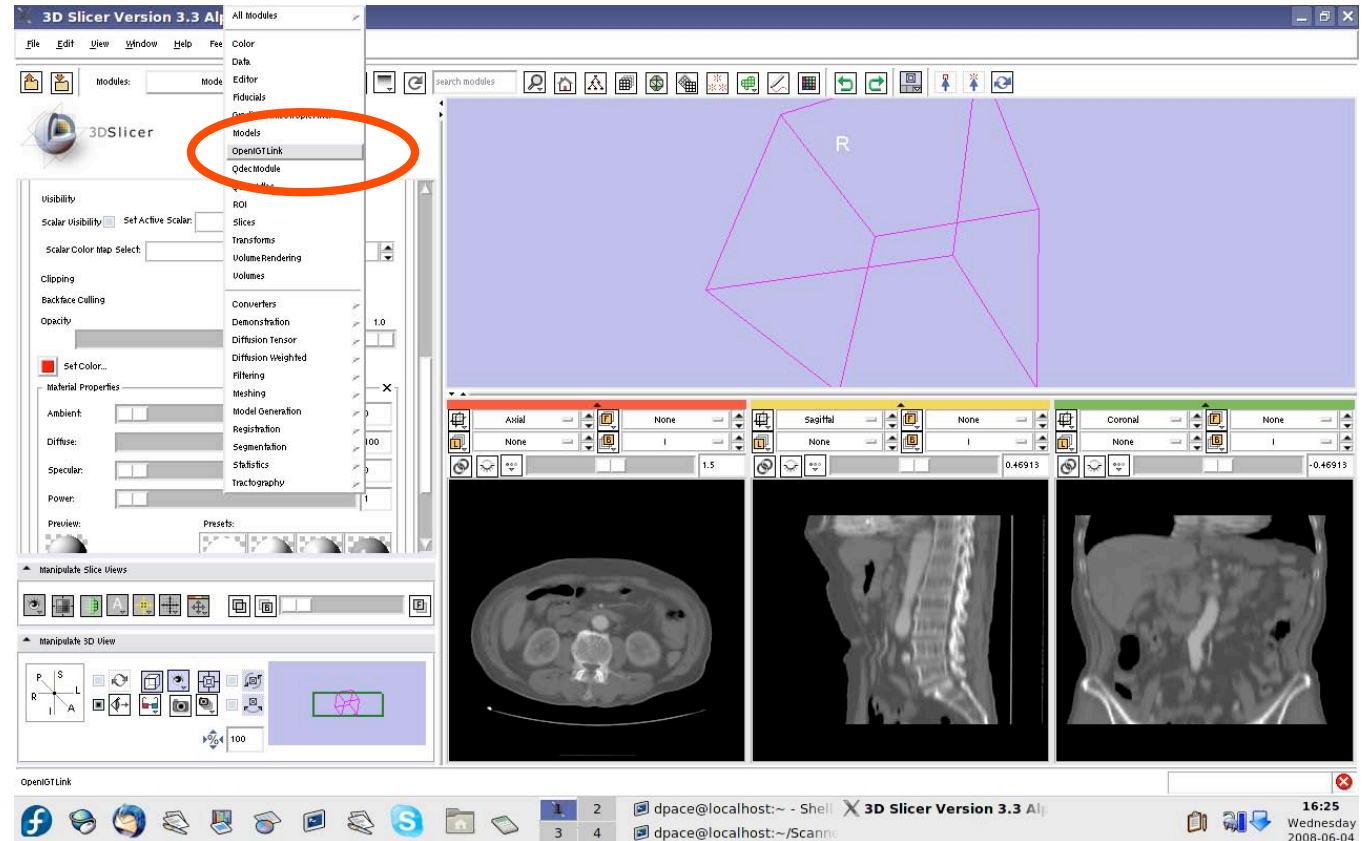






# Reslice the images using the tracker transform

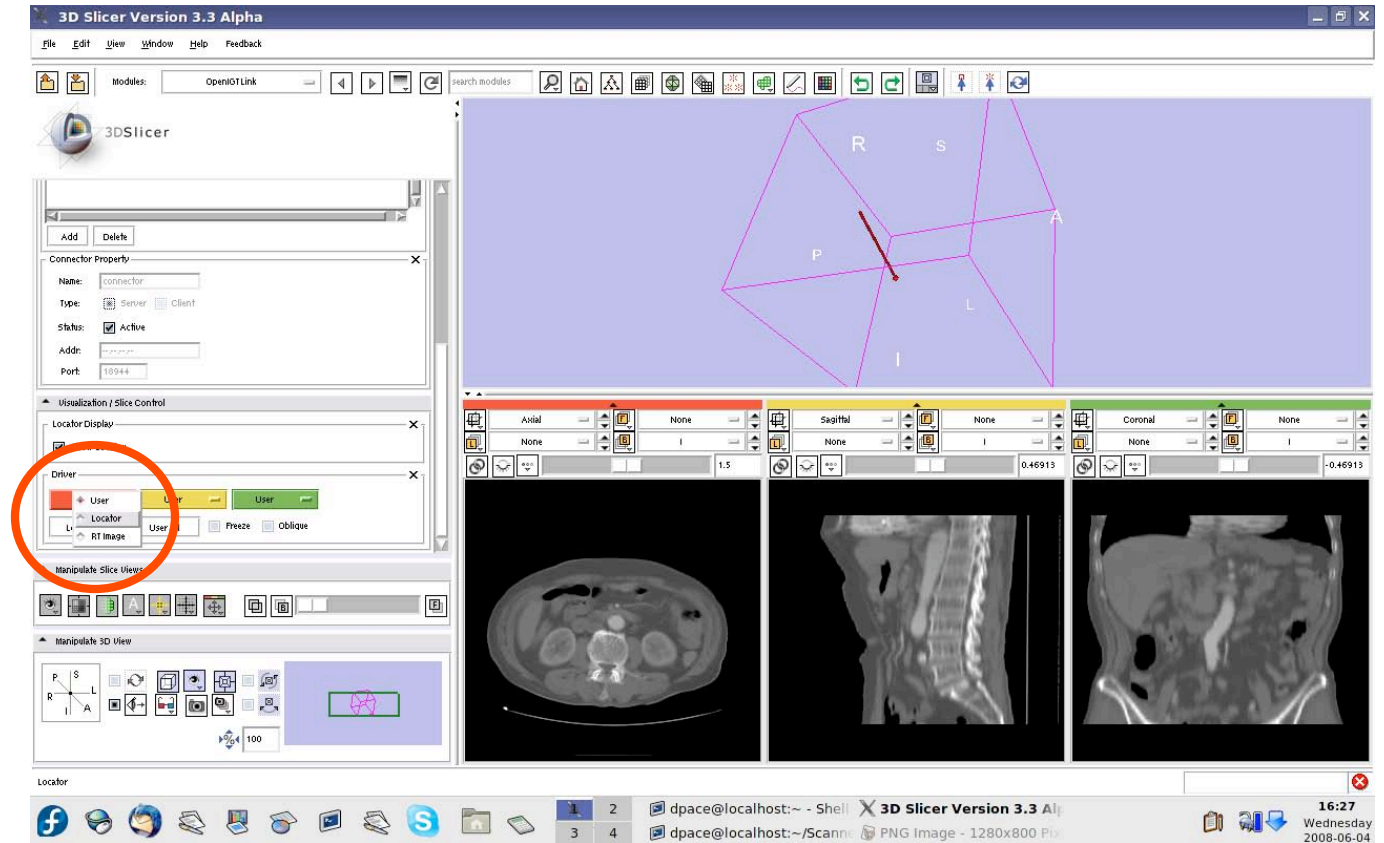
Open the  
OpenIGTLink  
module





# Reslice the images using the tracker transform

Set the driver for the red (axial) slice to "Locator"

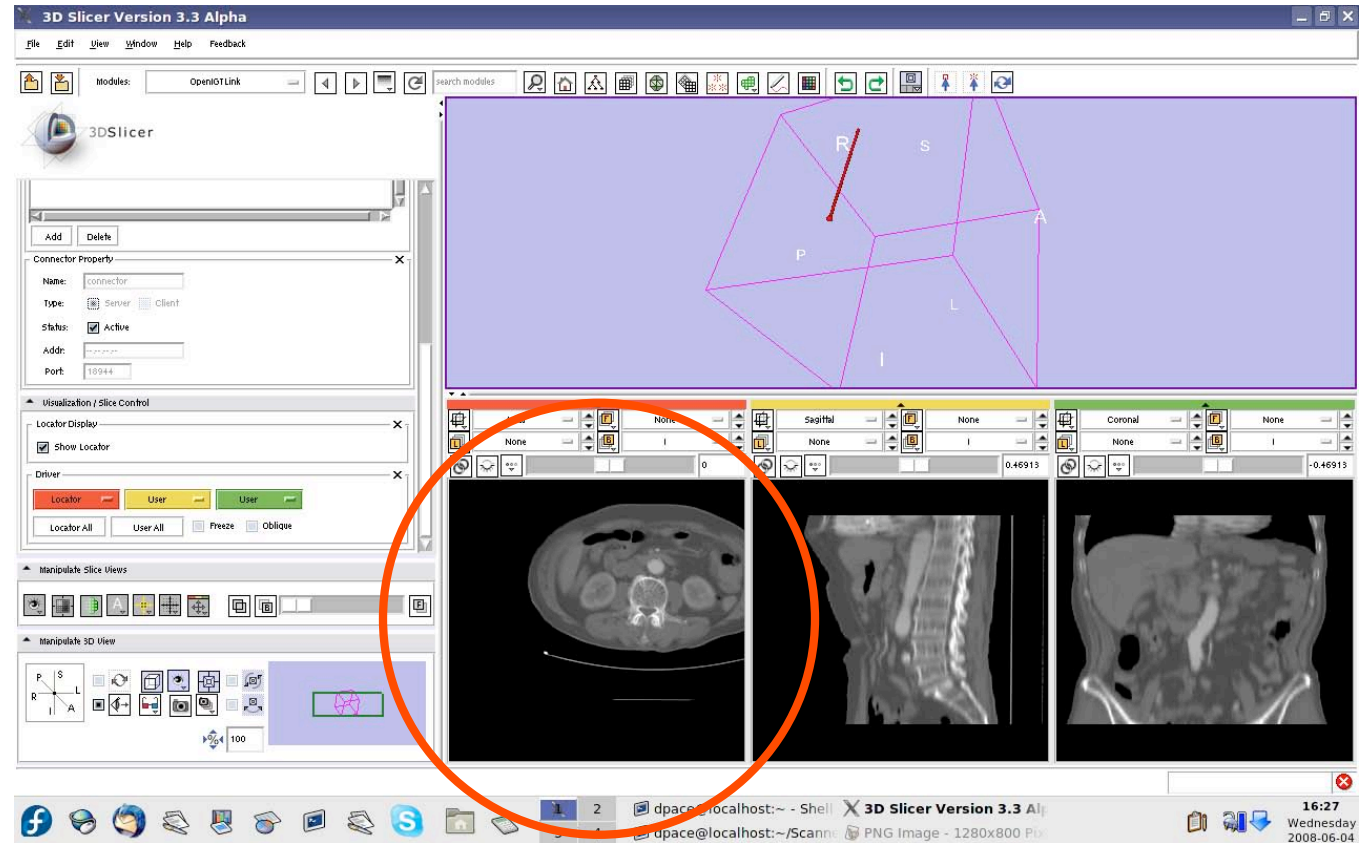






# Reslice the images using the tracker transform

The axial slice moves as the locator moves

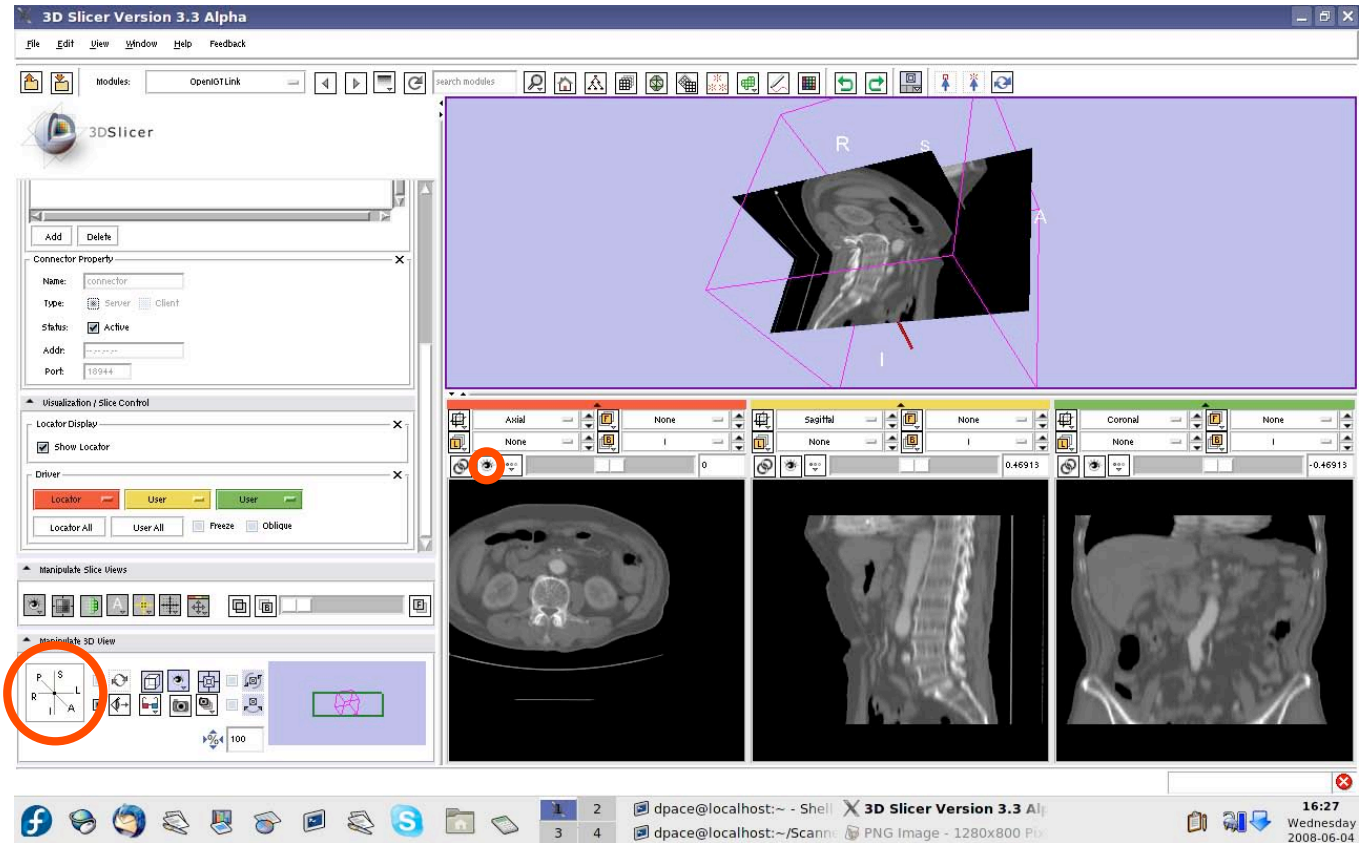




# Reslice the images using the tracker transform

Click on the “visibility” button

Change the view in the 3D viewer by clicking on the “I” (inferior) button on the “Manipulate 3D View” pane

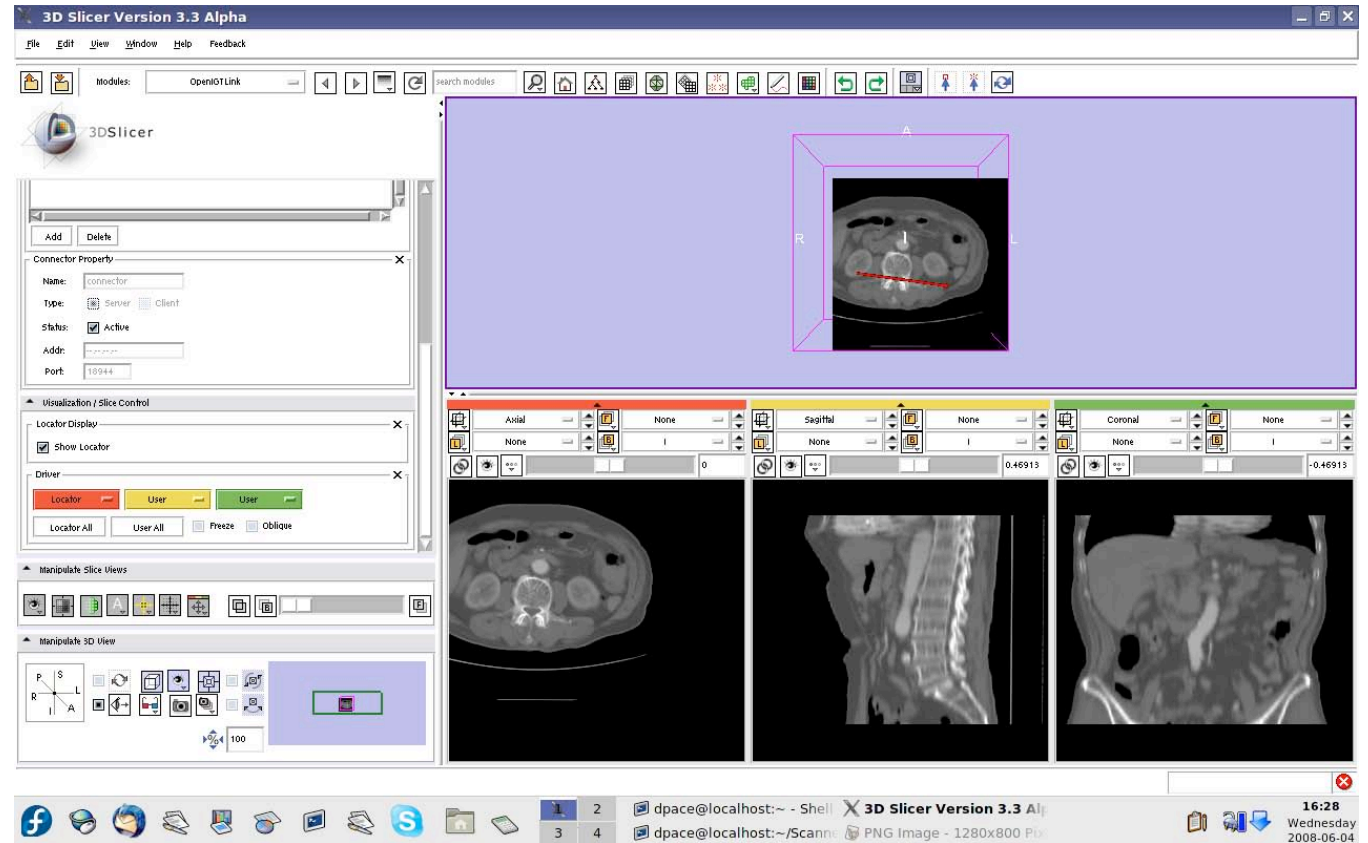




# Reslice the images using the tracker transform

Note that the axial slice moves as the locator moves

This is because the center of the axial slice is set to the locator's position

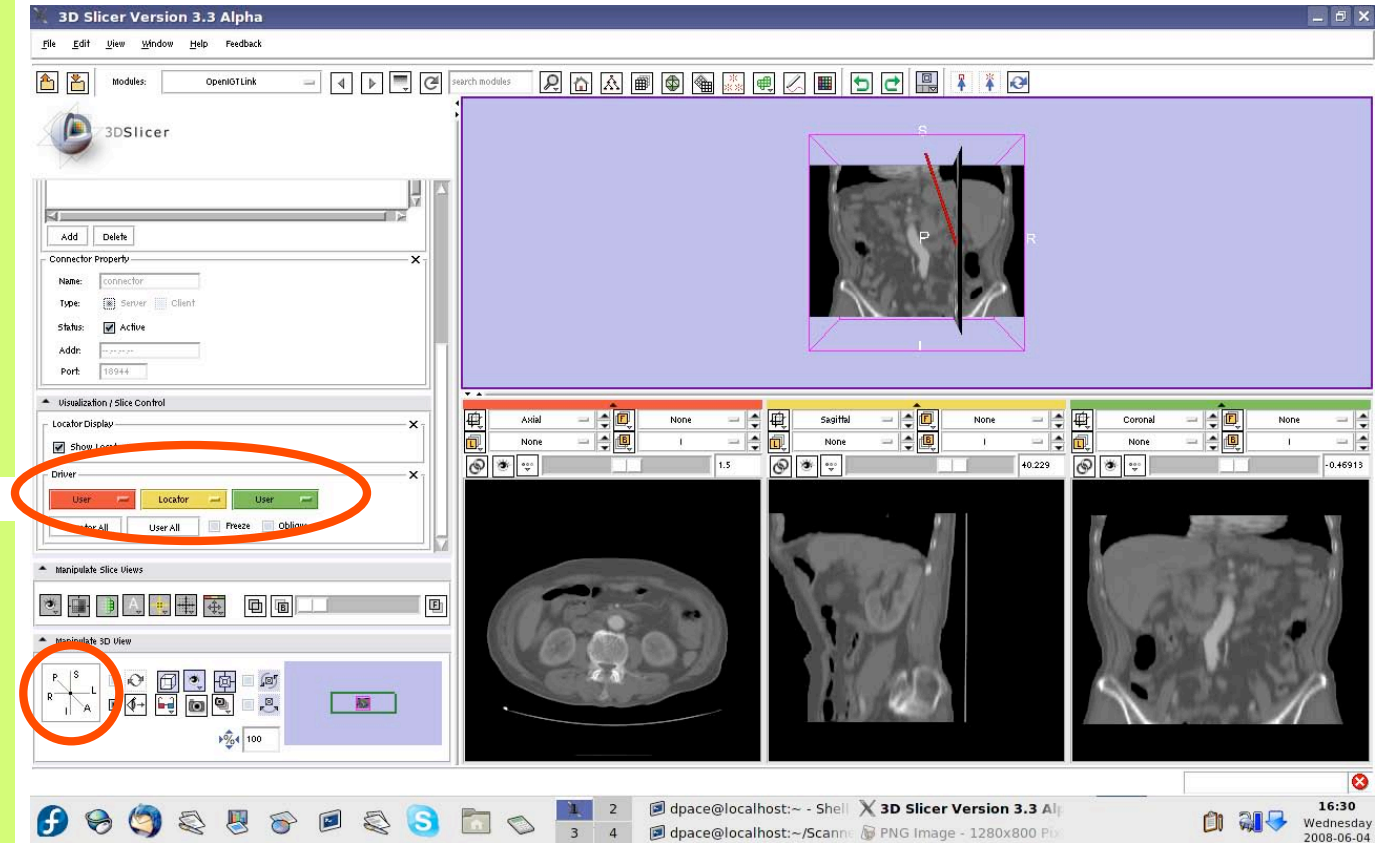




# Reslice the images using the tracker transform

Set the driver for the red (axial) slice to “User” and the driver for the yellow (sagittal slice) to “Locator”

Click on the “P” (posterior) button on the “Manipulate 3D View” pane

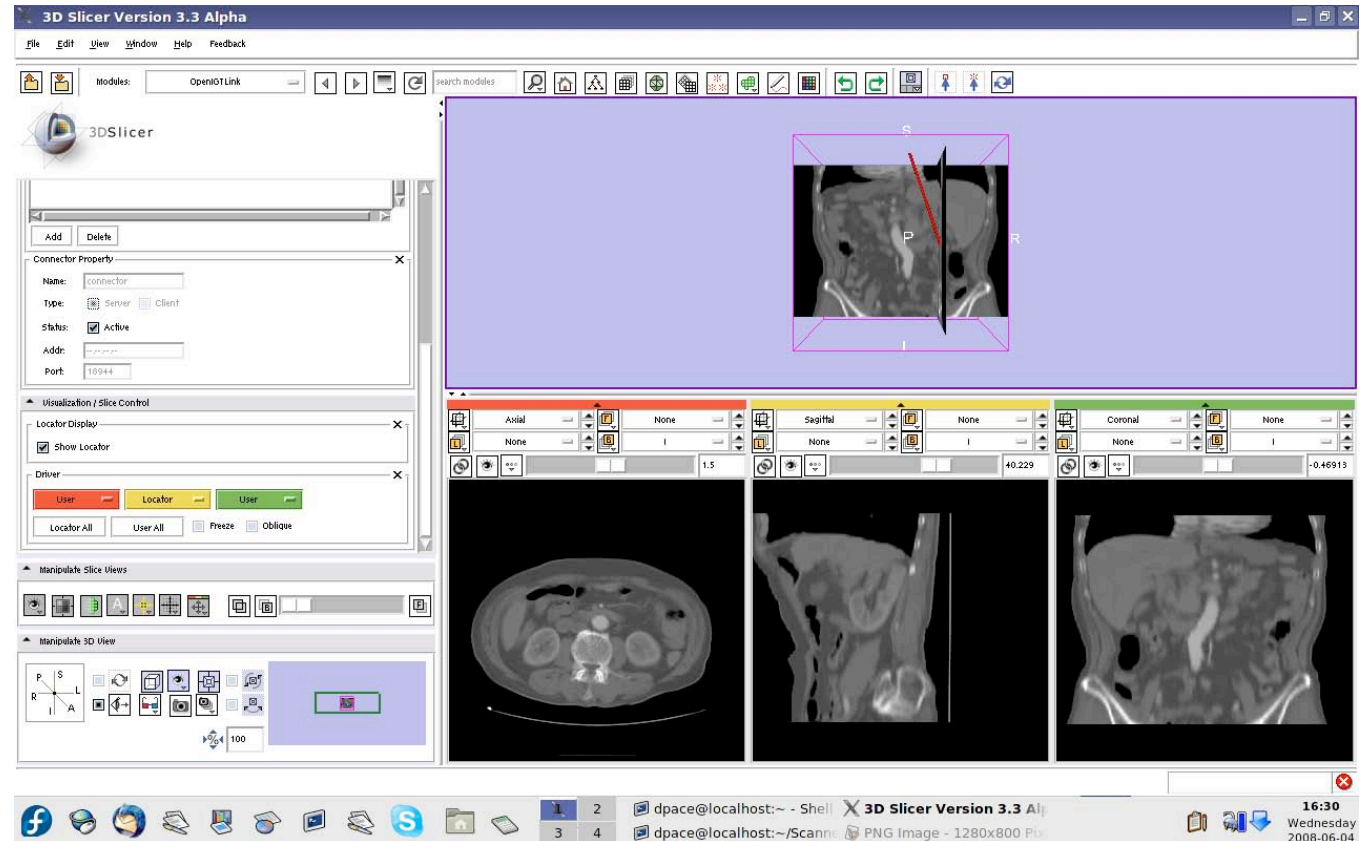




# Reslice the images using the tracker transform

Note that the sagittal slice moves from left to right as the locator moves

(The axial slice didn't move up and down because the locator does not move in the superior-inferior direction)



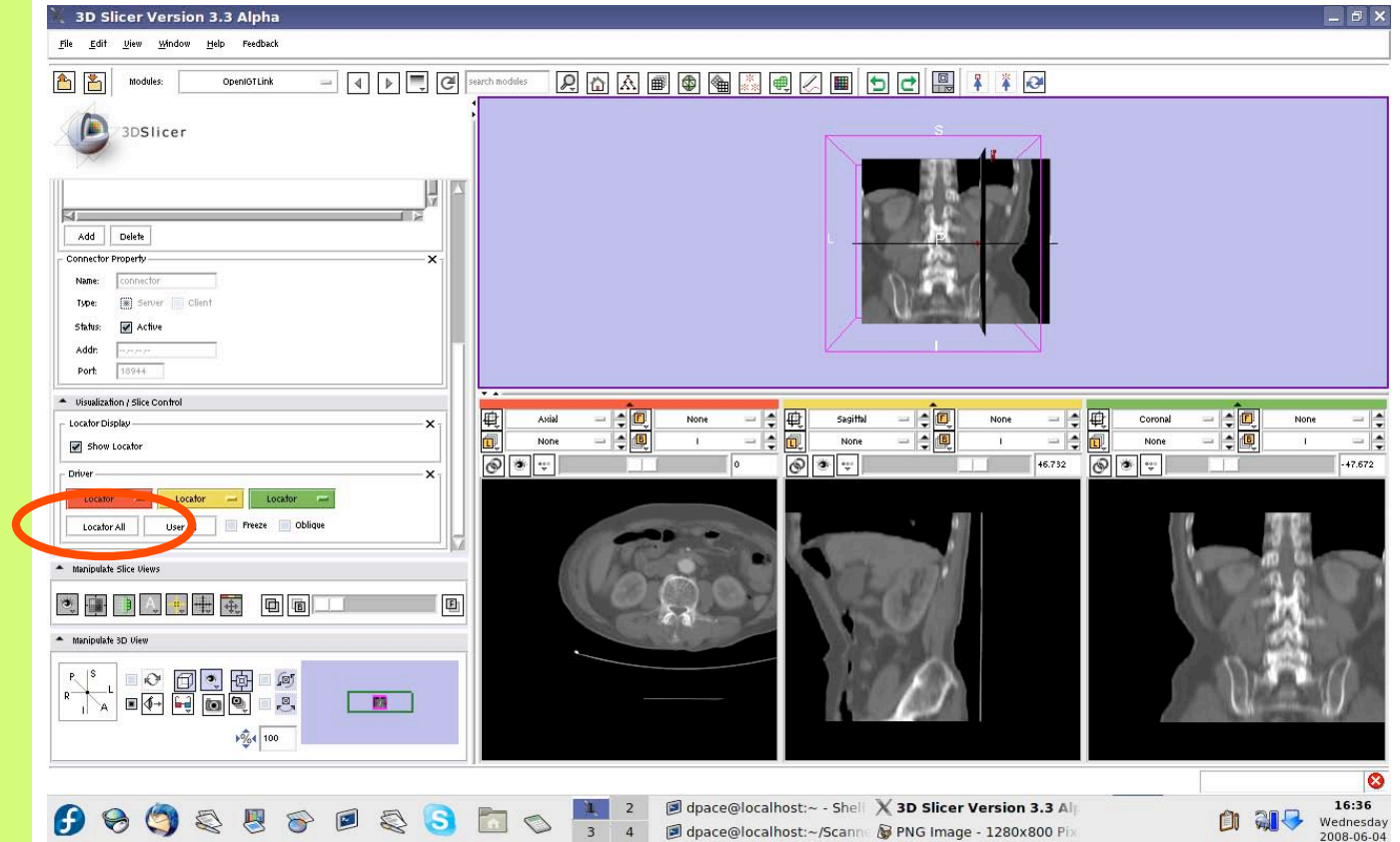




# Reslice the images using the tracker transform

You can click on the “Locator All” button to set the driver to “Locator” for all of the slice views.

The image origin is set to the locator’s position.

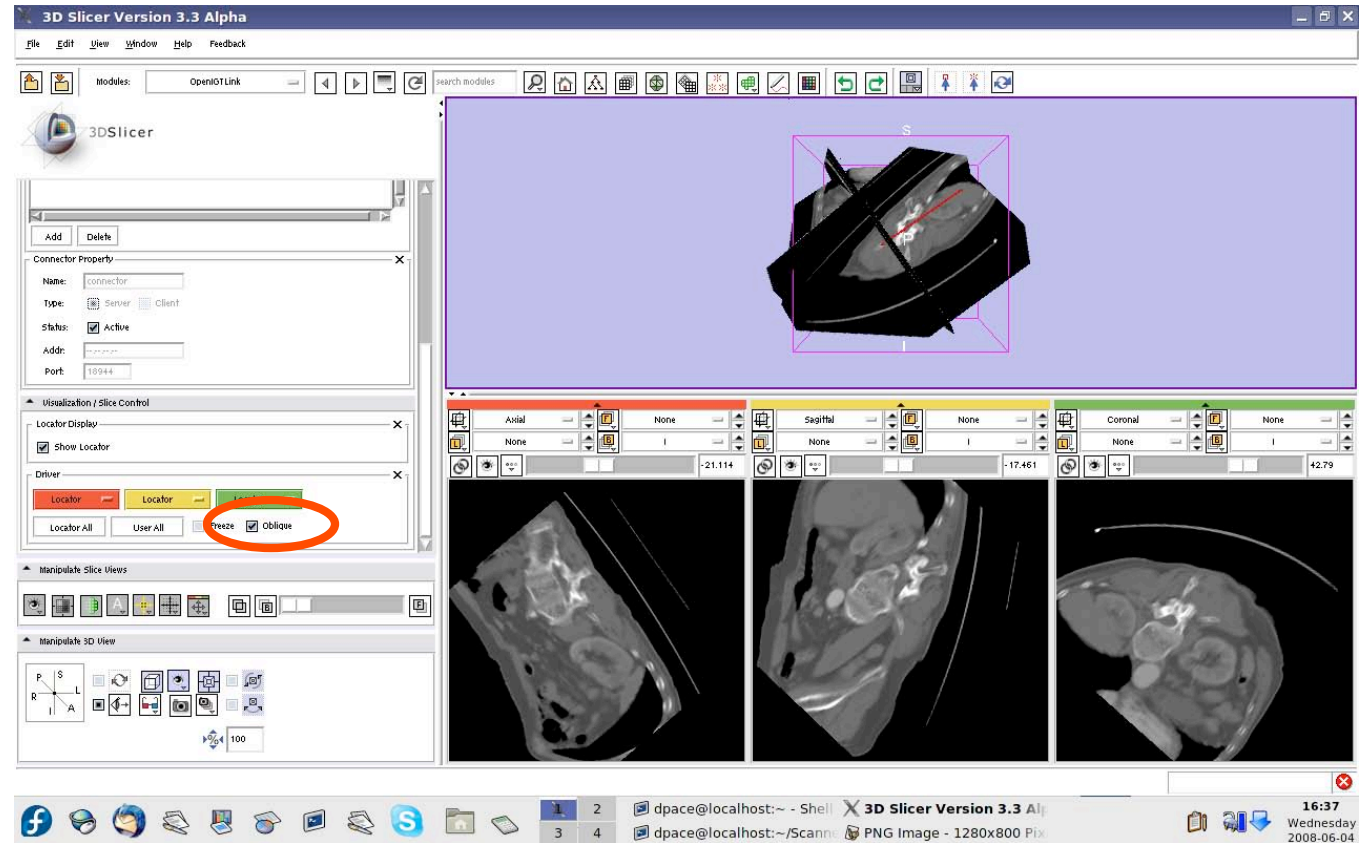






# Reslice the images using the tracker transform

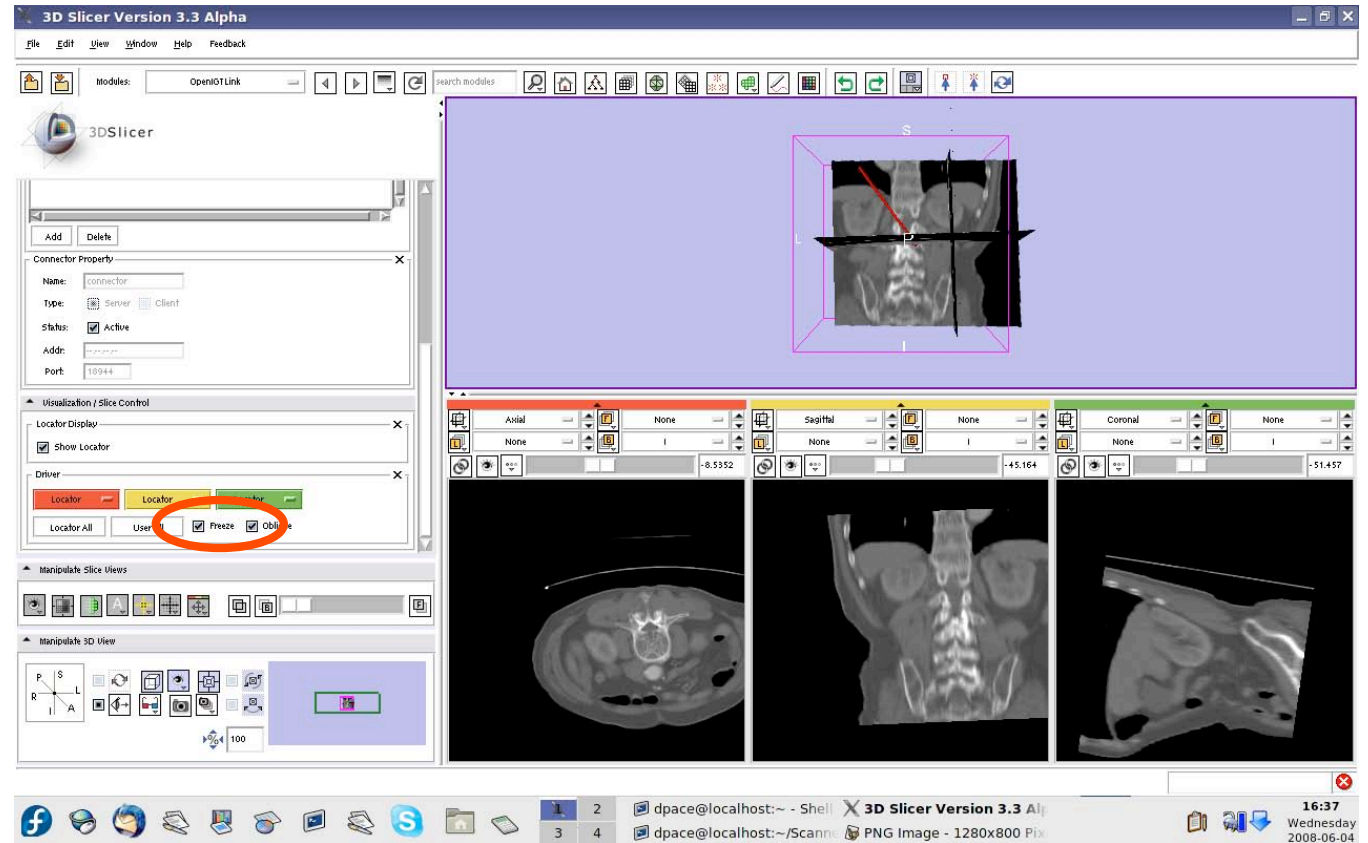
Check the “oblique” box to slice the image volume according to the tool’s orientation - the coordinate system is setup so that one axis is parallel to the locator’s orientation





# Reslice the images using the tracker transform

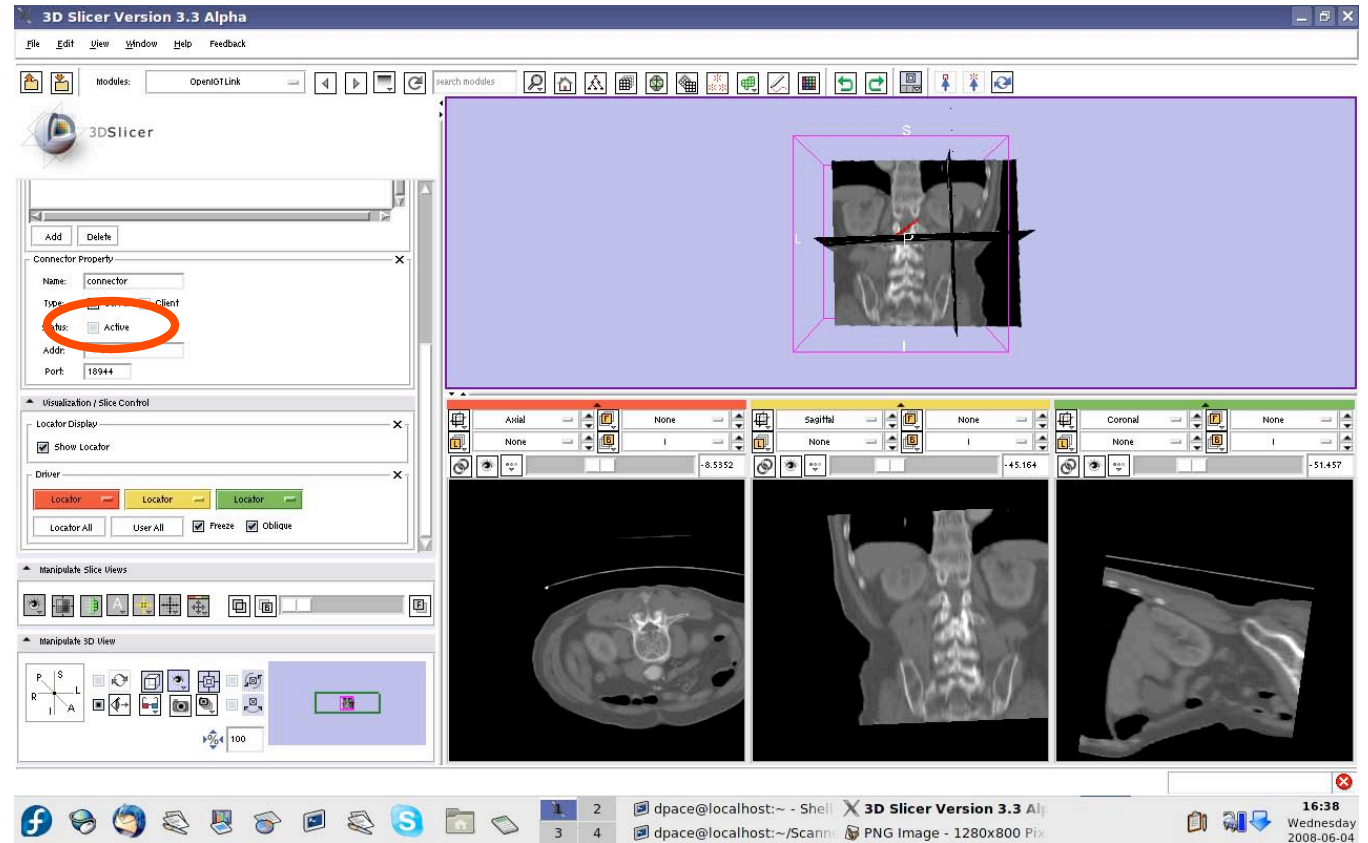
Check the “Freeze” box to freeze the images in both the 3D Viewer and the three slices viewers (the locator keeps moving)





# Turn off the OpenIGTLink connection

Click on the “Active” box to disconnect the OpenIGTLink connection





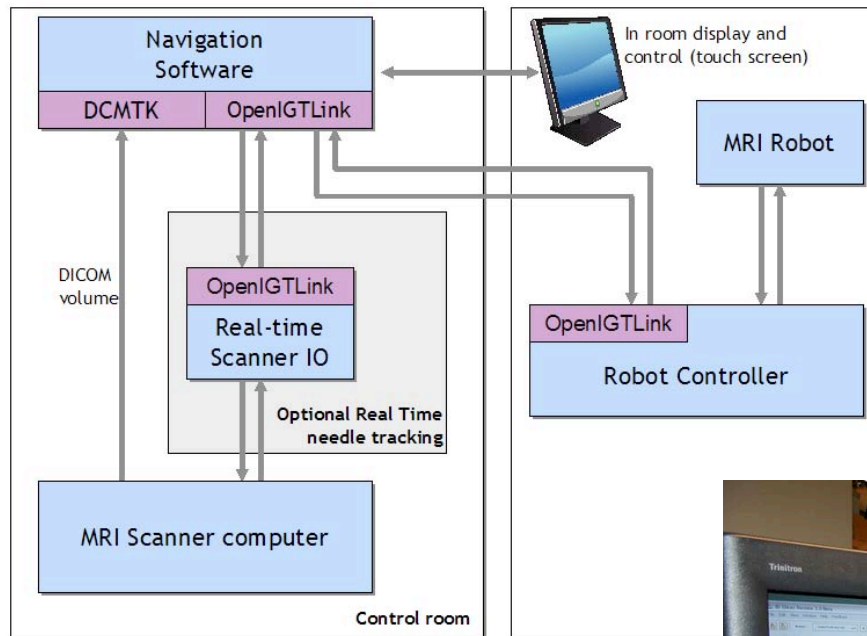
# Tutorial outline

---

1. Introduction to surgical navigation
2. Interfacing Slicer3 with external devices using OpenIGTLink
3. Hands-on navigation using a tracking simulator
4. **Examples of OpenIGTLink in use**



# Examples of OpenIGTLink in use



Prostate biopsy robot under MRI-guidance

Volume-rendered 4D ultrasound





# Overview

---

- In this tutorial, you learned:
  - How OpenIGTLink can be used to perform navigation in Slicer3
  - How to set up OpenIGTLink connections using the OpenIGTLink module in Slicer3
  - How to visualize the tracker transforms
  - How to reslice image volumes using the tracker transforms
  - How OpenIGTLink is currently being used in practice





# Conclusions

---

- Slicer3 can interact with common devices used in Image Guided Therapy
- OpenIGTLink is evolving technology - expect lots of active development!
- Slicer3 is free open-source software that allows IGT researchers to share algorithms and work within a common framework



# For more information...

---

- The Slicer3 IGT Advanced Navigation Tutorial uses the Aurora magnetic tracking devices from NDI, and provides a more thorough explanation of the OpenIGTLink protocol:

<http://wiki.na-mic.org/Wiki/index.php/IGT:ToolKit/Navigation-with-Aurora>

- For a description of the OpenIGTLink protocol:  
<http://www.na-mic.org/Wiki/index.php/OpenIGTLink>