SSMR/ISMR '19 Workshop

Building Software Systems for Image-Guided Robot-Assisted Interventions

Junichi Tokuda[1], Axel Krieger[2], Tamas Ungi[4], Simon Leonard[3],

[1] Brigham and Women's Hospital and Harvard Medical School, Boston MA
[2] University of Maryland, Collage Park, MD
[3] Queen's University, Kingston, ON
[4] Johns Hopkins University, Baltimore, MD

Acknowledgements

- This work is supported in part by:
  - U.S. National Institutes of Health (R01EB020667, R01EB020610, P41EB015898).
  - Ontario Consortium for Adaptive Interventions in Radiation Oncology (OCAIRO), SparKit project
  - CANARIE's Research Software Program

- Contributors:
  - Tobias Frank, M.Sc. Leibniz Universität Hannover
  - Niravkumar Patel, Ph.D., Johns Hopkins University

Background

- Growing Interest in Robot Assisted Interventions
  - Robot-assisted laparoscopy
  - Robotic catheter systems
  - Robotic radiosurgery, etc.
- R&D of Surgical Robot System
  - Image processing and visualization for surgical planning
  - Kinematics and motion planning for robot control
  - Device management and control

➔ Requires a wide range of tools and methods developed in robotics and medical image computing fields

Background (2)

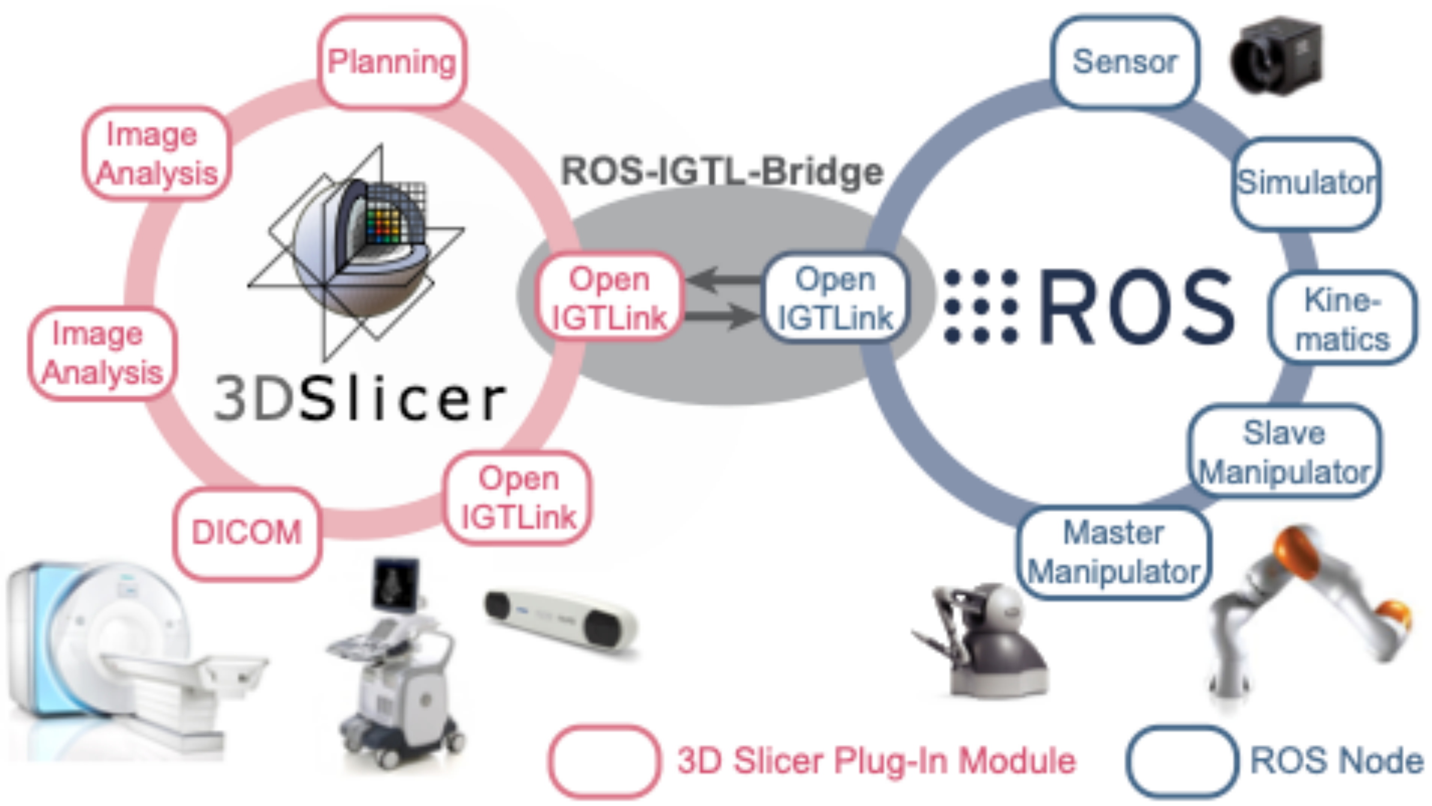
- Common research platforms
  - Medical Image Computing
    - 3D Slicer
    - MITK
    - NifTK
    - OsiriX…
  - Medical Robotics
    - da Vinci Toolkit (dVRK)
    - Raven II
    - KUKA Lightweight Robot

Data sharing interface based on OpenIGTLink

Integrated with Robot Operating System (ROS)

➔ Need for a bridge between ROS and OpenIGTLink

Architecture



BRIGHAM AND WOMEN'S HOSPITAL

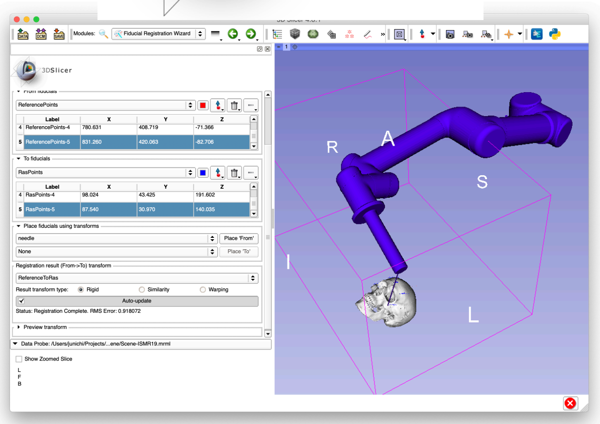
Goals of This Tutorial

- Prototype surgical robot system using open-source software
  - 3D Slicer as a planning interface
  - Robot Operating System (ROS) as robot control software
  - Robot arm (UR-5)

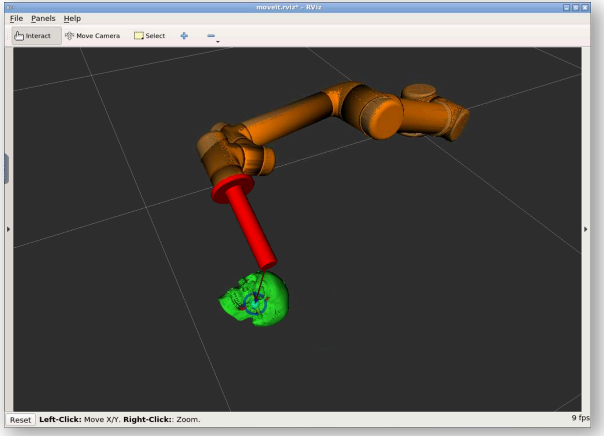
- Through this tutorial, you will:
  - Learn software architecture and clinical workflow for surgical robot systems
  - Acquire hands-on experience of software-hardware integration for medical robotics

System Diagram for Tutorial

3DSlicer



ROS



Target/entry points

Positions of links

3D medical image (MRI)

- Surgical planning
- Procedure monitoring

Robot Arm

- Kinematics
- Path planning

Challenges…

- Exchanging data between two different systems

- Two coordinate systems
  - Defined by the base link on ROS
  - Defined by the patient on 3D Slicer

Prerequisite

- Computer with one of the following operating systems:
  - Windows 7 or higher (Windows 10 recommended)
  - Mac OS X (macOS) 10.7 or higher (macOS Sierra or higher is recommended)
  - Linux

- Software
  - 3D Slicer Version 4.8.1* (http://www.slicer.org/)
  - Docker (https://wwww.docker.com/)
  - Git Client (https://git-scm.com/)

- See http://rosmed.github.io/ismr2019/prerequisite for detail.

* Some of the features used in this tutorial have not been supported in Slicer 4.10.x

Step 1: Setting up Environment

Overview of Step 1

- Set up ROS – two options:
  - [Option 1] Virtual environment (Docker)
  - [Option 2] Native Linux machine

- Set up 3D Slicer
  - SlicerIGT extension (plug-in)

- Test communication between ROS and 3D Slicer

Options for ROS Environment

- [Option 1] Virtual environment (recommended for tutorial)
  - Runs on the same host where 3D Slicer is installed
  - Pre-configured OS image
  - Minimum installation effort / no extra cost
  - Suitable for learning and prototyping software

- [Option 2] Dedicated hardware
  - Runs on a dedicated Linux computer
  - Performance advantage
  - Require Linux installation
  - Cost for the computer hardware
  - Can be extended for actual system

https://www.goodfreephotos.com

[Option 1] Downloading Docker Image

- Pre-configured Docker image contains:
  - Ubuntu 16.04
  - ROS Kinetic
  - OpenIGTLink
  - ROS-IGTL-Bridge
  - HTML5 VNC (for desktop environment)

- To download Docker image, open a terminal on the host and run:

```
$ docker pull rosmed/docker-ros-igtl
```

ubuntu

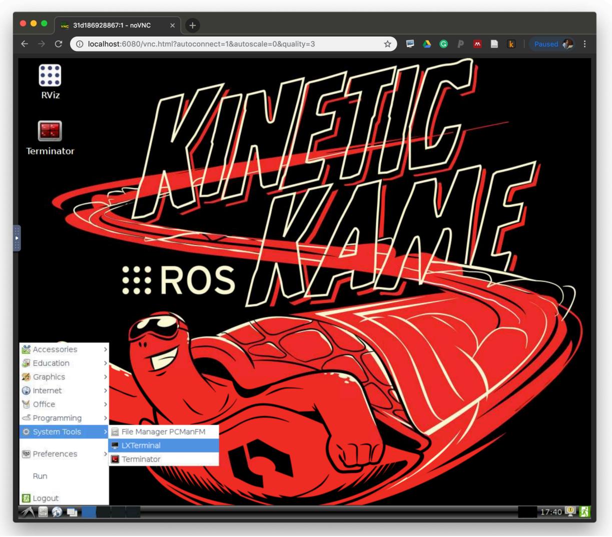
[Option 1] Launching ROS on Docker

1. From the command prompt on the host :

```
$ docker run -it --rm -p 6080:80 -p 28944:18944
rosmed/docker-ros-igtl
```

2. From an HTML5 web browser (e.g. Chrome), open:

```
http://localhost:6080
```



[Option 2] Setting up ROS Environment

- Please follow: http://rosmed.github.io/ismr2019/ros_environment

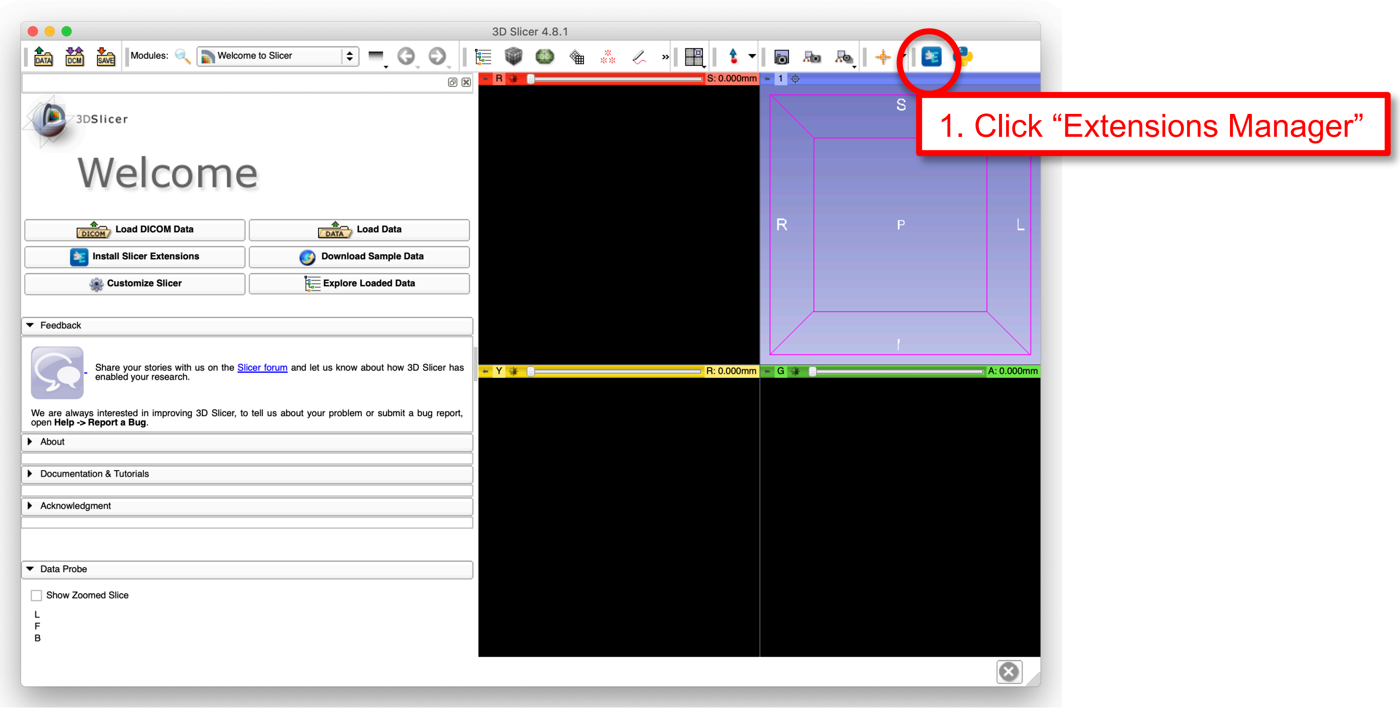
Installing 3D Slicer (1)

- Download 3D Slicer 4.8.1
  - Windows: http://slicer.kitware.com/midas3/download/item/329467/Slicer-4.8.1-win-amd64.exe
  - Mac: http://slicer.kitware.com/midas3/download/item/330418/Slicer-4.8.1-macosx-amd64.dmg
  - Linux: http://slicer.kitware.com/midas3/download/item/330417/Slicer-4.8.1-linux-amd64.tar.gz

- Follow instructions at: https://www.slicer.org/wiki/Documentation/4.8/SlicerApplication/Installation

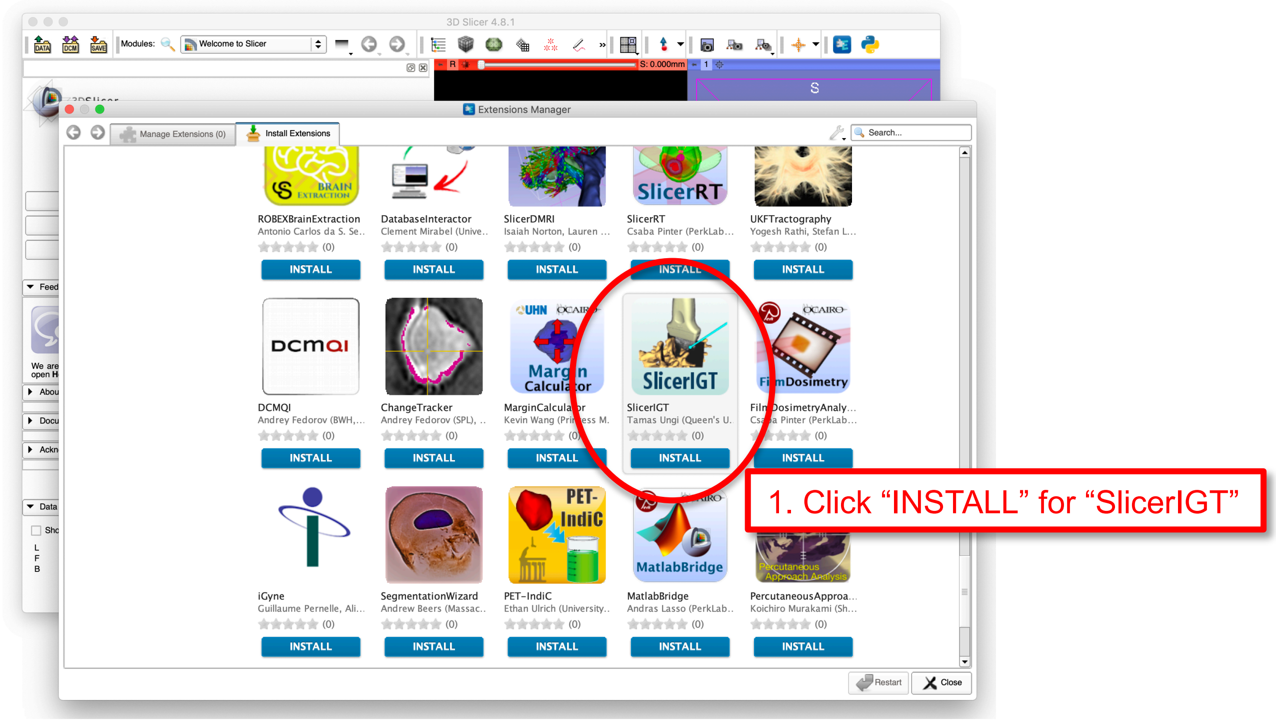
Installing 3D Slicer (2)

- After installing 3D Slicer, open Extensions Manager.



Installing 3D Slicer (3)

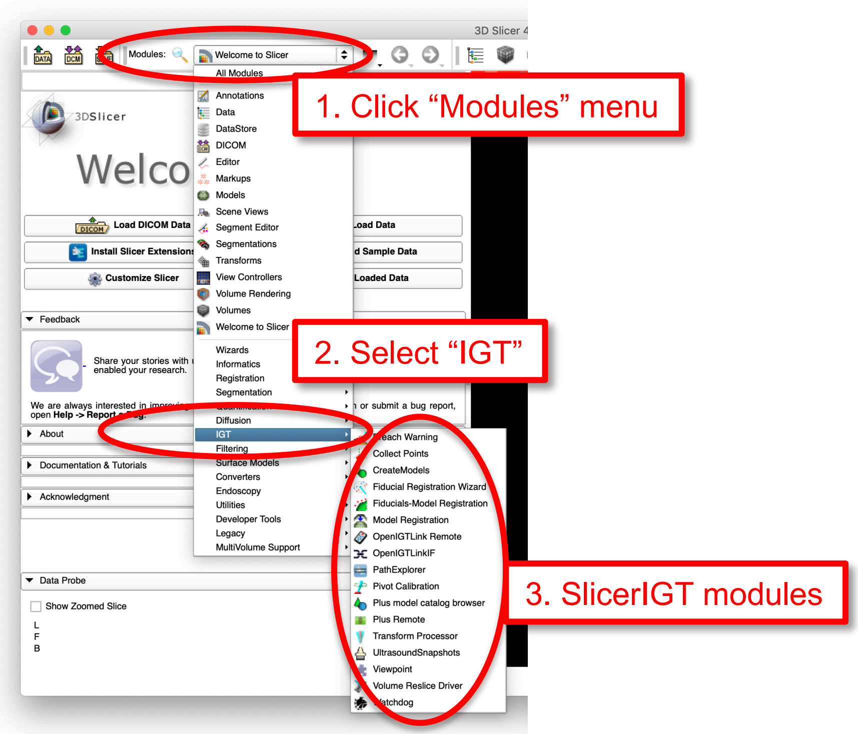
- Install “SlicerIGT” extension and restart 3D Slicer.



BRIGHAM AND WOMEN’S HOSPITAL

Installing 3D Slicer (4)

- Modules for SlicerIGT can be found under “IGT” section of the Modules menu.



Testing Communication (1)

- Open a first terminal on the ROS computer (Menu -> System Tools -> LXTerminal), and start ‘roscore’:

```
$ roscore
```

- Open a second terminal, and launch the bridge node:

```
$ cd ~/catkin_ws
$ source devel/setup.bash
$ roslaunch ros_igtl_bridge bridge.launch
```

Testing Communication (2)

- The terminal asks to enter the type. Choose ‘1’ (Server)

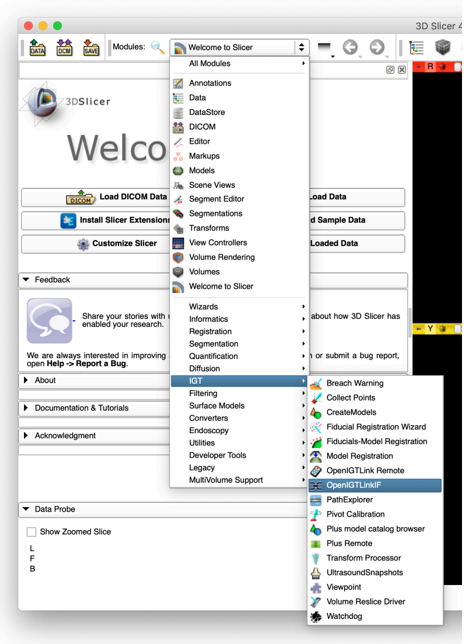
```
[ROS-IGTL-Bridge] Please try <1> or <2> to run node as
OpenIGTLink client or server:
1: Server
2: Client
1
```

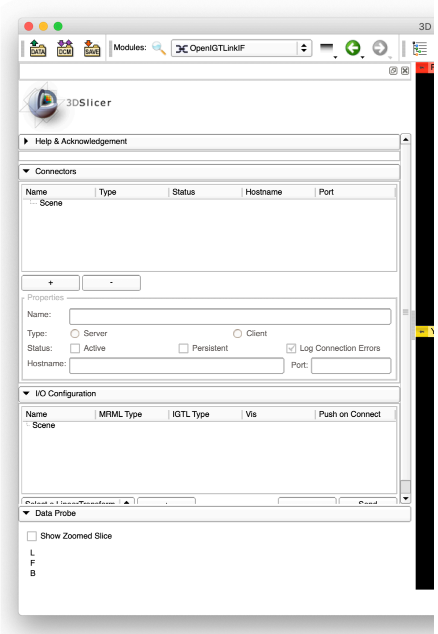
- Then the terminal asks to enter the port number. Enter ‘18944’.

```
[ROS-IGTL-Bridge] Input socket port:
18944
```

Testing Communication (3)

- In 3D Slicer, Open “Modules” -> “IGT” -> “OpenIGTLink IF”



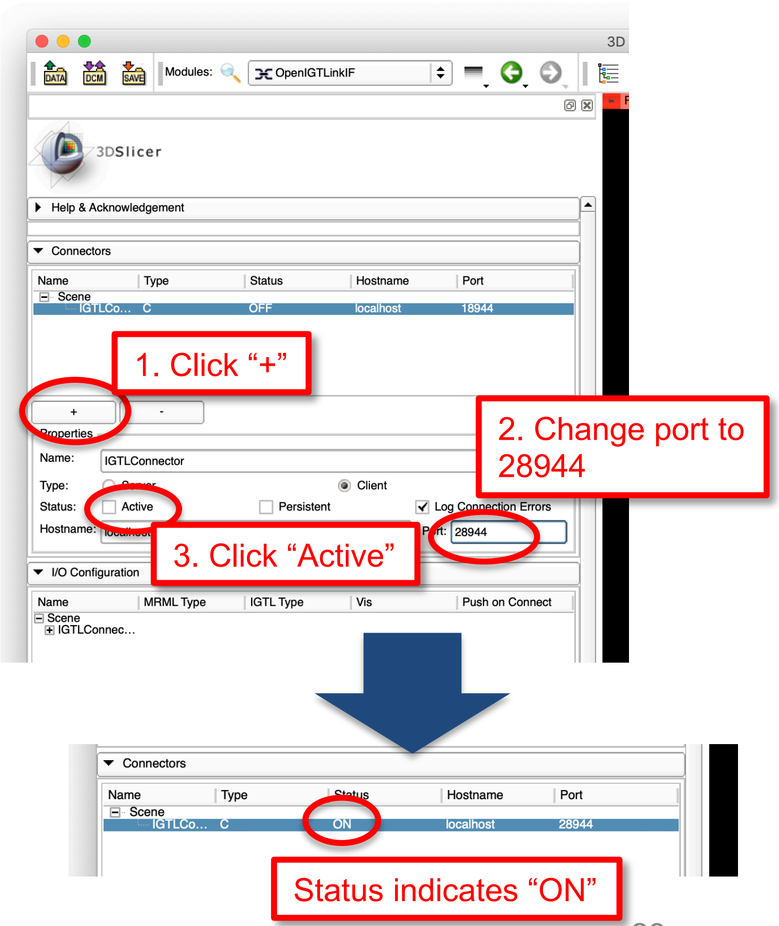


Testing Communication (4)

- Create a connector node and configure:

Name	"IGTLConnector" (default)
Type	"Client" (default)
Status	Unchecked (default)
Hostname	"localhost" (if Docker is used)
Port	"28944"

- After configuring the node, click "Active". If successful, the status becomes "ON".

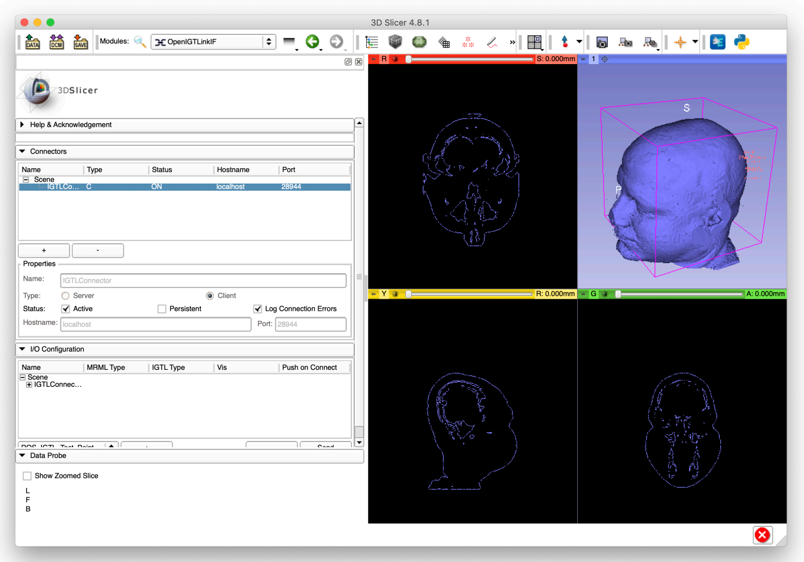


Testing Communication (4)

- Open a third terminal on the ROS computer and run the test node:

```
$ cd ~/catkin_ws
$ source devel/setup.bash
$ roslaunch ros_igtl_bridge test.launch
```

- If successful, image and points will show up on 3D Slicer.



Testing Communication (5)

- After confirming the communication between 3D Slicer and ROS:
  - On 3D Slicer, turn the “Active” checkbox off in the status panel of the ”OpenIGTLink IF” module.
  - On the third terminal, press ”Ctrl + c” to stop the test node.
  - On the second terminal, press “Ctrl + c” to stop the bridge node.
  - On the first terminal, press “Ctrl + c” to stop the roscore.
  - Close all the terminals.

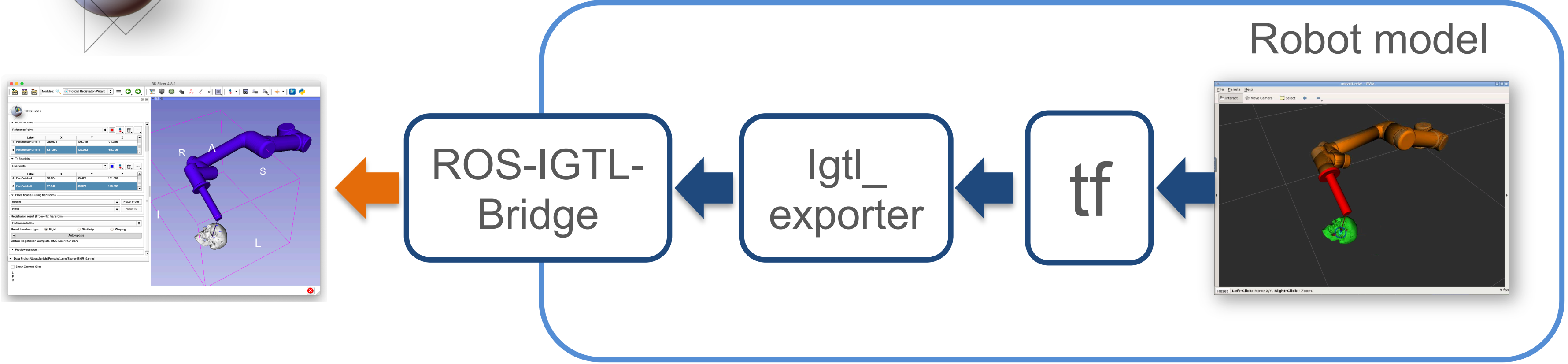
Step 2: Setting up Universal Robot Arm on ROS

Overview of Step 2

- Messaging from ROS to 3D Slicer

3DSlicer

ROS



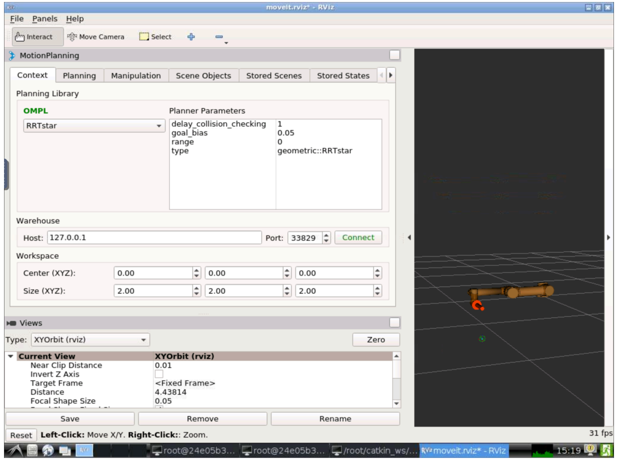
Launching Universal Robot (1)

- On a terminal on the ROS computer:

```
$ cd ~/catkin_ws
$ source devel/setup.bash
$ roslaunch ismr19_moveit demo.launch
```

- rviz software with a 3D model of the Universal Robot appears.
- Next, launch IGTL Exporter/Importer from a new terminal:

```
$ cd catkin_ws
$ source devel/setup.bash
$ rosrun ismr19_control igtl_exporter.py
```



Launching Universal Robot (2)

- Start ROS-IGTL-Bridge from a terminal on ROS computer.

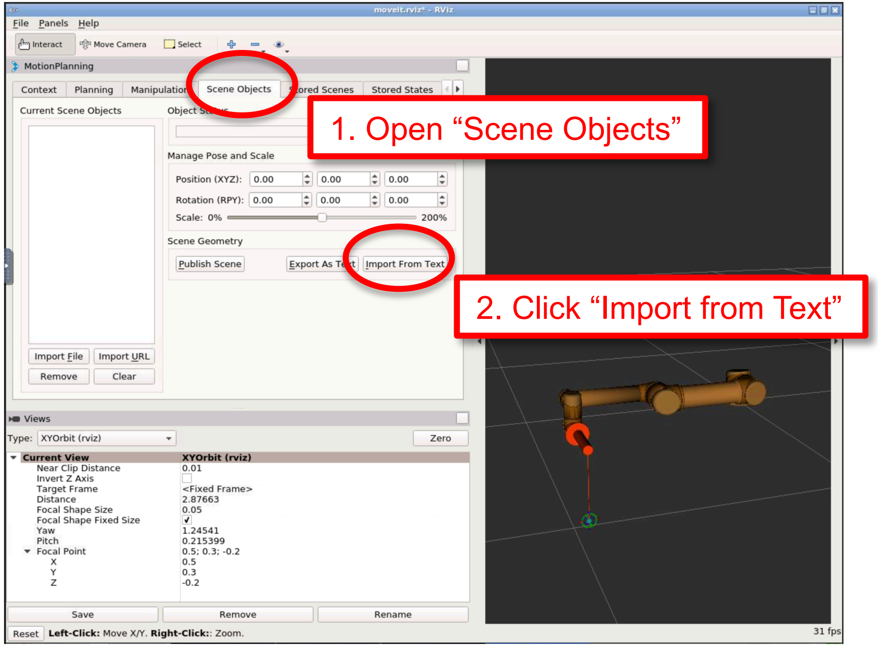
```
$ cd ~/catkin_ws
$ source devel/setup.bash
$ roslaunch ros_igtl_bridge bridge.launch
```

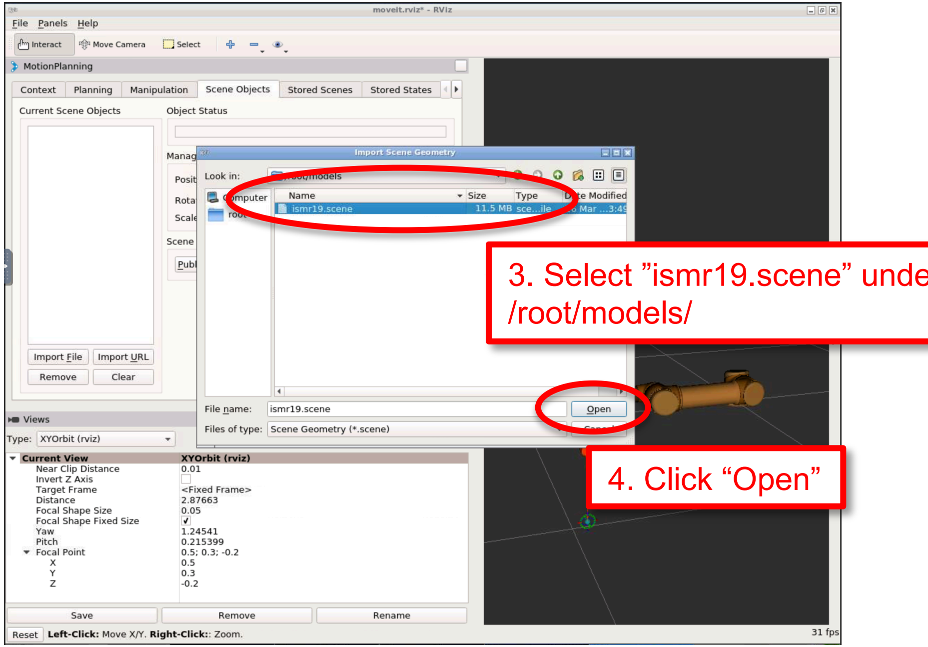
- Choose ‘1’ (Server) and enter port # 18944

```
[ROS-IGTL-Bridge] Please try <1> or <2> to run node
as OpenIGTLink client or server:
1: Server
2: Client
1
[ROS-IGTL-Bridge] Input socket port:
18944
```

Loading 3D Patient Model on rviz (1)

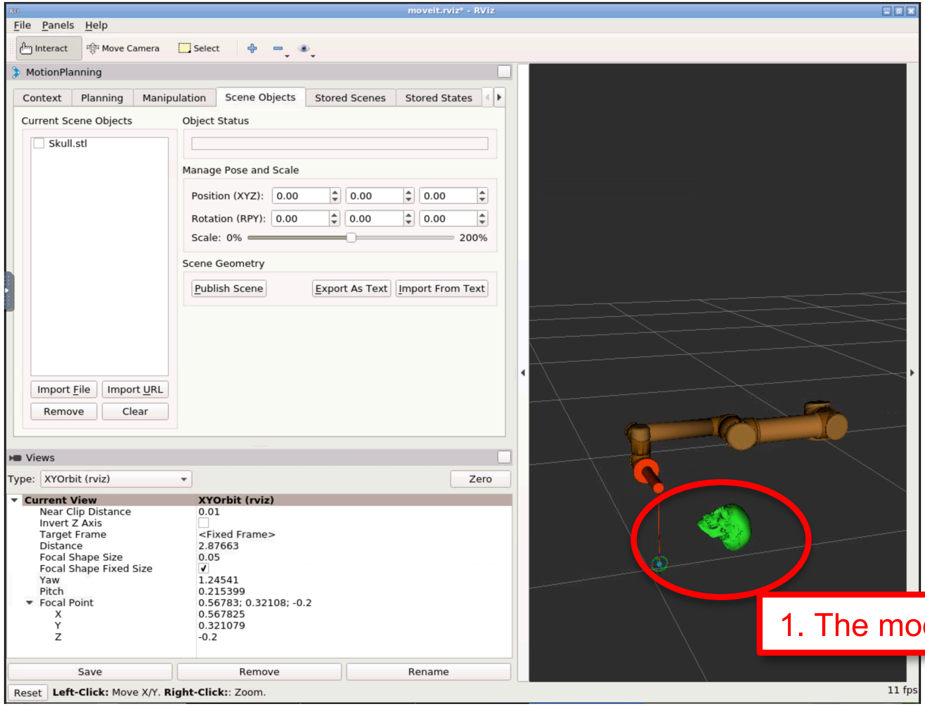
- Import a scene file with a patient model (skull).





Loading 3D Patient Model on rviz (2)

- Check if the patient model has appeared.



1. The model appears as a green skull.

Step 3: Planning Procedure on 3D Slicer

Overview of Step 3

- Surgical planning using 3D images (e.g. MRI, CT)
  - Considerations:
    - Target location
    - Entry point on the skin
    - Critical structures around the needle path
  - Output: needle trajectory
    - Coordinates of the target (patient coordinate system)
    - Coordinates of the entry point (patient coordinate system)

Loading Image and Model of the Patient (1)

- Download zipped scene data from http://bit.ly/2HTFHTl and rename to "SlicerScene-ISMR19.zip". Alternatively, you can download using wget command:

```
$ cd <working folder>
$ wget –O SlicerScene-ISMR19.zip http://bit.ly/2HTFHTl
```

- Decompress the .zip file. On a Linux/Mac terminal:

```
$ unzip SlicerScene-ISMR19.zip
```

Loading Image and Model of the Patient (2)

- The scene contains:
  - Scene description (.mrml format)
  - MR image (.nrrd format)
  - 3D surface model of the skull (.stl format)
  - 3D surface models of the robot arm (.stl format)
  - Transforms of the links of the robot arm (.tfm format)

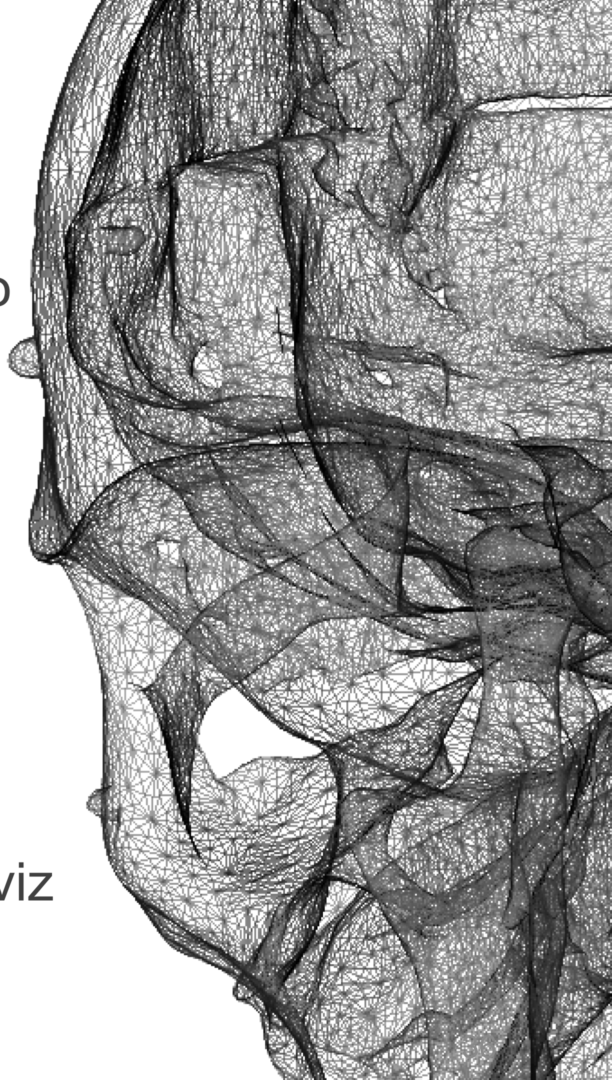
Notes on STL file format

- A STL (stereolithography) file contains the coordinates of vertices and connection of them to define polygons.

- The STL format is widely used to exchange 3D surface model data.

- The file format does not contain the unit.
  - ‘Meter’ is used in rviz.
  - ‘Millimeter’ is used in 3D Slicer

- STL files must be scaled for each software (i.e. rviz and 3D Slicer)

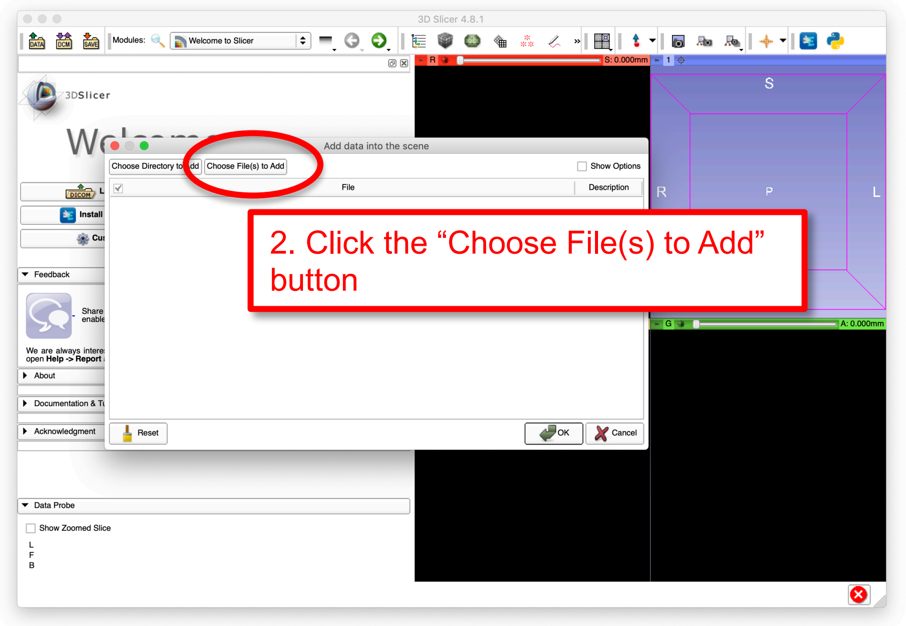


Loading Image and Model of the Patient (3)

- Launch 3D Slicer.

- Click “Data” icon on the menu bar.

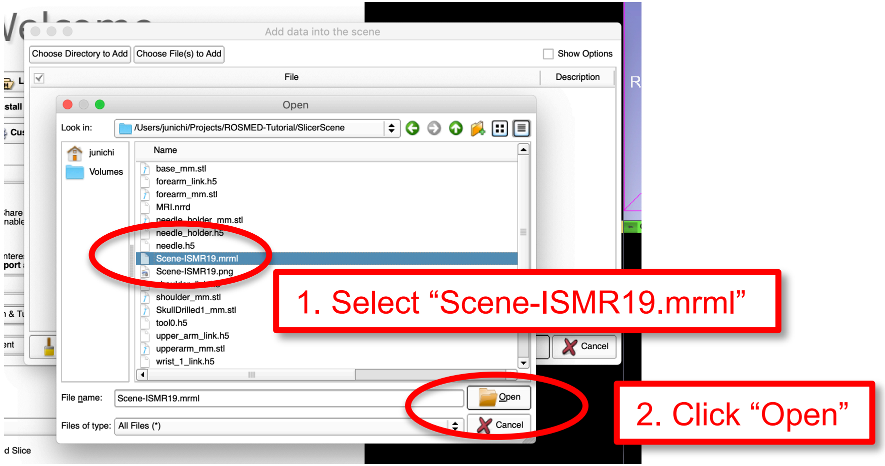
- Click “Choose File(s) to Add”



BRIGHAM AND WOMEN’S HOSPITAL

Loading Image and Model of the Patient (4)

- In the “Open” dialog box, select ”Scene-ISMR19.mrml” in the folder extracted from the .zip archive.



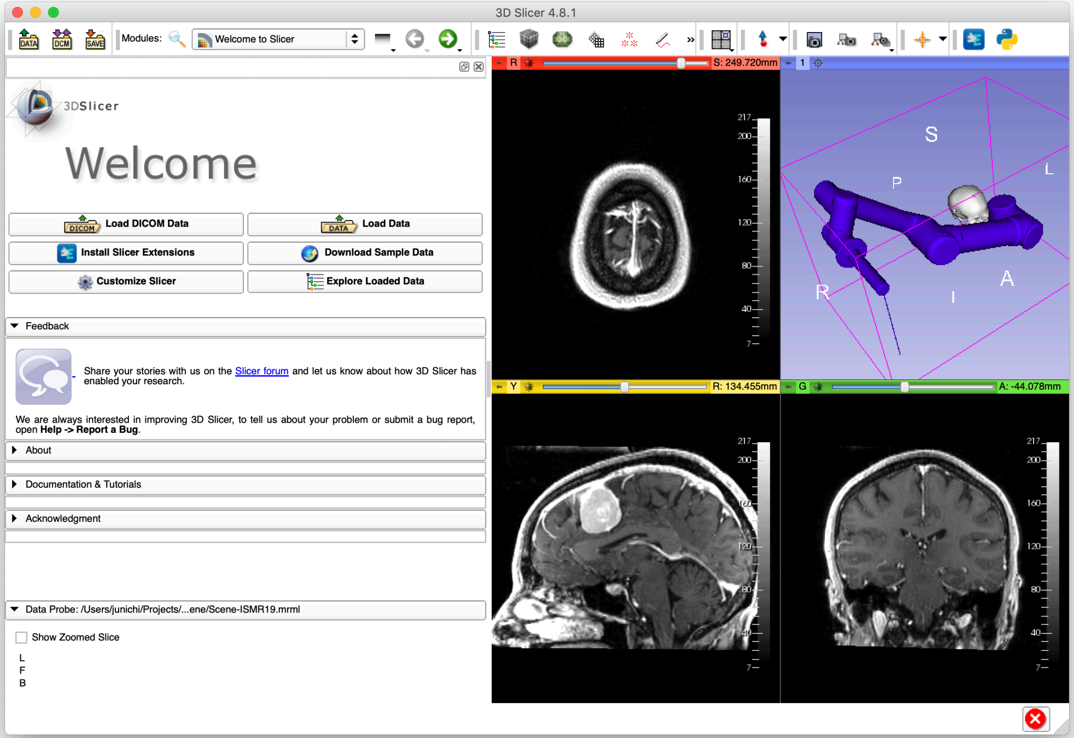
- Click “Open”.

- Click “OK” on the “Add data into the scene” dialog box.

BRIGHAM AND WOMEN’S HOSPITAL

Loading Image and Model of the Patient (5)

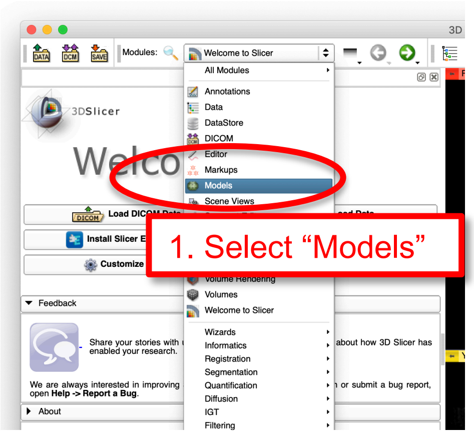
- MR image and 3D models appear on 3D Slicer.

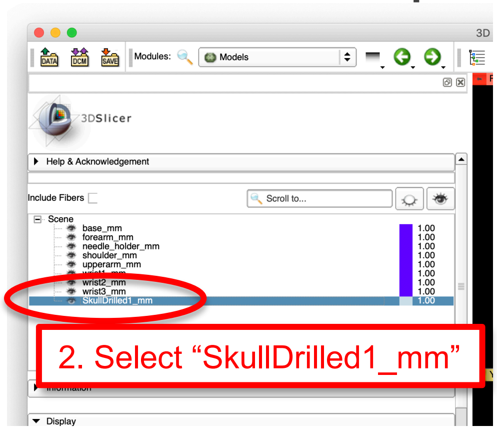


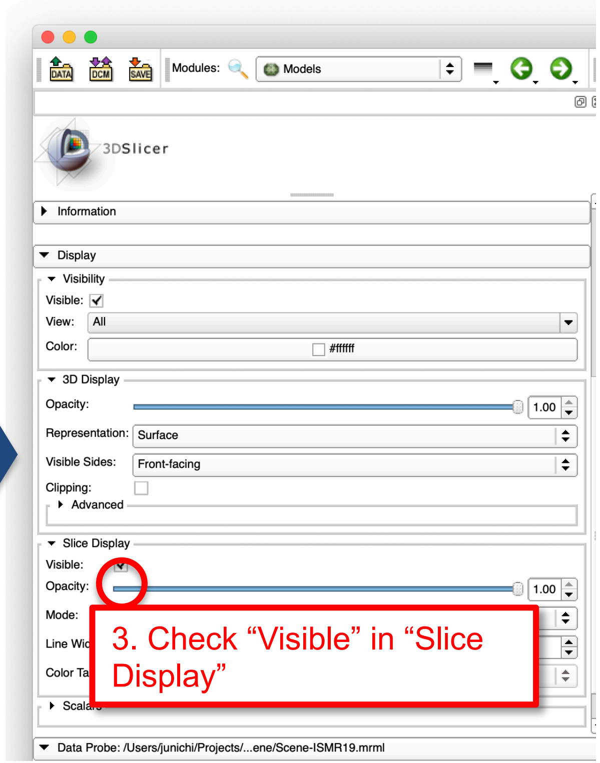
BRIGHAM AND WOMEN'S HOSPITAL

Showing Cross Section of the Skull Model (1)

- To show the cross section of the model on the 2D views:
  - Open the “Models” module.
  - Choose “SkullDrilled1_mm”
  - Check “Visible” in the “Slice Display”



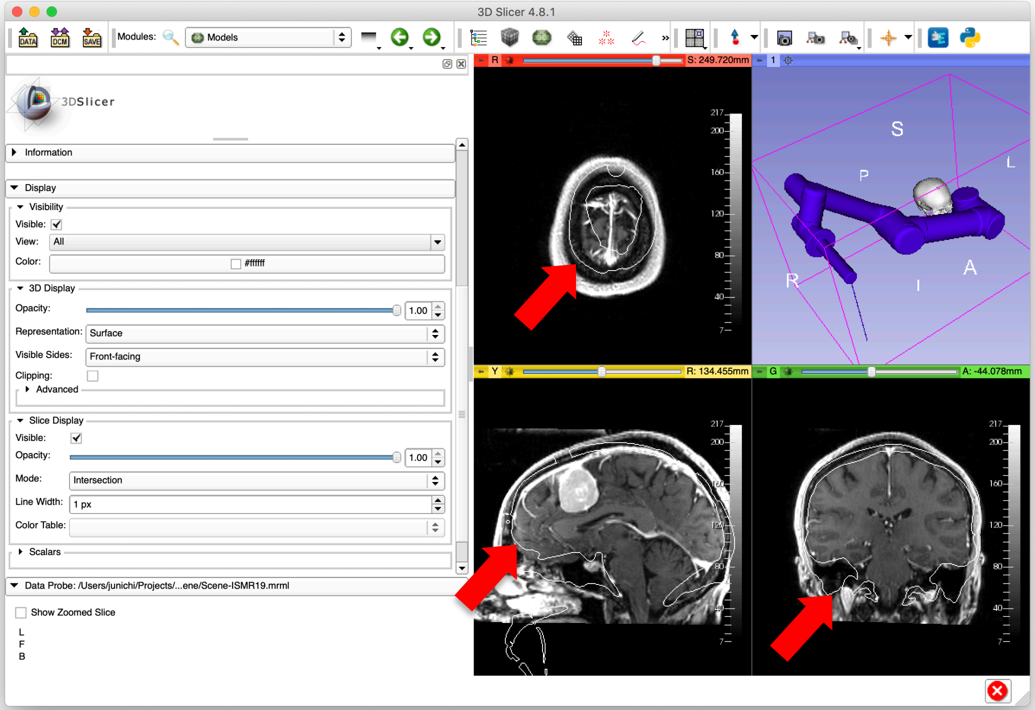




BRIGHAM AND WOMEN’S HOSPITAL

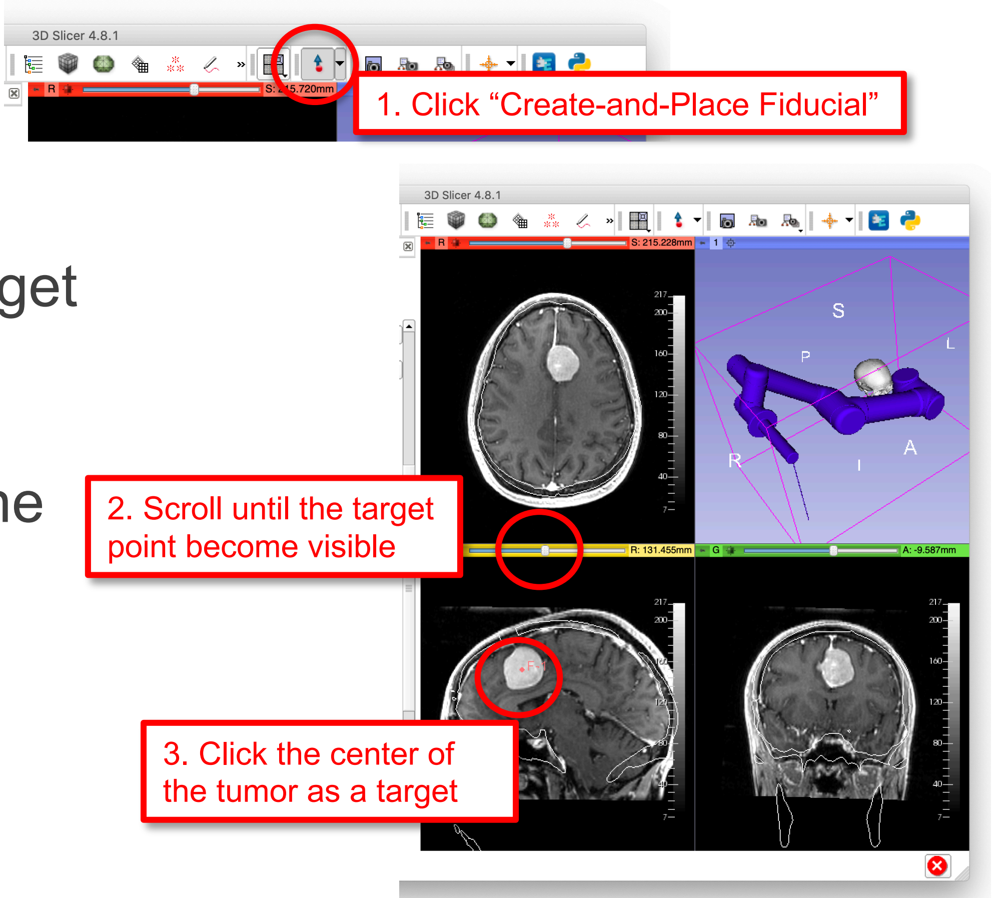
Showing Cross Section of the Skull Model (2)

- The cross sections of the skull model show up on the 2D views.



Selecting Target Point

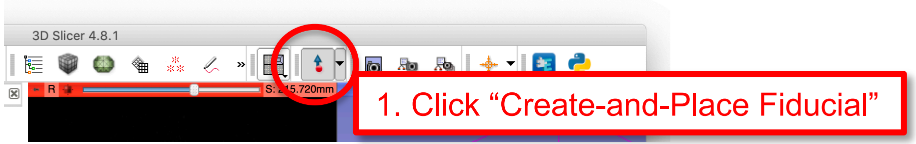
- Click "Create-and-Place Fiducial" button.
- Scroll 2D view to find the target point (center of the tumor)
- Place a fiducial by clicking the target point on the 2D view (either red, yellow, or green view)

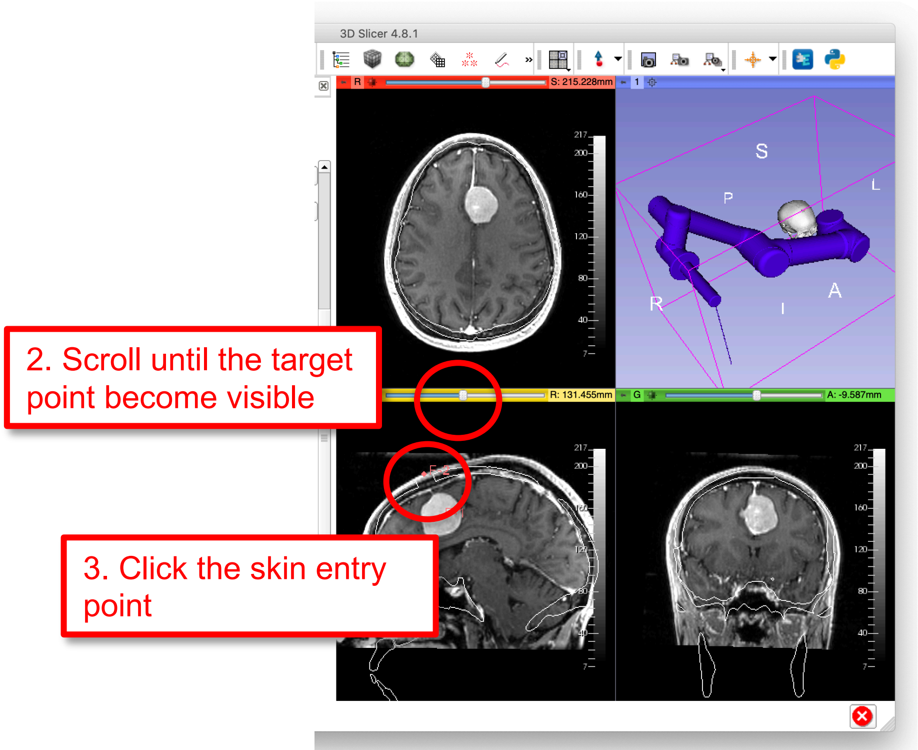


BRIGHAM AND WOMEN'S HOSPITAL

Selecting Entry Point

- Click “Create-and-Place Fiducial” button.
- Scroll 2D view to find the entry point (on the skin above the tumor)
- Place a fiducial by clicking the entry point on the 2D view (either red, yellow, or green view)

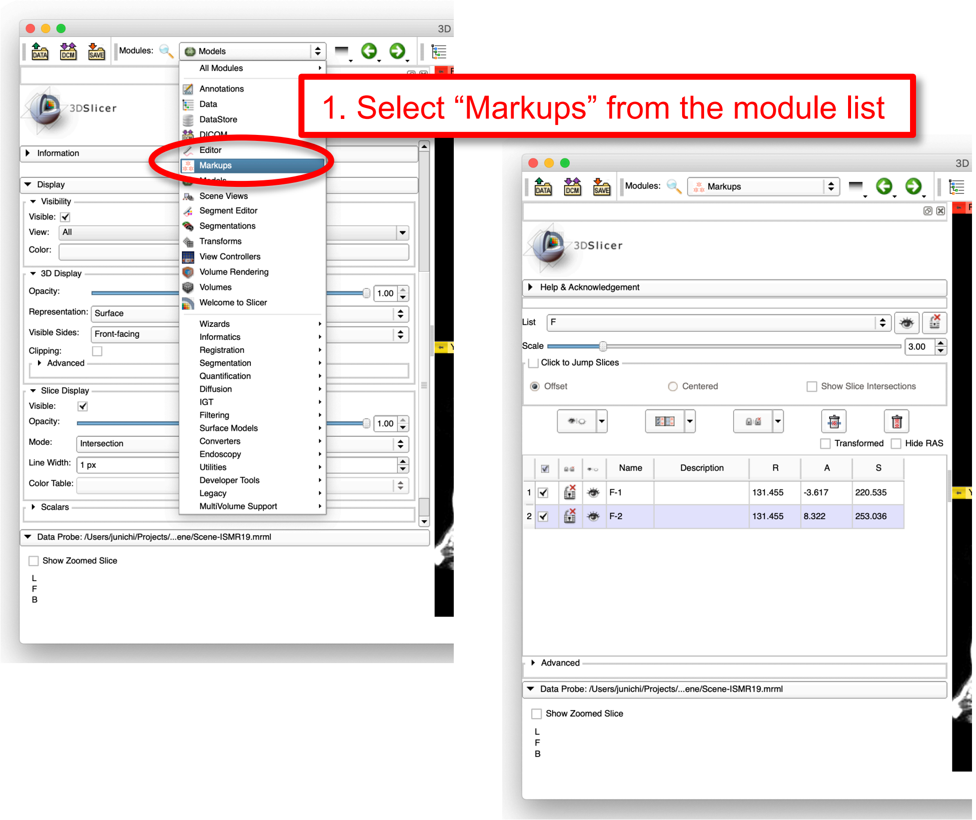




Naming Target and Entry Points (1)

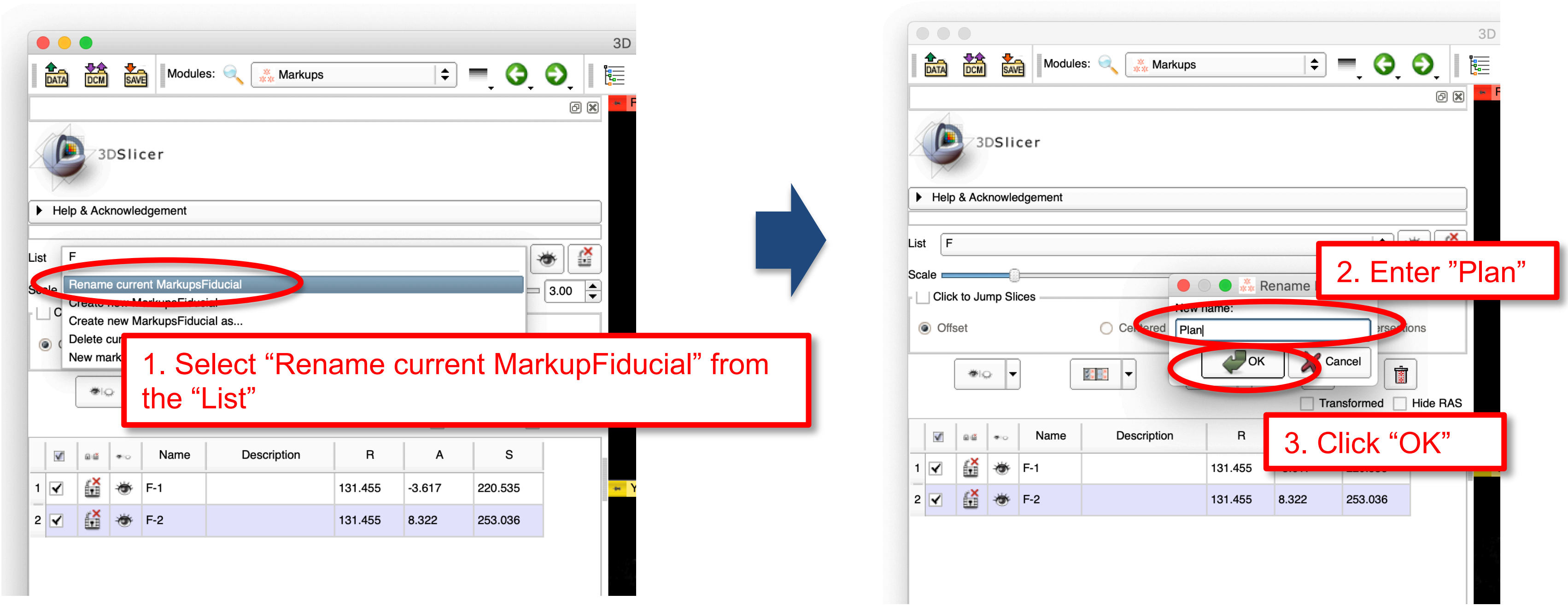
- Open “Markups”.

- Please note:
  - The coordinates and the names of the points will be exported to ROS over OpenIGTLink.
  - The names edited in this step will be used for identification on ROS, and are case-sensitive.



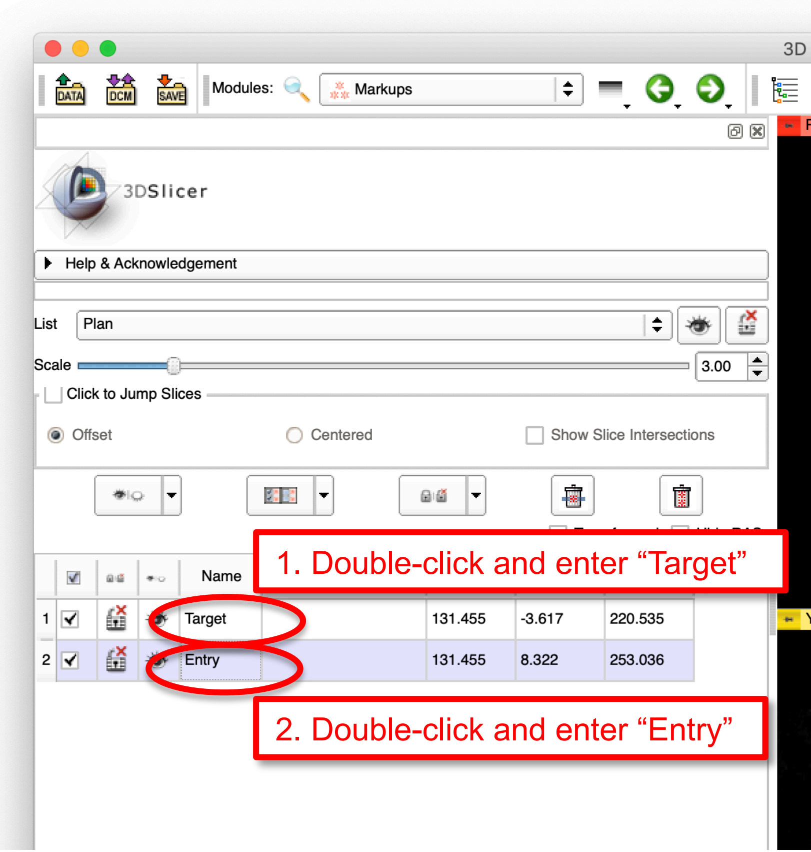
Naming Target and Entry Points (2)

- Rename the list (MarkupFiducials) to "Plan"



Naming Target and Entry Points (3)

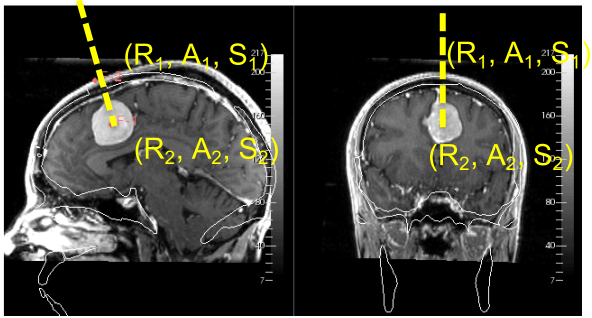
- Rename the first point to “Target”.
- Rename the second point to “Entry”.



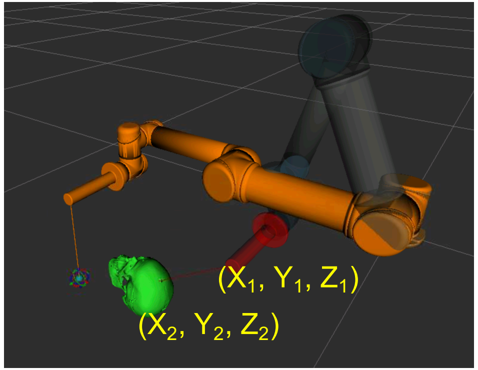
Step 3: Robot-to-Image Registration

Overview of Step 4

- Two coordinate systems
  - Points on the images are defined in the **patient coordinate system** (**RAS** coordinate system)
  - Points for execution are given in the **robot coordinate system** (**Reference** coordinate system)



Surgical Plan (RAS)



Execution (Reference)

Overview of Step 4

- Fiducial registration
  - Estimate a transformation matrix between the **RAS** and **Reference** coordinate systems by matching the corresponding landmarks
  - Outcome of registration is a matrix that transforms coordinates from the Reference coordinate system to the RAS coordinate system ($\boldsymbol{T_{Ref,RAS}}$)

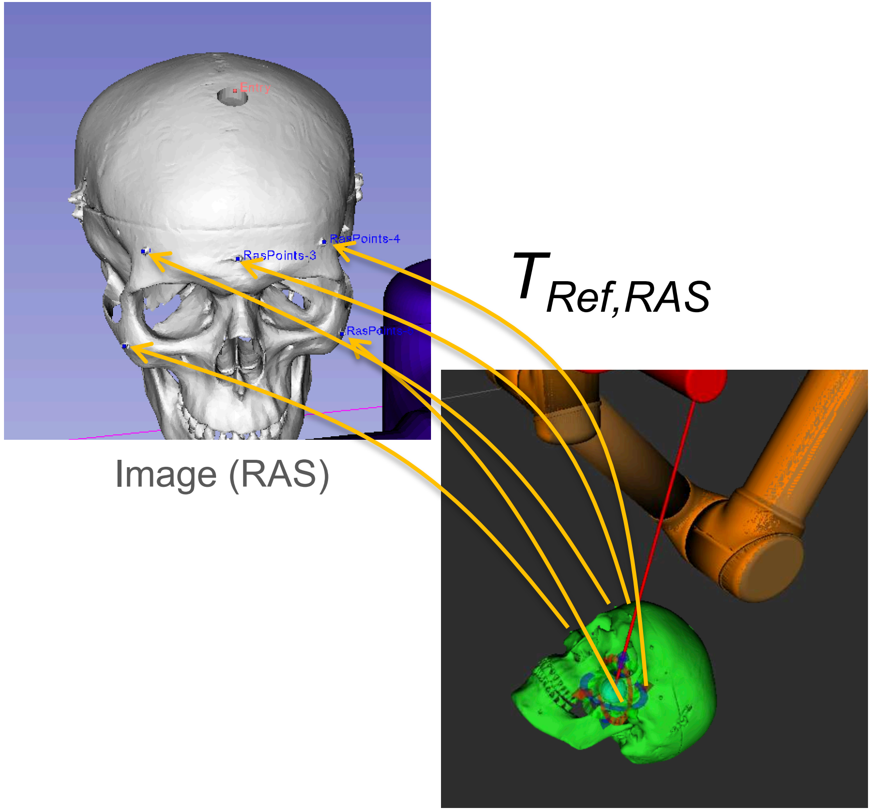
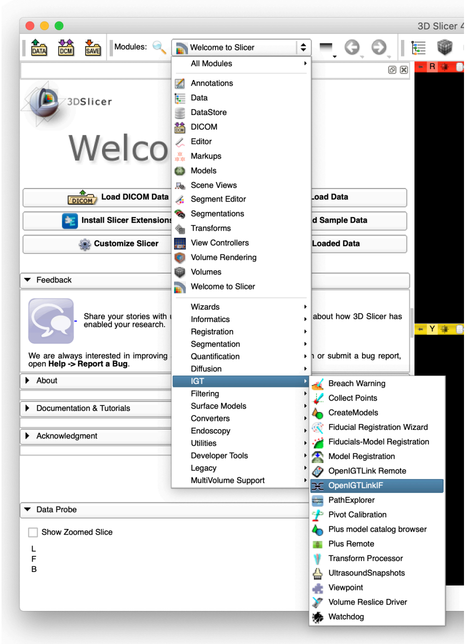


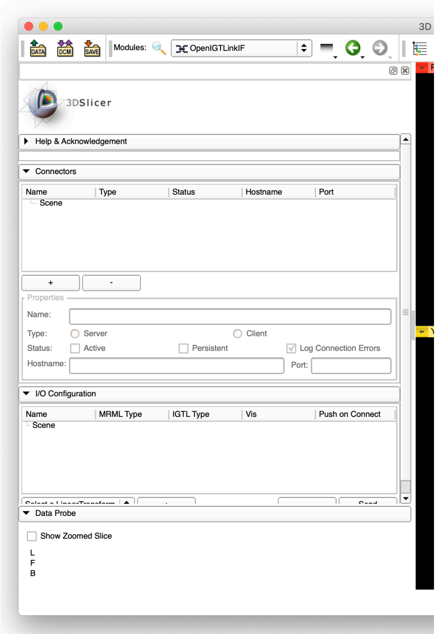
Image (RAS)

Robot (Reference)

Connecting from 3D Slicer (1)

- In 3D Slicer, Open “Modules” -> “IGT” -> “OpenIGTLink IF”



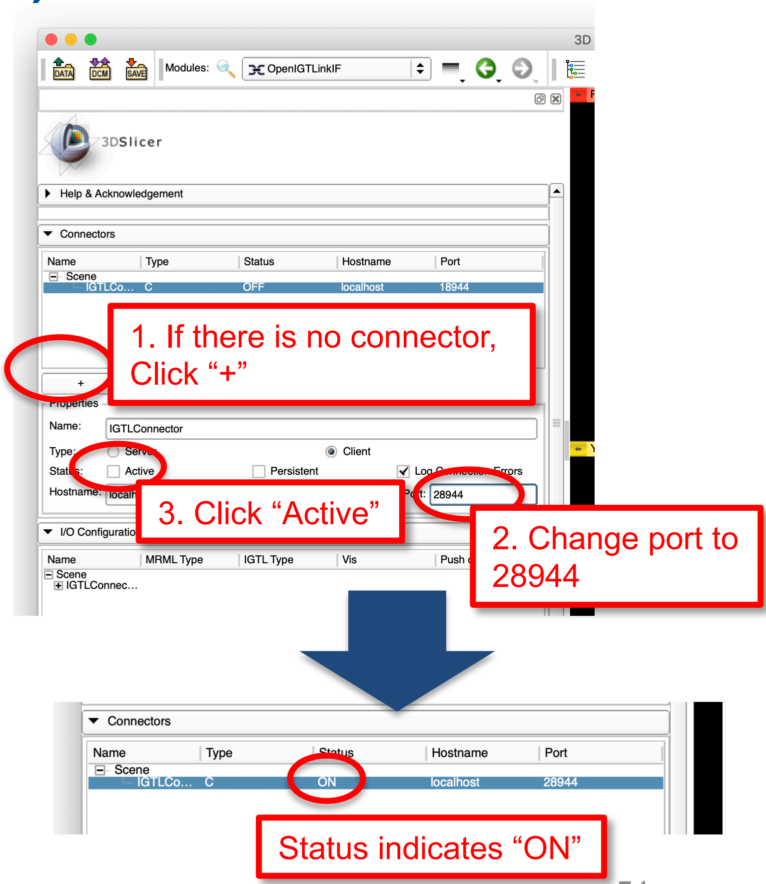


Connecting from 3D Slicer (2)

- Configure a connector node and configure:

Name	"IGTLConnector" (default)
Type	"Client" (default)
Status	Unchecked (default)
Hostname	"localhost" (if Docker is used)
Port	"28944"

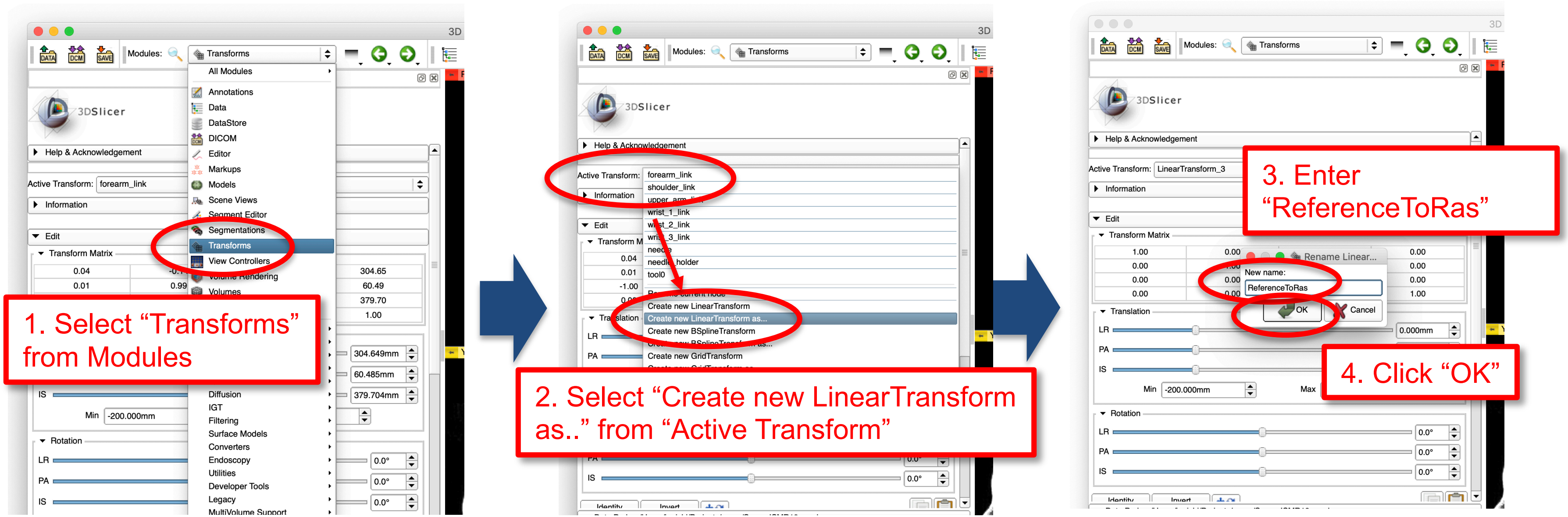
- After configuring the node, click "Active". If successful, the status becomes "ON".



BRIGHAM AND WOMEN'S HOSPITAL

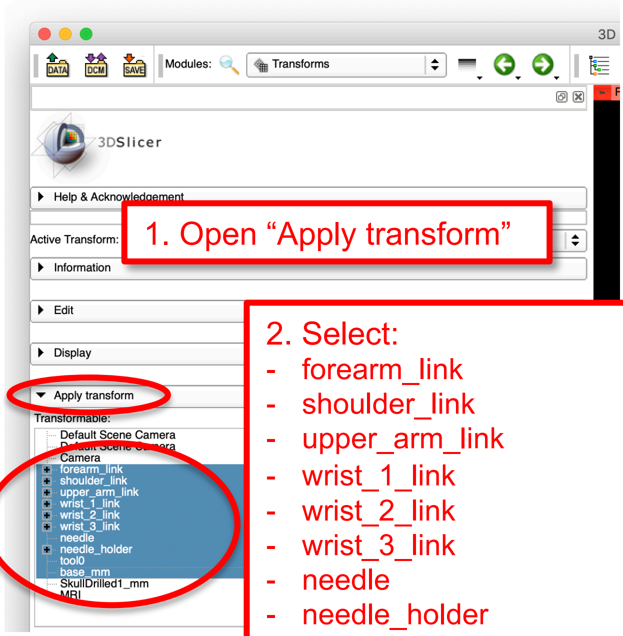
Creating Registration Transform (1)

- Open “Transforms” module.
- Create a new transform as “ReferenceToRas”.



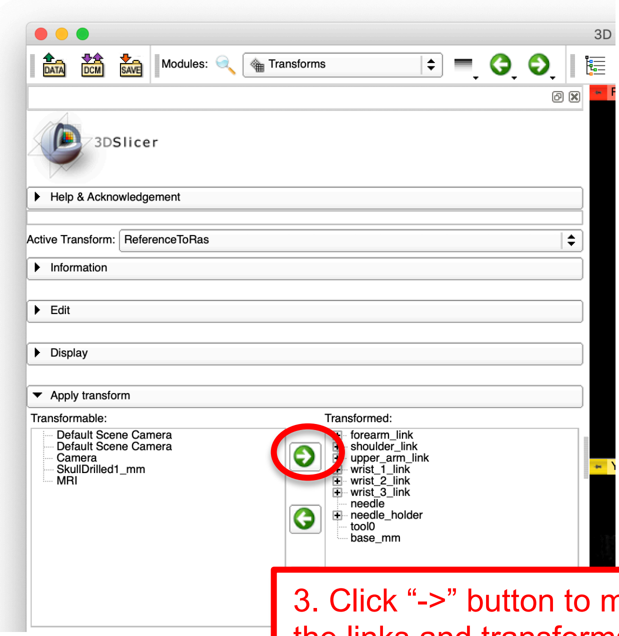
Creating Registration Transform (2)

- In “Apply transform” frame, move the all links of the robot and their transforms to ”Transformed”.



1. Open “Apply transform”

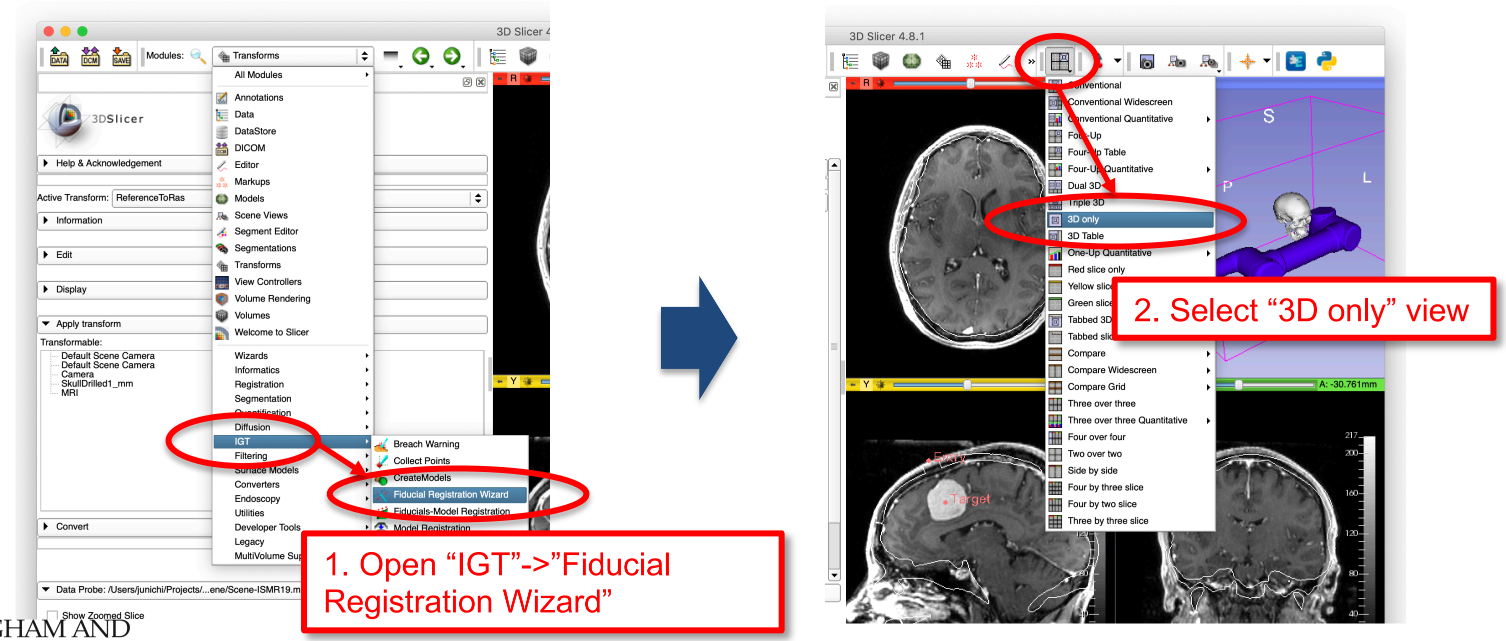
2. Select:
- forearm_link
- shoulder_link
- upper_arm_link
- wrist_1_link
- wrist_2_link
- wrist_3_link
- needle
- needle_holder
- tool0
- base_mm



3. Click “->” button to move the links and transforms to “Transformed”

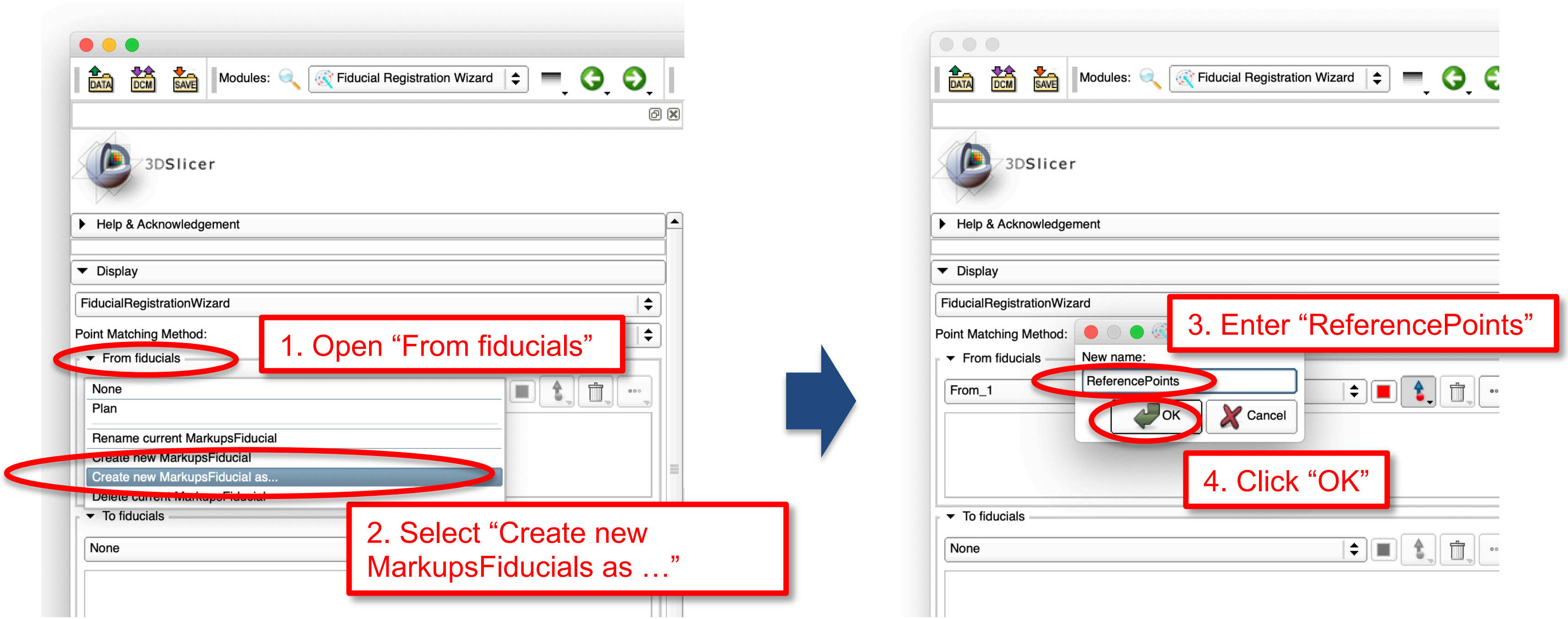
Fiducial Registration Wizard

- Open ”IGT” -> “Fiducial Registration Wizard”.
- Change to “3D only” view.



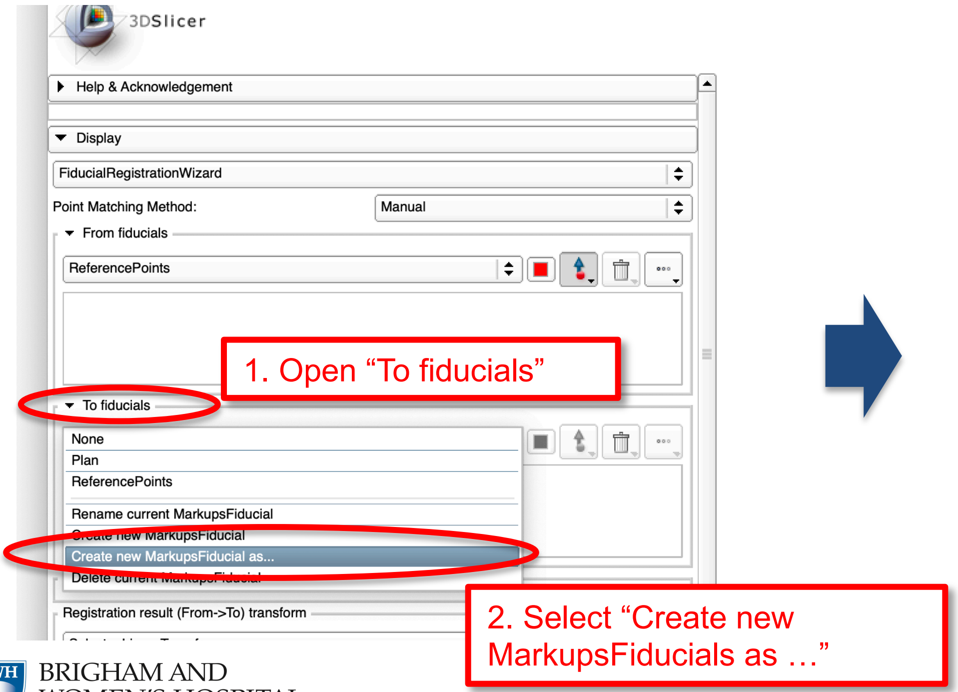
Creating Fiducial Lists (1)

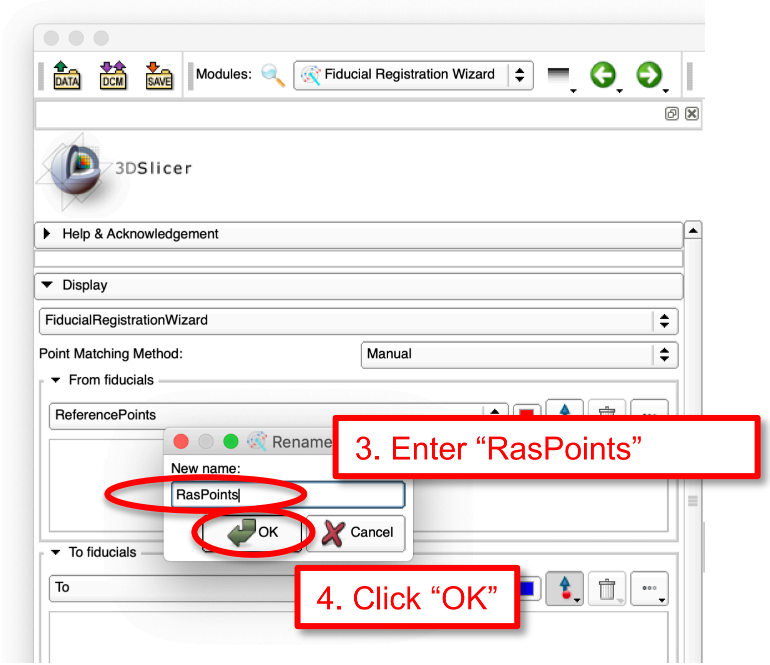
- Create a fiducial list for robot as “ReferencePoints”



Creating Fiducial Lists (2)

- Create a fiducial list for image as “RasPoints”

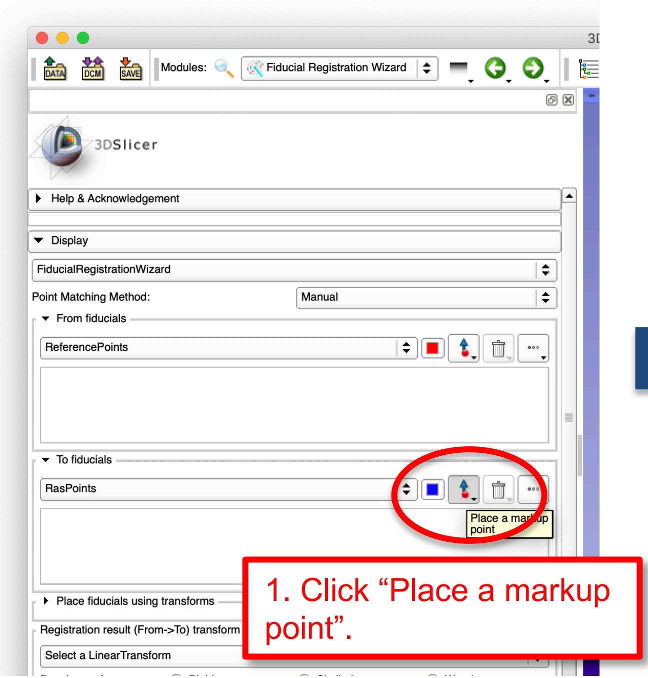


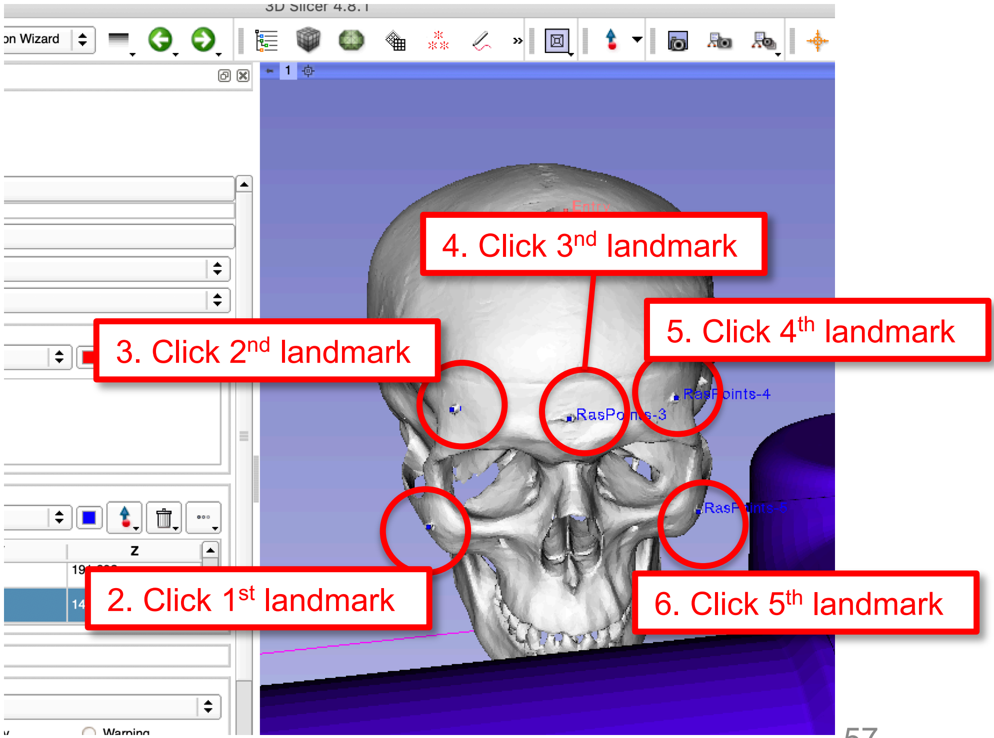


BRIGHAM AND WOMEN’S HOSPITAL

Selecting Landmarks on Image

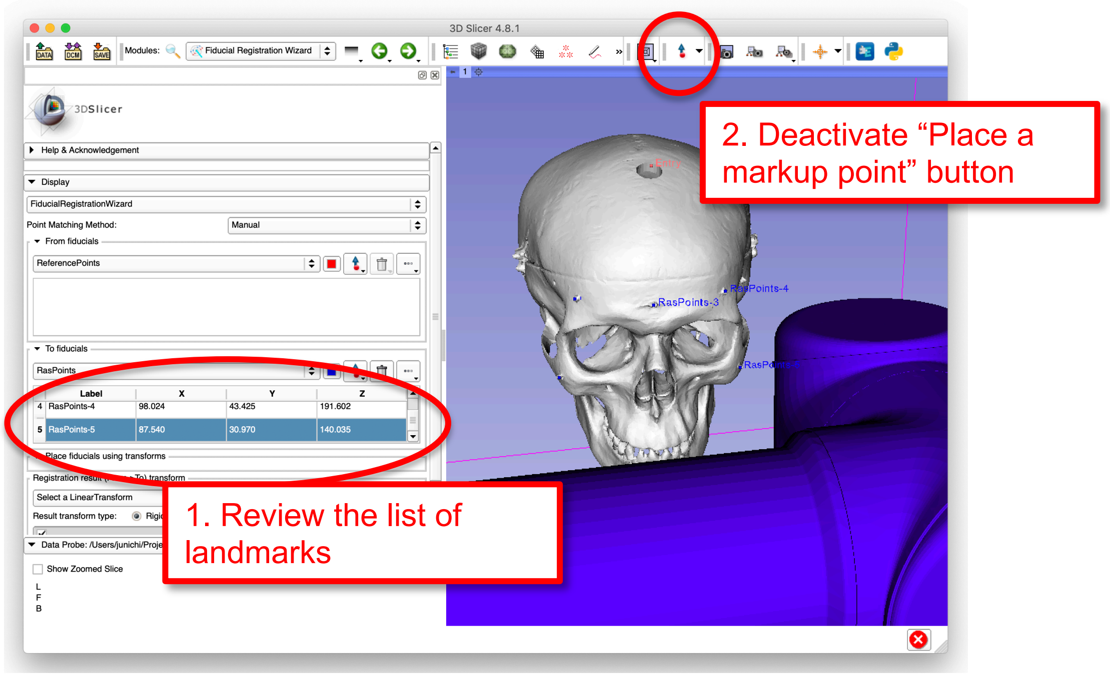
- Click five fiducial markers on the skull in the 3D view.





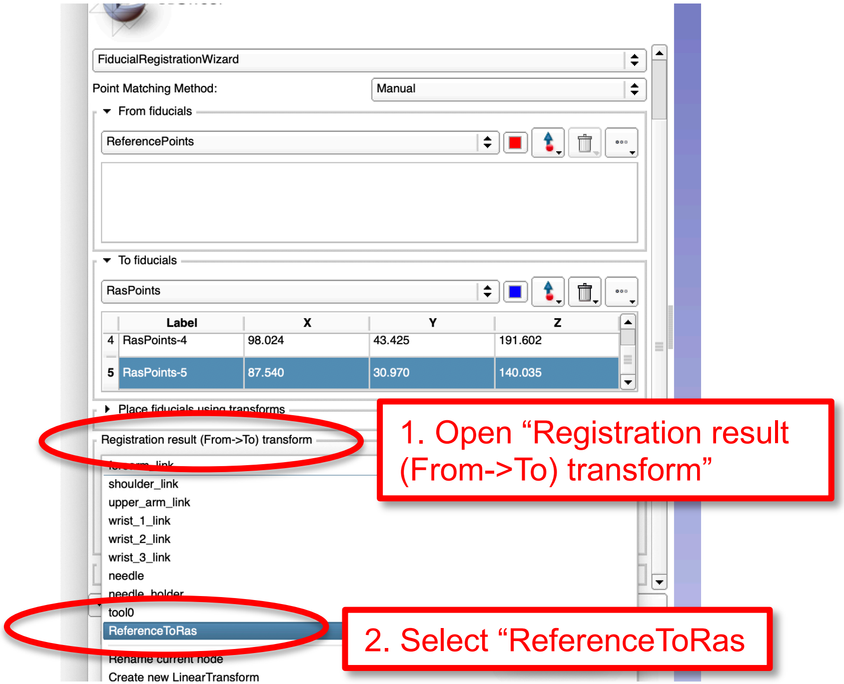
Reviewing Landmarks

- The list of the landmarks are shown in “To Fiducials”.
- Make sure to deactivate the “Place a markup point” button



Setting Registration Transform

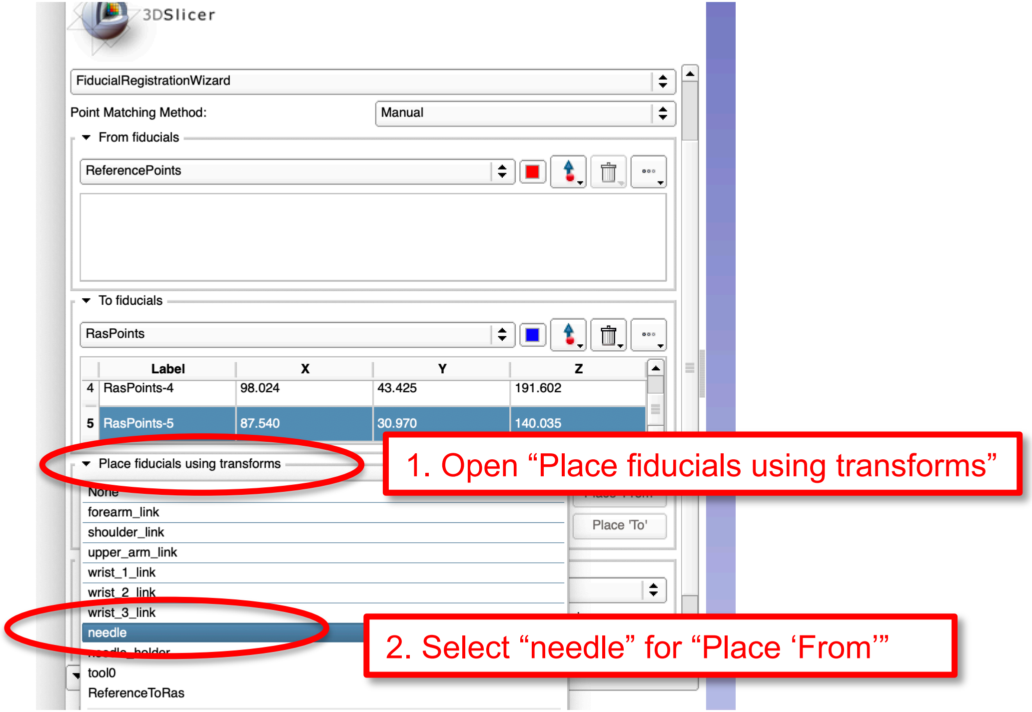
- Select "ReferenceToRas" in "Registration result transform"



BRIGHAM AND WOMEN'S HOSPITAL

Selecting Landmarks on Patient (1)

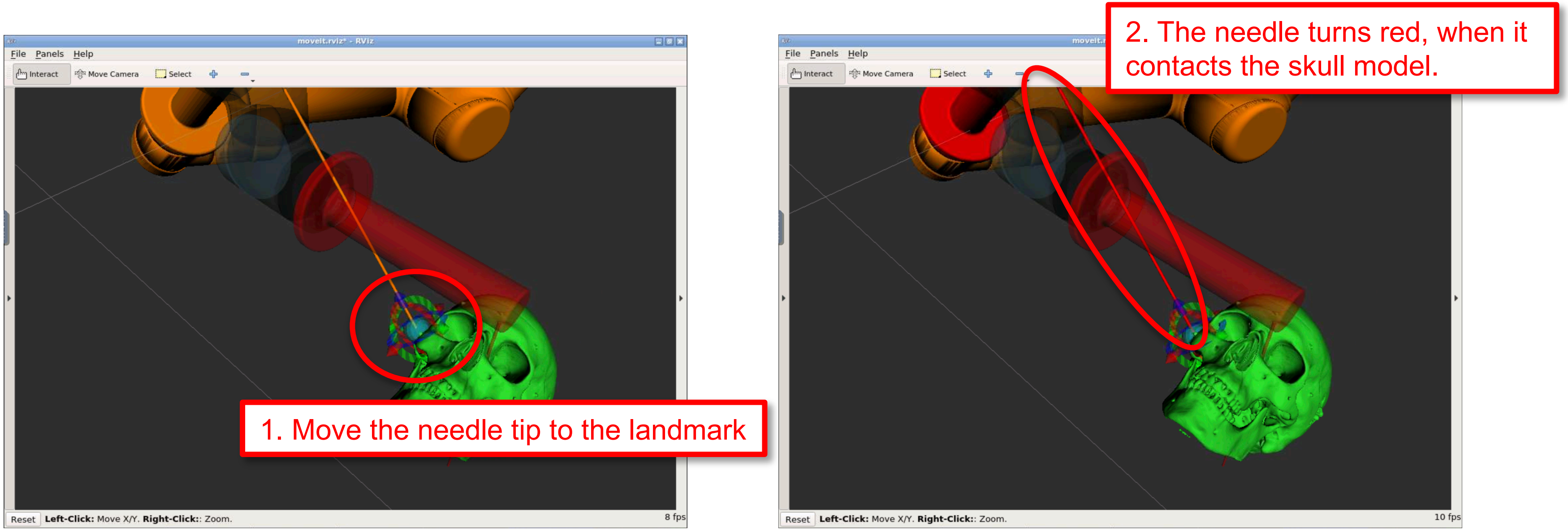
- Select the needle transform on 3D Slicer



BRIGHAM AND WOMEN'S HOSPITAL

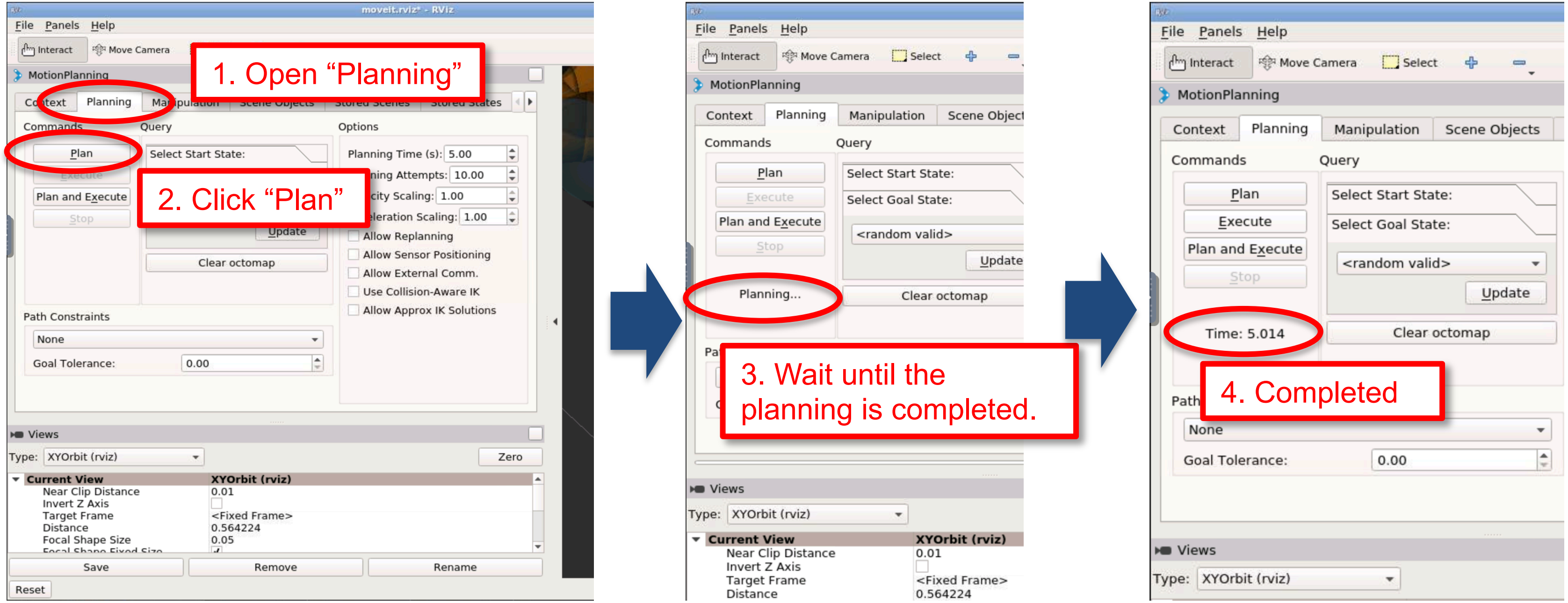
Selecting Landmarks on Patient (2)

- On rviz, move the planned needle tip to the first landmark using the arrows and the disks.



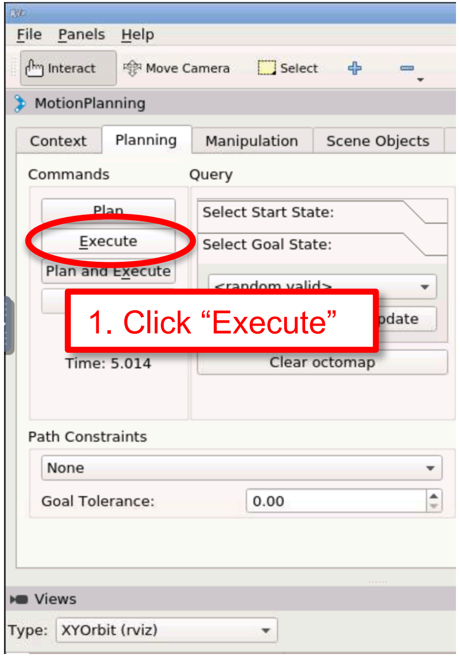
Selecting Landmarks on Patient (3)

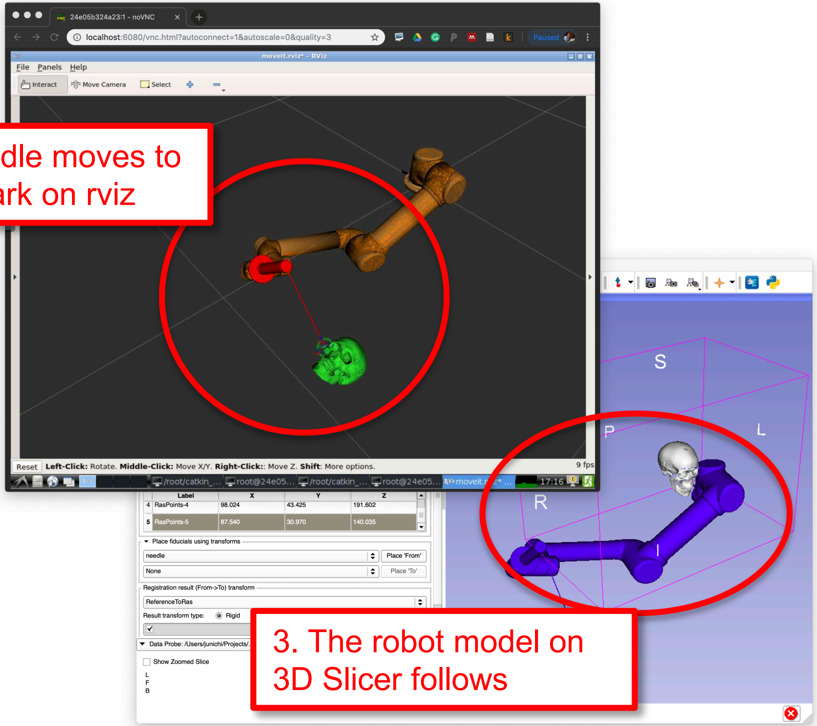
- Generate a plan for execution.



Selecting Landmarks on Patient (4)

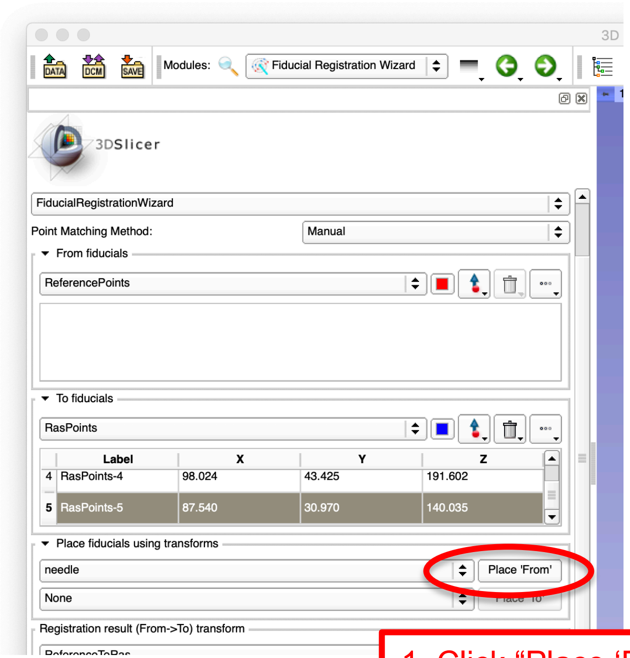
- Execute the plan to bring the needle to the landmark.



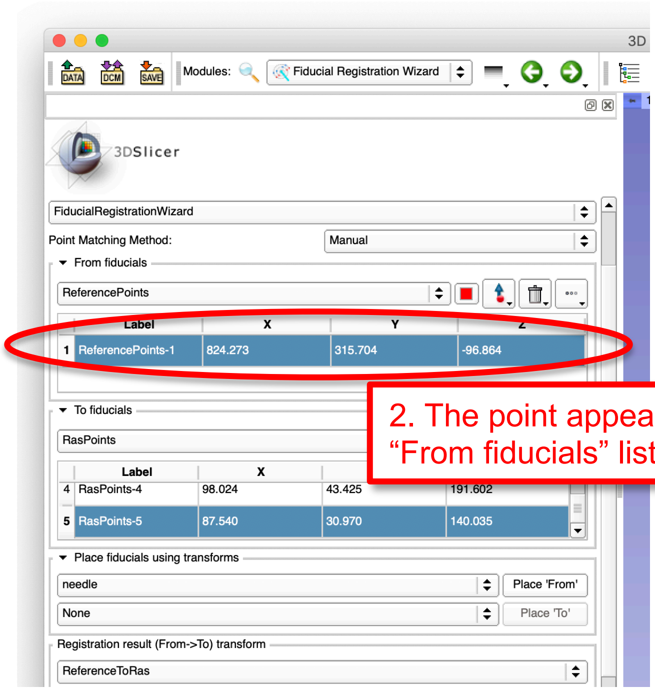


Selecting Landmarks on Patients (5)

- Record the coordinates on 3D Slicer.



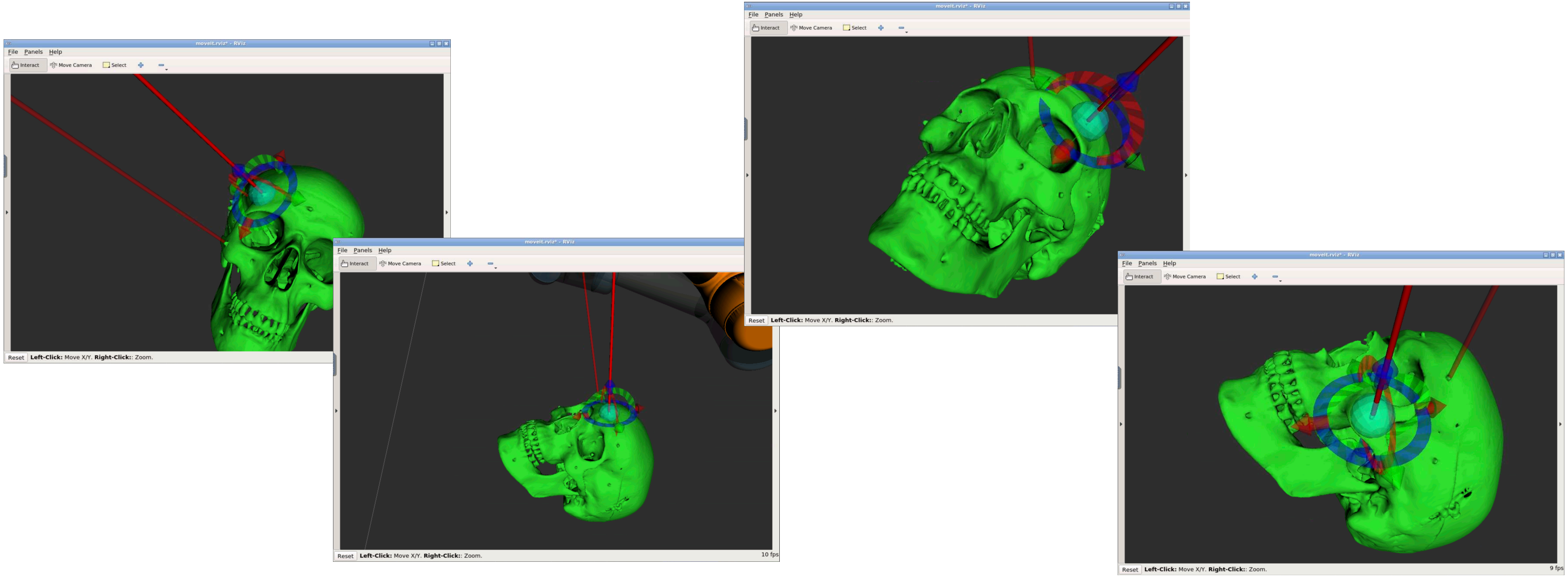
1. Click “Place ‘From’”



2. The point appears on the “From fiducials” list

Selecting Landmarks on Patients (6)

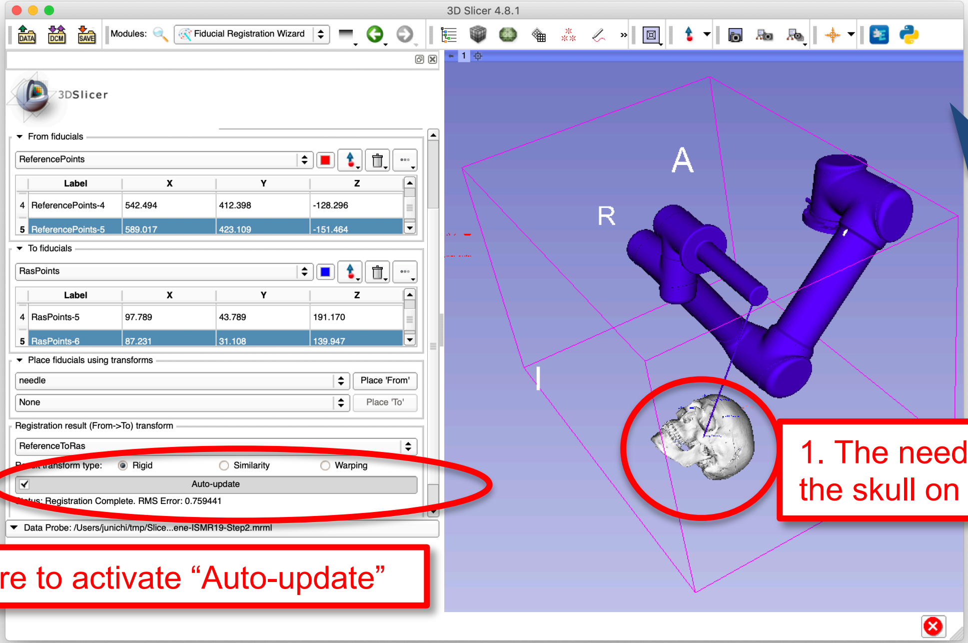
- Repeat the previous steps for the rest of the landmarks.

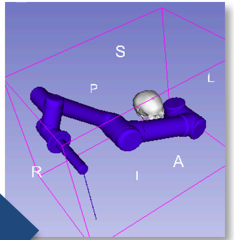


BRIGHAM AND
WOMEN'S HOSPITAL

Selecting Landmarks on Patients (7)

- The registration matrix is updated as the landmarks are recorded on 3D Slicer.





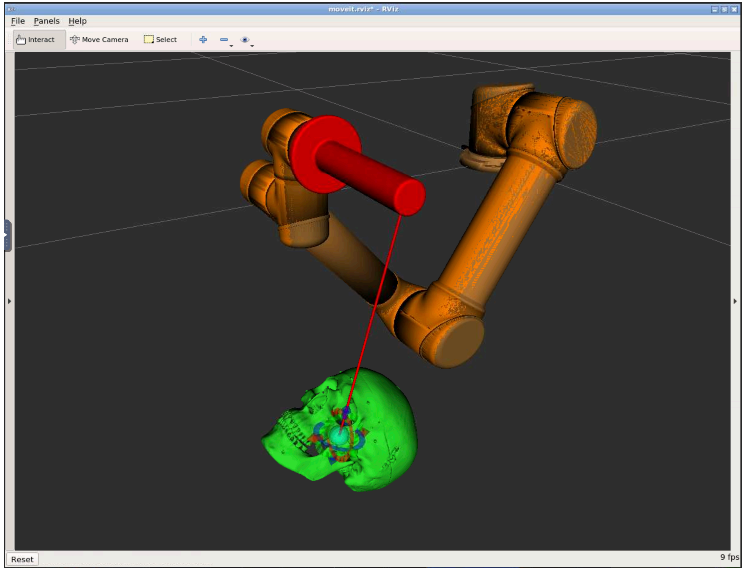
1. The needle is now pointing the skull on 3D Slicer

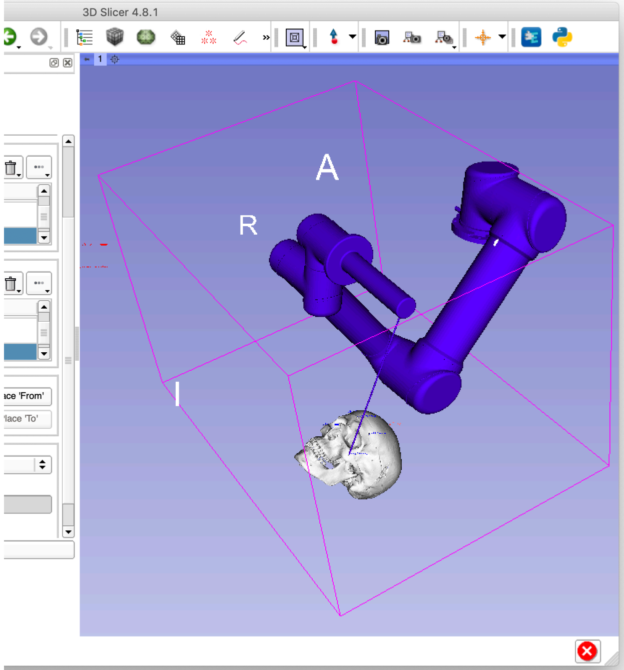
Make sure to activate "Auto-update"

BRIGHAM AND WOMEN'S HOSPITAL

Confirming Registration

- Review the relationship between the robot and the patient on rviz and 3D Slicer





Step 4: Targeting

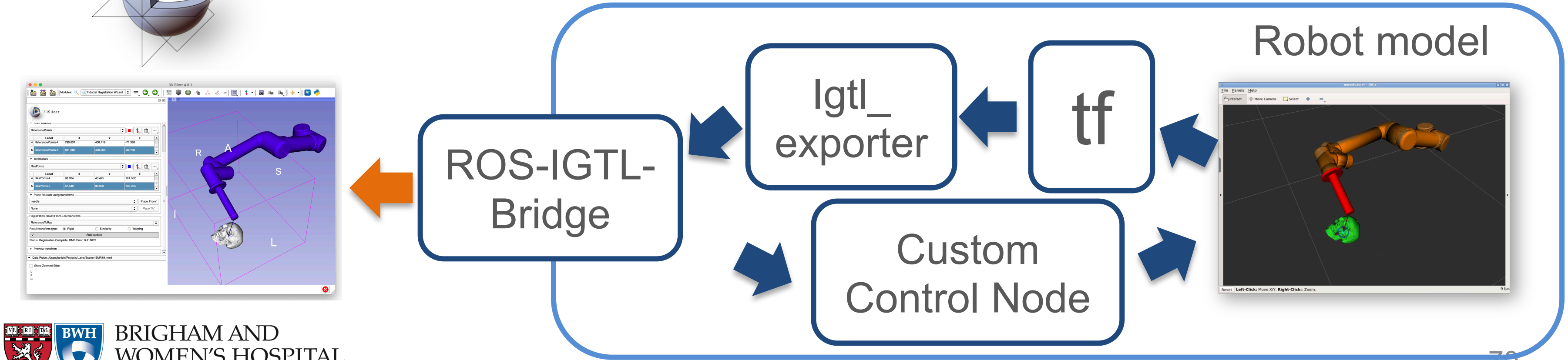
Overview of Step 4 (1)

- Coordinate systems for targeting:
  - The planned points (i.e. ‘Target’ and ‘Entry’) are defined in the image coordinate system.
  - The plan is executed by the robot arm in the robot coordinate system.

- Execution of the plan:
  - Transform points from the RAS coordinates to the Reference coordinates by applying $T_{RAS,Ref}$
  - $T_{RAS,Ref}$ is the inverse matrix of $T_{Ref,RAS}$

Overview of Step 4 (2)

- A custom control node for this tutorial:
  - Receives the coordinates of entry and target points
  - Calculate needle orientation and motion plan
  - Execute



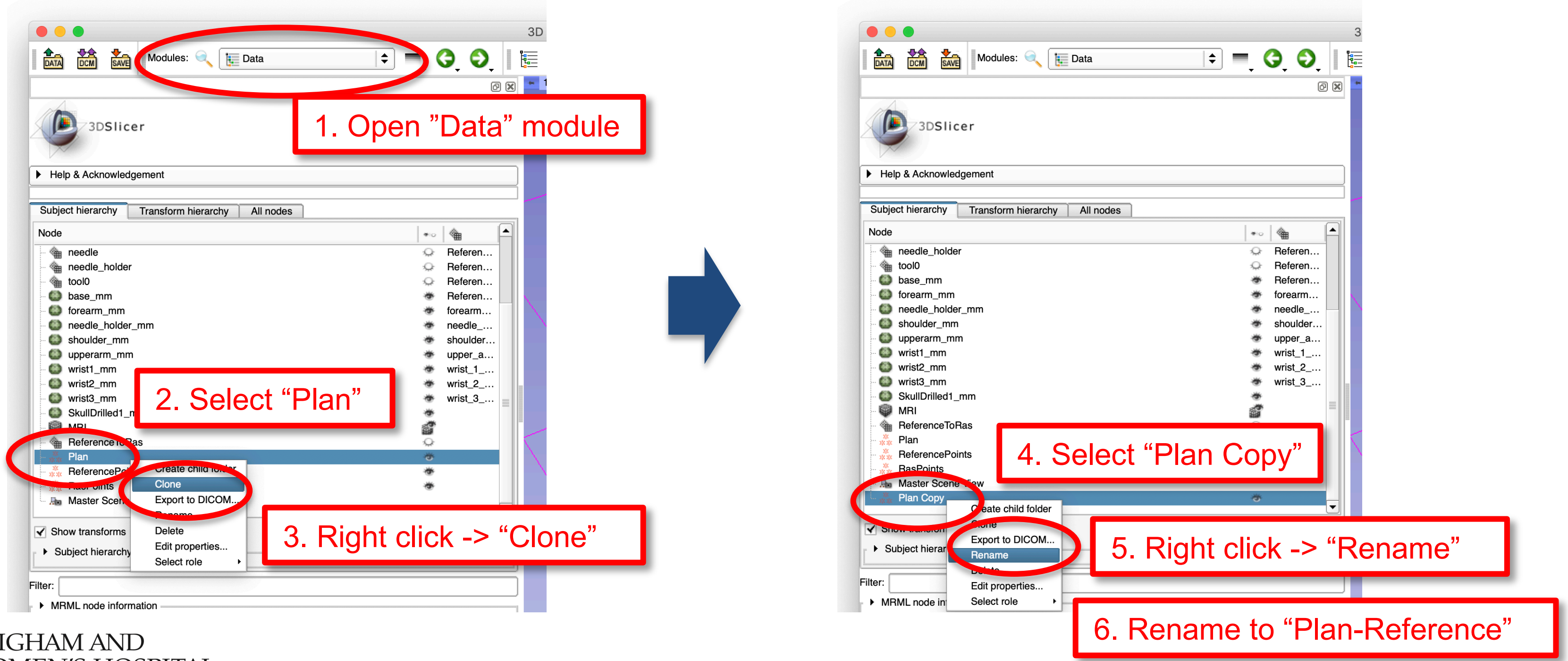
Starting Custom Control Node

- On a terminal on the ROS computer:

```
$ cd ~/catkin_ws
$ source devel/setup.bash
$ rosrun ismr19_control ismr19_control_node
```

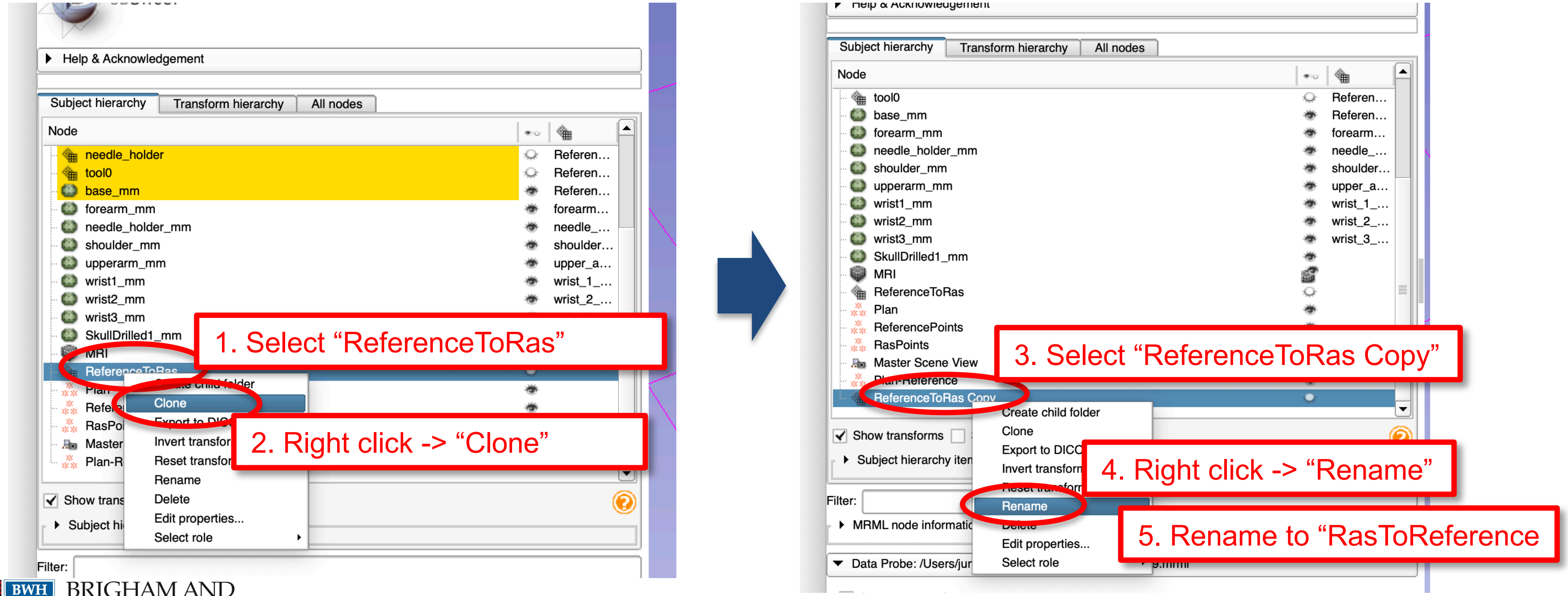
Transforming Planned Points (1)

- Copy “Plan” to “Plan-Reference”



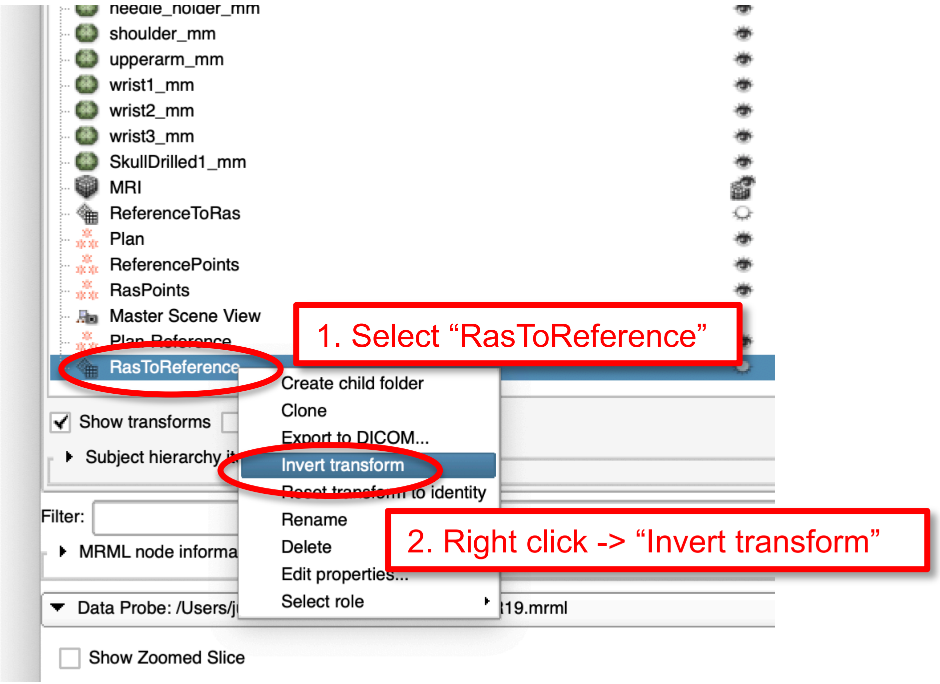
Transforming Planned Points (2)

- Copy “ReferenceToRas” to “RasToReference”



Transform Planned Points (3)

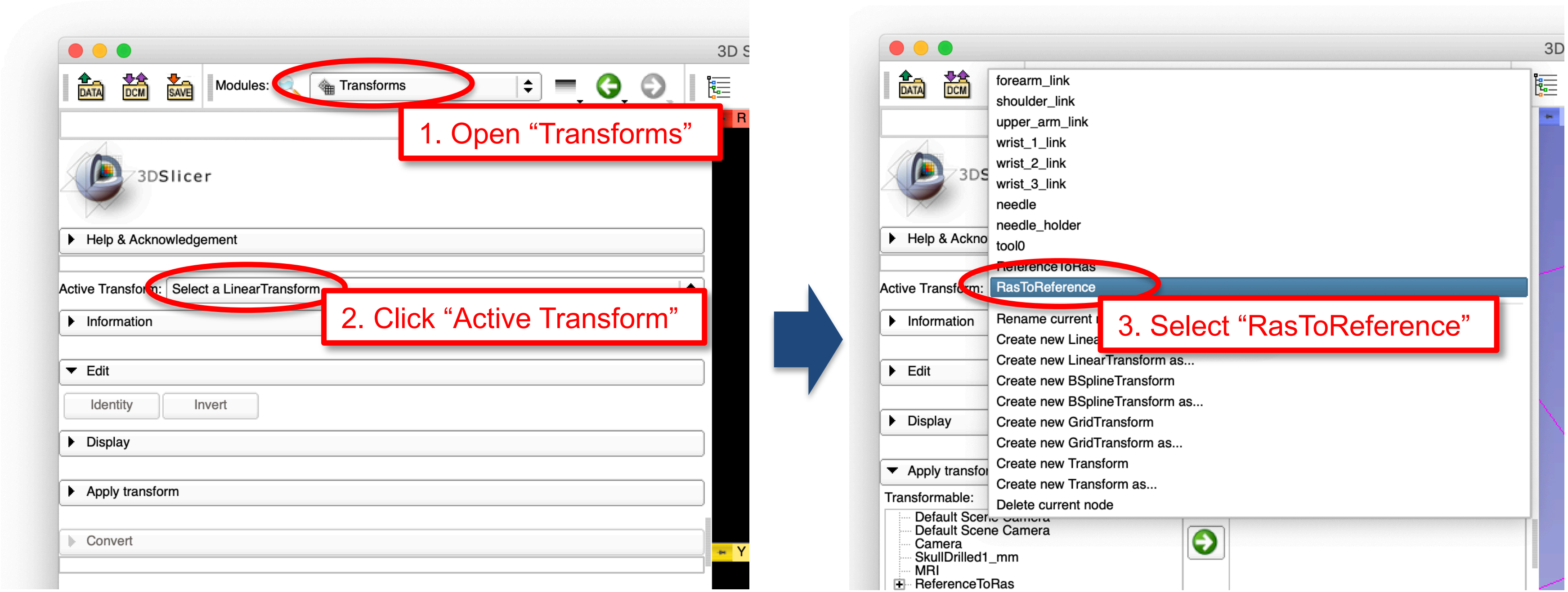
- Invert Transform



BRIGHAM AND WOMEN’S HOSPITAL

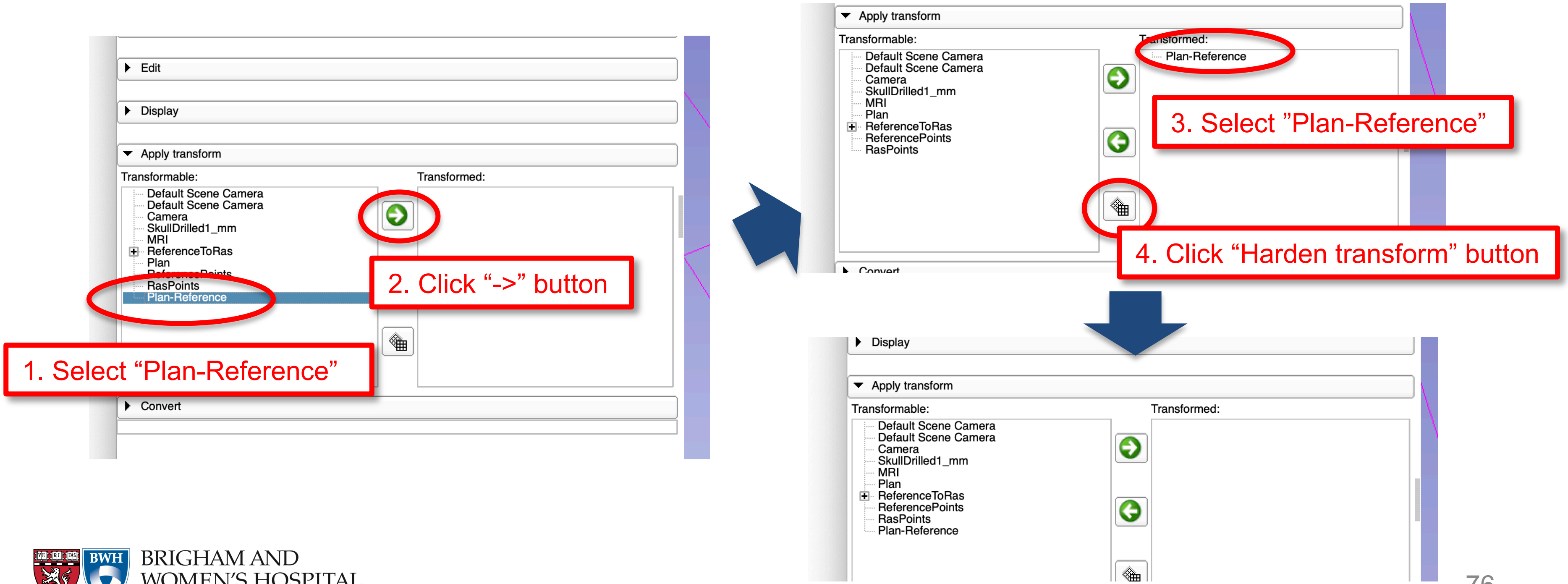
Transform Planned Point (4)

- Open “RasToReference” transform in “Transforms”



Transform Planned Points (5)

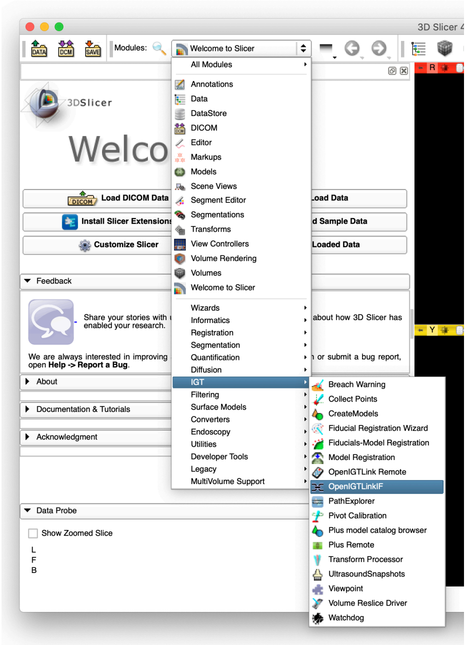
- Apply “RasToReference” transform to “Plan-Reference”

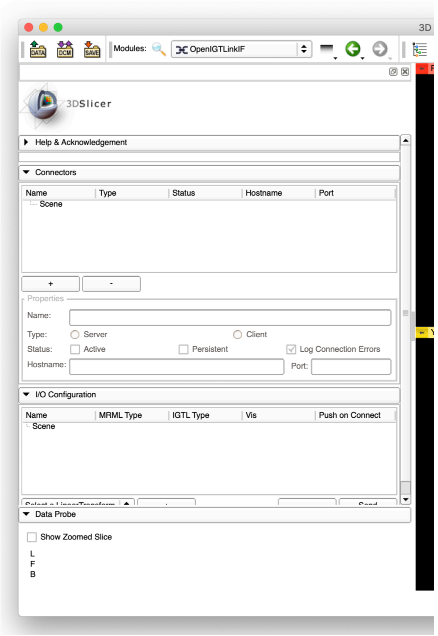


BRIGHAM AND WOMEN’S HOSPITAL

Sending Plan from 3D Slicer to ROS (1)

- Open “OpenIGTLink IF” module

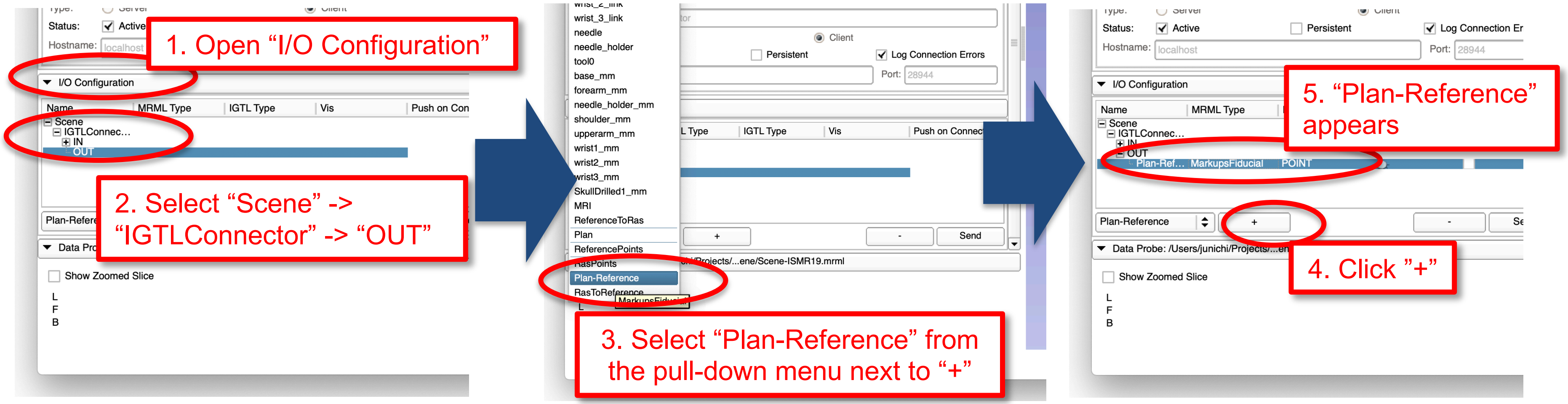




77

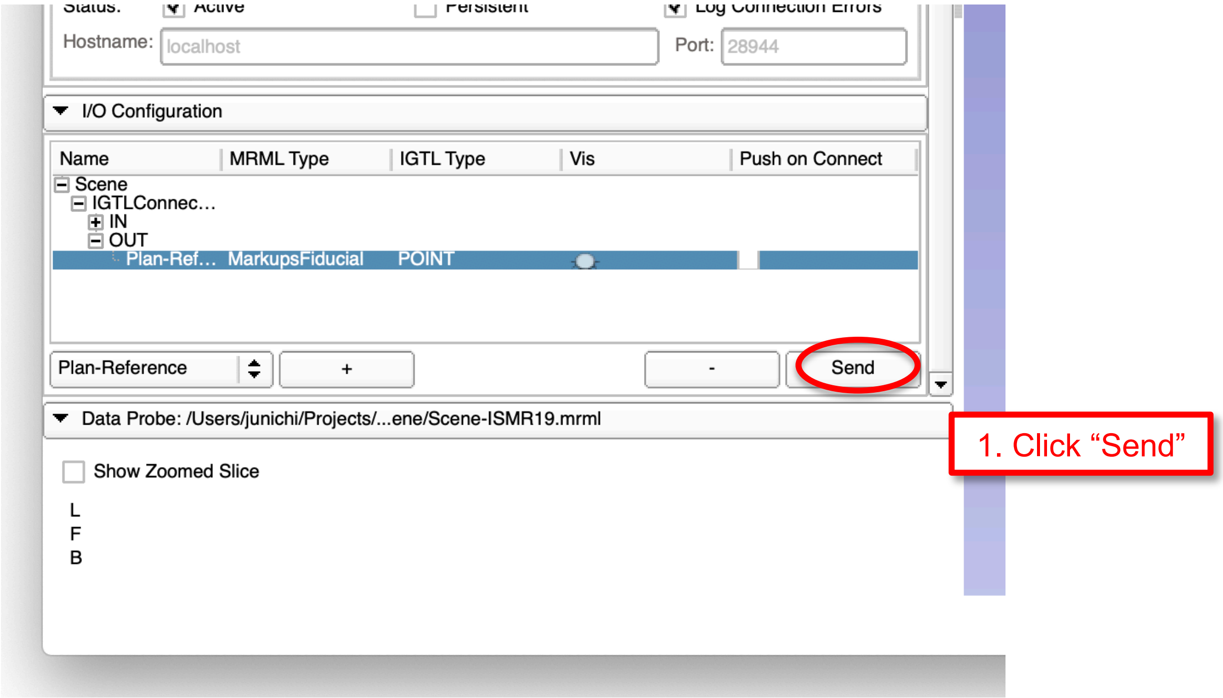
Sending Plan from 3D Slicer to ROS (2)

- Add “Plan-Reference” in “I/O Configuration”



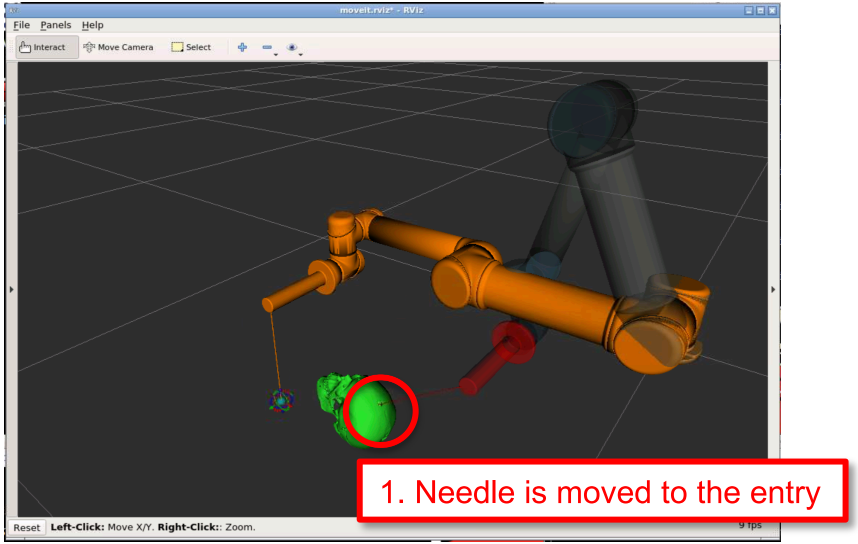
Sending Plan from 3D Slicer to ROS (3)

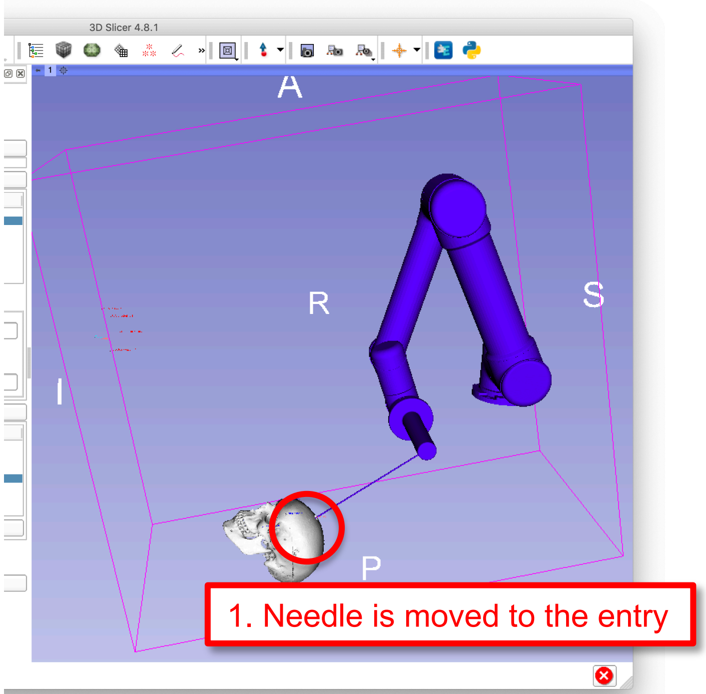
- Send “Plan-Reference” in “I/O Configuration”



Move Needle to Target

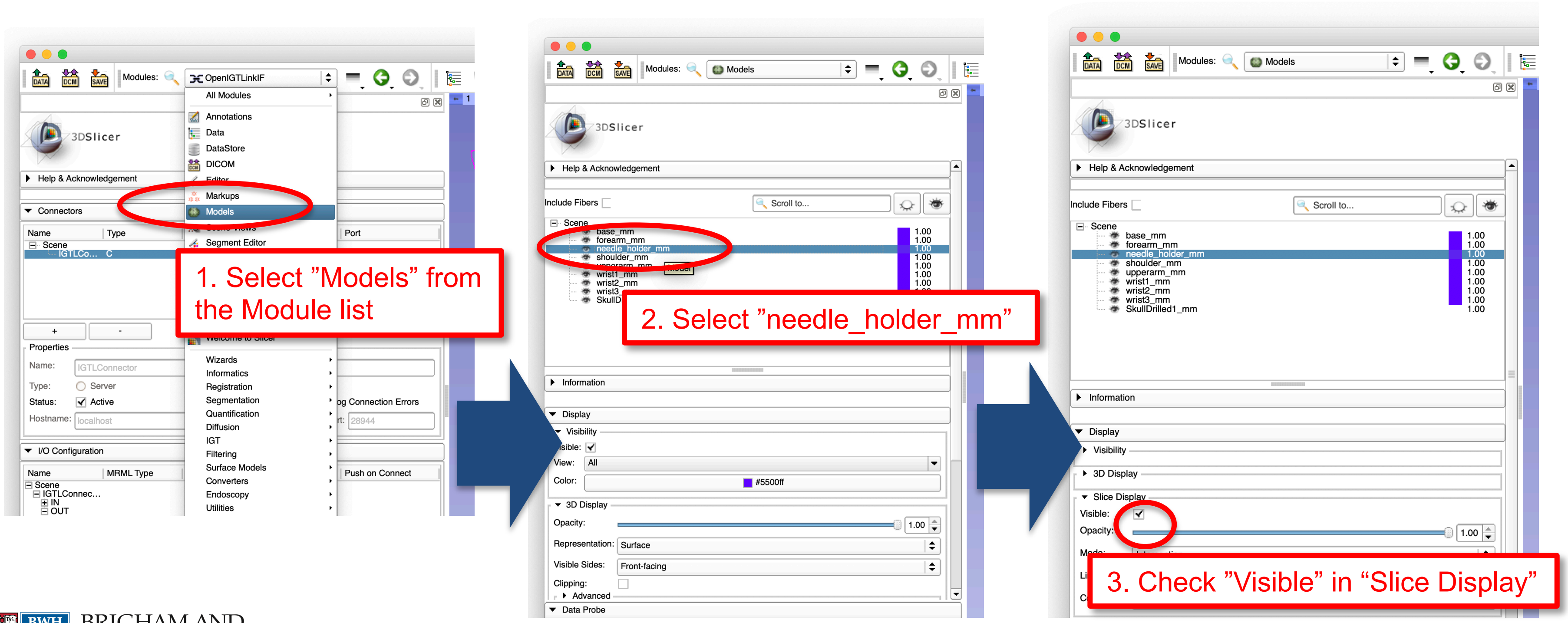
- Once the Plan is sent to ROS, the robot moves to the entry, and then the target.





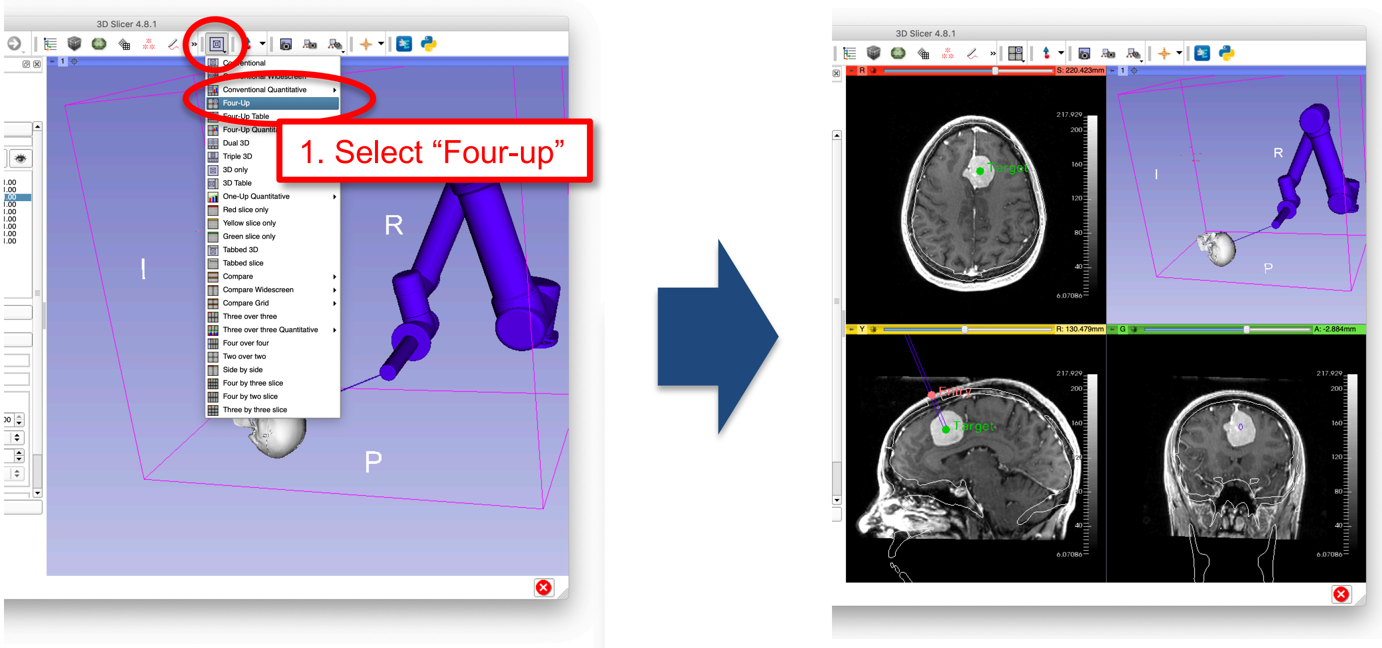
[Optional] Visualizing Trajectory on MRI

- Make the needle visible on MRI



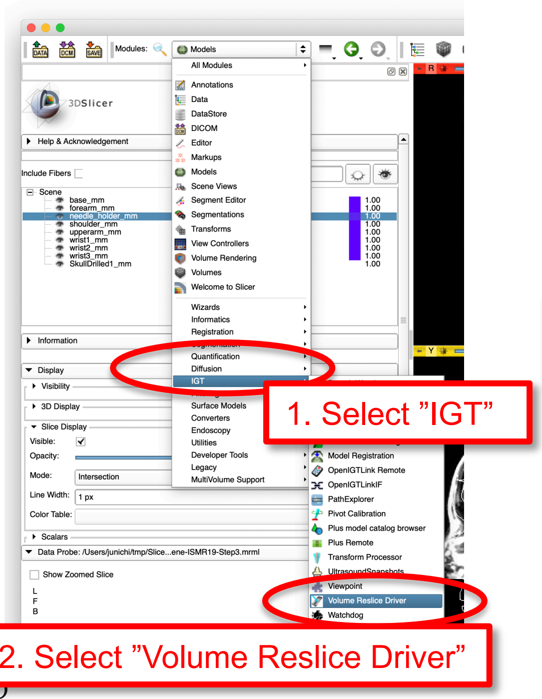
[Optional] Visualizing Trajectory on MRI

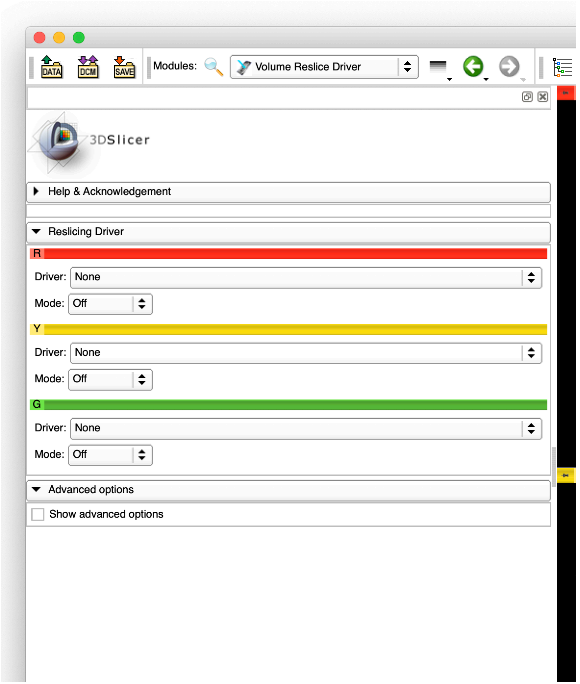
- Switch to ”Four-up” view.



[Optional] Visualizing Trajectory on MRI

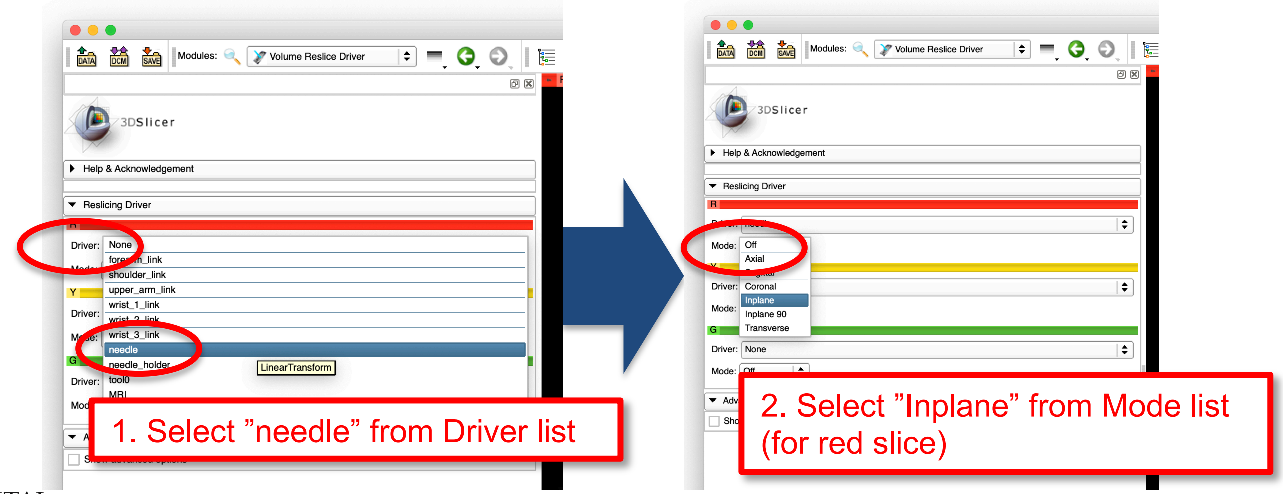
- Align slices with the needle using “Volume Reslice Driver”.





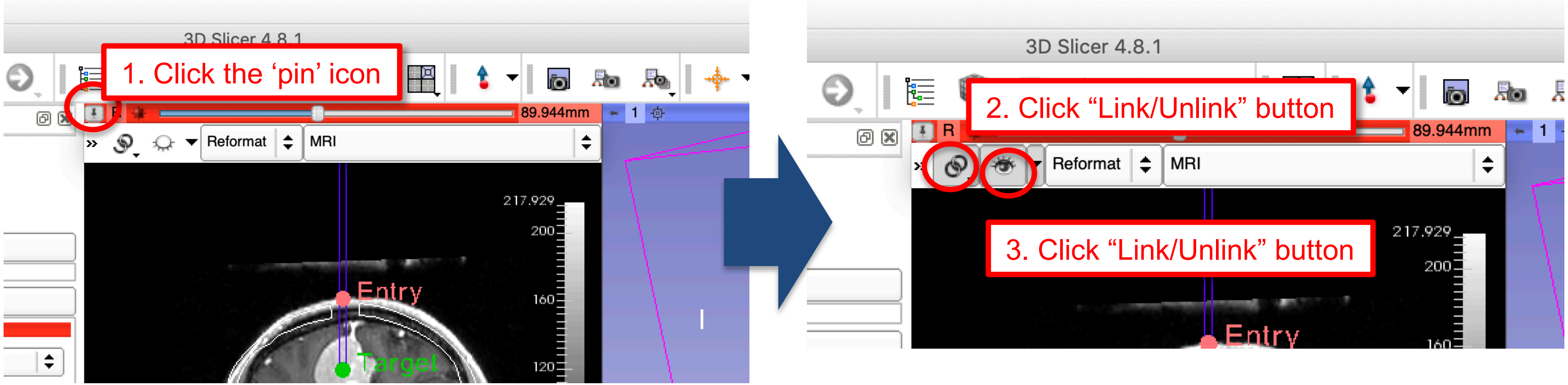
[Optional] Visualizing Trajectory on MRI

- Select:
  - R: Driver: ”needle”, Mode: “Inplane”
  - Y: Driver: “needle”, Mode: “Inplane 90”
  - G: Driver: “needle”, Mode” “Transverse”



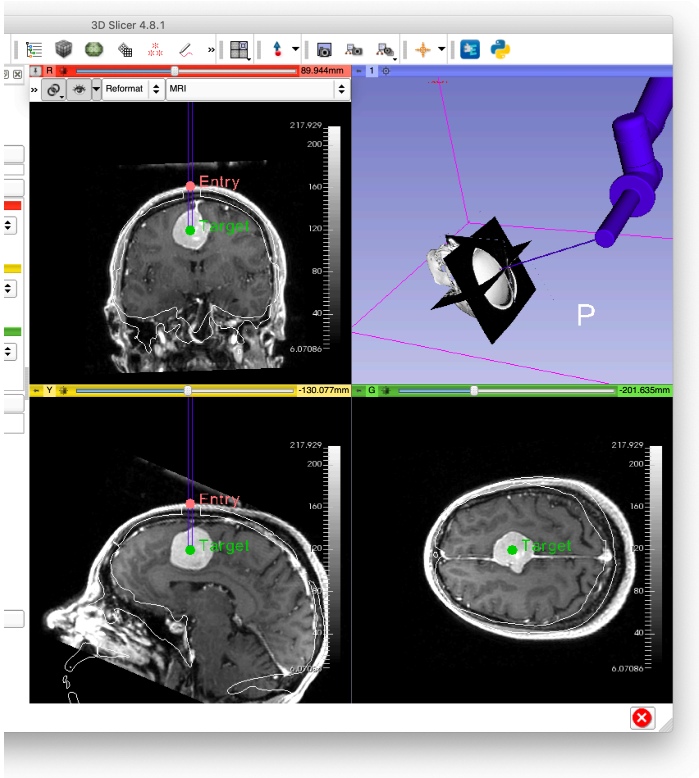
[Optional] Visualizing Trajectory on MRI

- Turn on slice visibility in 3D View



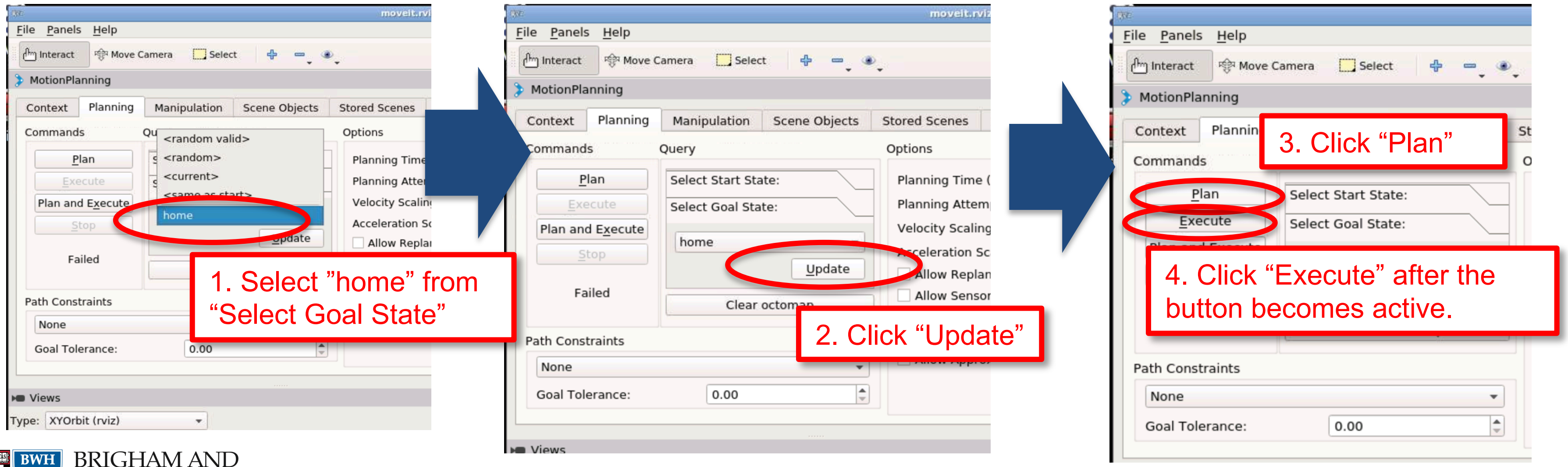
[Optional] Visualizing Trajectory on MRI

- Reformatted 2D slices along the needle is now visible.



[Optional] Visualizing Trajectory while Moving

- Terminate the custom control node by pressing Ctrl+c on the terminal.
- Move the robot to the home position.



[Optional] Visualizing Trajectory while Moving

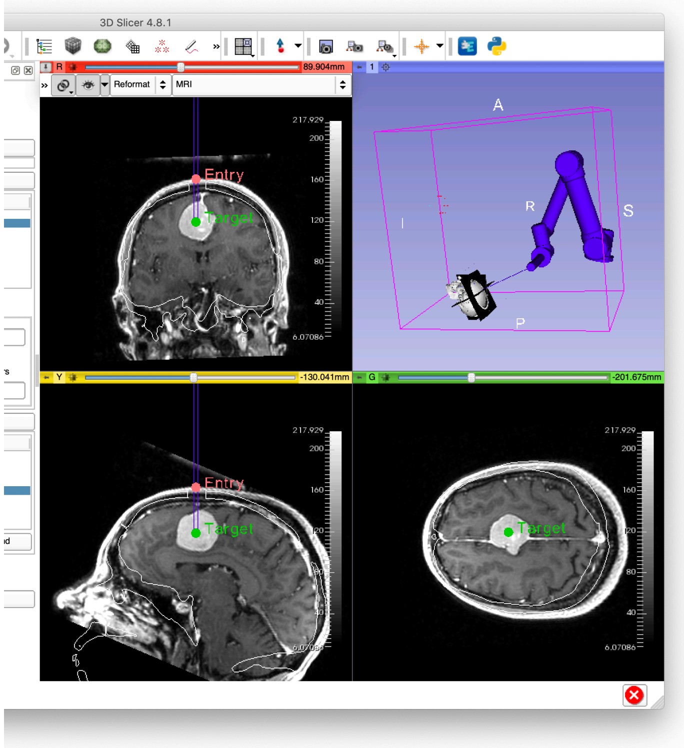
- Start the custom control node

```
$ rosrun ismr19_control ismr19_control_node
```

- Send the target (see page 68)

[Optional] Visualizing Trajectory while Moving

- The reformatted plane follows the needle.



References (Web)

- Tutorial Page: https://rosmed.github.io/
- 3D Slicer: http://www.slicer.org/
- ROS: http://www.ros.org/
- OpenIGTLink: http://openigtlink.org/
- ROS-IGTL-Bridge: https://github.com/openigtlink/ROS-IGTL-Bridge

References (Papers)

- Fedorov A, Beichel R, Kalpathy-Cramer J, Finet J, Fillion-Robin JC, Pujol S, Bauer C, Jennings D, Fennessy F, Sonka M, Buatti J, Aylward S, Miller JV, Pieper S, Kikinis R. 3D Slicer as an image computing platform for the Quantitative Imaging Network. Magn Reson Imaging. 2012 Nov;30(9):1323-41. doi: 10.1016/j.mri.2012.05.001. Epub 2012 Jul 6. PubMed PMID: 22770690; PubMed Central PMCID: PMC3466397.
- Frank T, Krieger A, Leonard S, Patel NA, Tokuda J. ROS-IGTL-Bridge: an open network interface for image-guided therapy using the ROS environment. Int J Comput Assist Radiol Surg. 2017 Aug;12(8):1451-1460. doi: 10.1007/s11548-017-1618-1. Epub 2017 May 31. PubMed PMID: 28567563; PubMed Central PMCID: PMC5543207.
- Tauscher S, Tokuda J, Schreiber G, Neff T, Hata N, Ortmaier T. OpenIGTLink interface for state control and visualisation of a robot for image-guided therapy systems. Int J Comput Assist Radiol Surg. 2015 Mar;10(3):285-92. doi: 10.1007/s11548-014-1081-1. Epub 2014 Jun 13. PubMed PMID: 24923473; PubMed Central PMCID: PMC4265315.
- Tokuda J, Fischer GS, Papademetris X, Yaniv Z, Ibanez L, Cheng P, Liu H, Blevins J, Arata J, Golby AJ, Kapur T, Pieper S, Burdette EC, Fichtinger G, Tempany CM, Hata N. OpenIGTLink: an open network protocol for image-guided therapy environment. Int J Med Robot. 2009 Dec;5(4):423-34. doi: 10.1002/rcs.274. PubMed PMID: 19621334; PubMed Central PMCID: PMC2811069.
- Ungi T, Lasso A, Fichtinger G. Open-source platforms for navigated image-guided interventions. Med Image Anal. 2016 Oct;33:181-186. doi: 10.1016/j.media.2016.06.011. Epub 2016 Jun 15. Review. PubMed PMID: 27344106.

Contact

- If you have any questions or feedback regarding this tutorial, please contact:

Junichi Tokuda, Ph.D.

Department of Radiology

Brigham and Women's Hospital / Harvard Medical School

tokuda @ bwh.harvard.edu