Tutorial para desarrolladores de Slicer: Programación en Slicer

Sonia Pujol, Ph.D.
Profesora adjunta de radiología
Directora de formación y educación en 3D Slicer
Hospital de mujeres de Brigham
Facultad de Medicina de Harvard

Steve Pieper, Ph.D.
Arquitecto jefe de 3D Slicer
Isomics Inc.

Objetivo del tutorial


```
def threshold(t):
    n=getNode('T2')
    a=array('T2')
    a[a<t]=0
    arrayFromVolumeModified(n)
    print('Thresholding done')
```


```
b=qt.QPushButton('Toggle')
b.connect('clicked()',toggle)
b.styleSheet = "font-size: 24pt; color:
aqua; margin: 20px"
b.show()
```

Este tutorial es una introducción a la consola de Python y al widget Qt de la versión 5 de 3D Slicer

Esquema del Tutorial

Parte 1: Visión general de los módulos de 3D Slicer

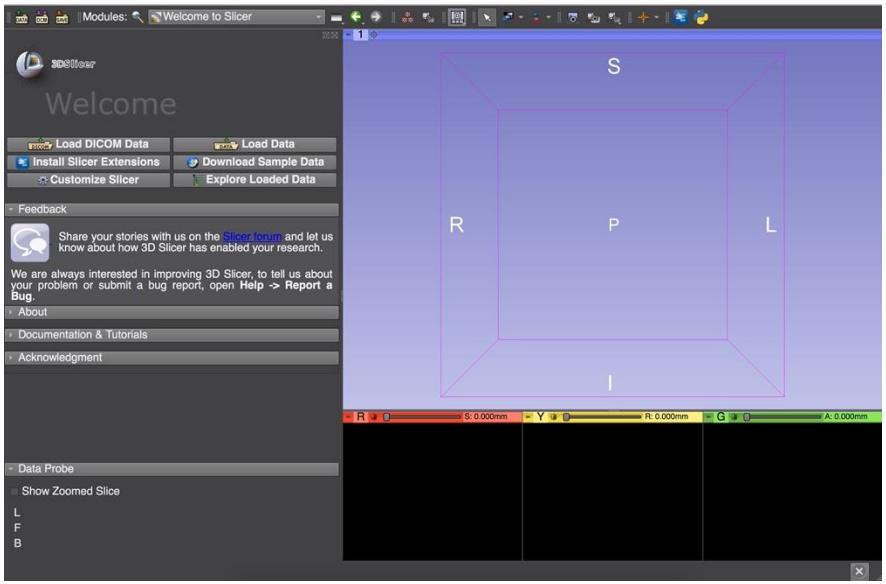
Parte 2: Familiarizarse con el entorno Python en 3D Slicer

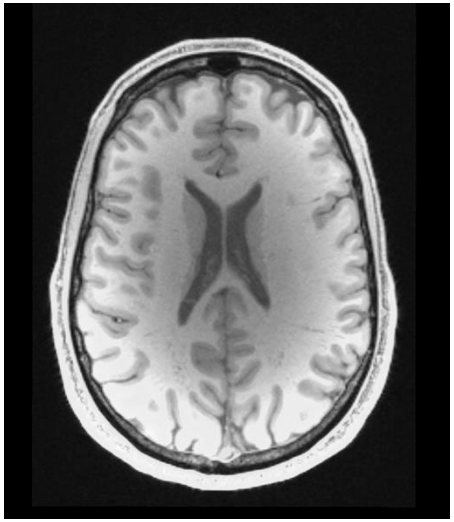
Parte 3: Familiarizarse con el conjunto de herramientas de widgets Qt en 3D Slicer

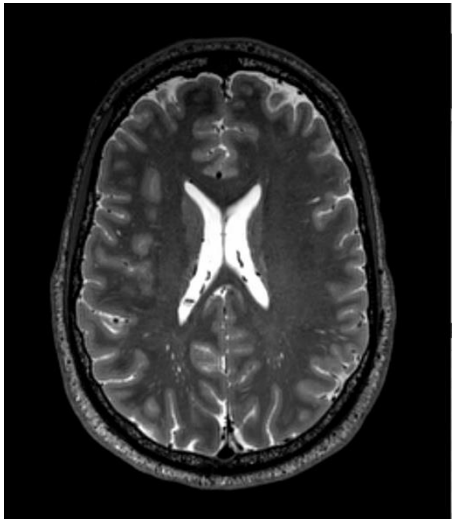
Descargo de responsabilidad

- 3D Slicer es una aplicación de software libre de código abierto distribuida bajo una licencia de estilo BSD.
- El software no cuenta con la aprobación de la FDA ni el marcado CE y es para uso exclusivo en investigación.

Materiales didácticos





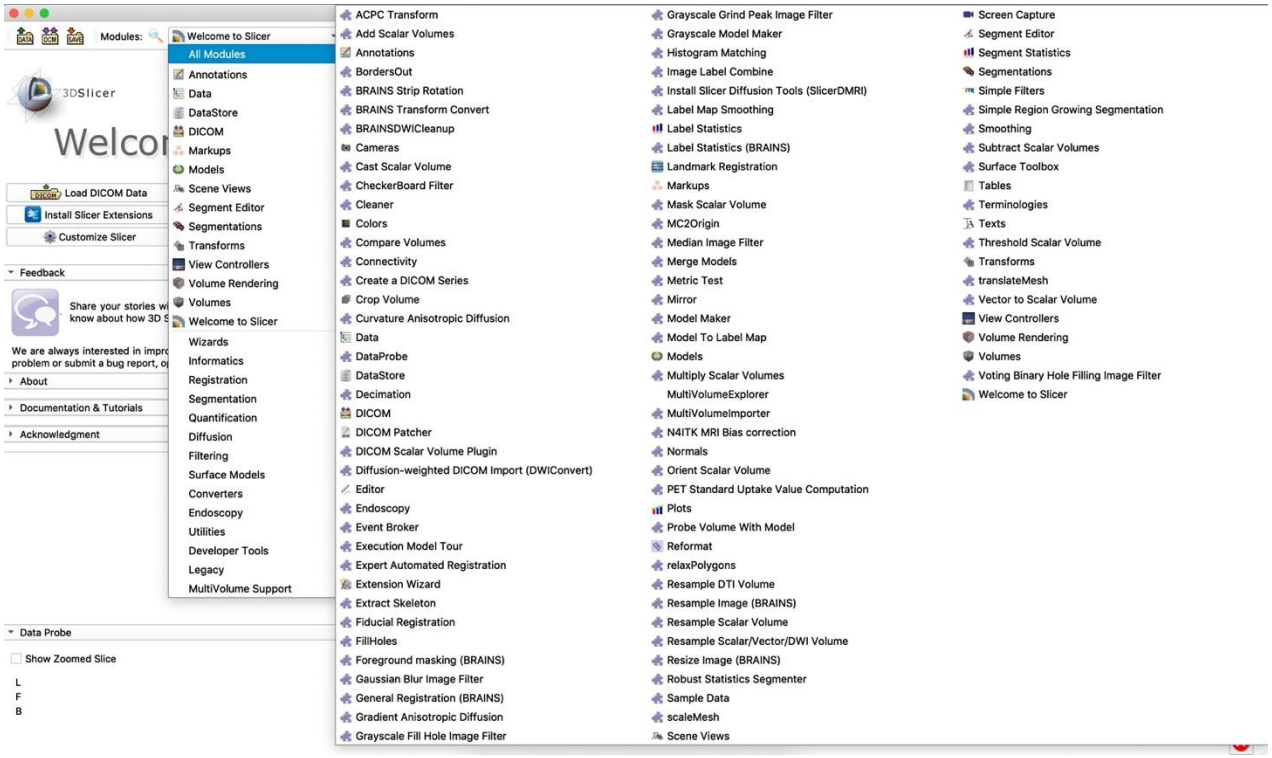


SlicerProgrammingTutorialData.zip
link

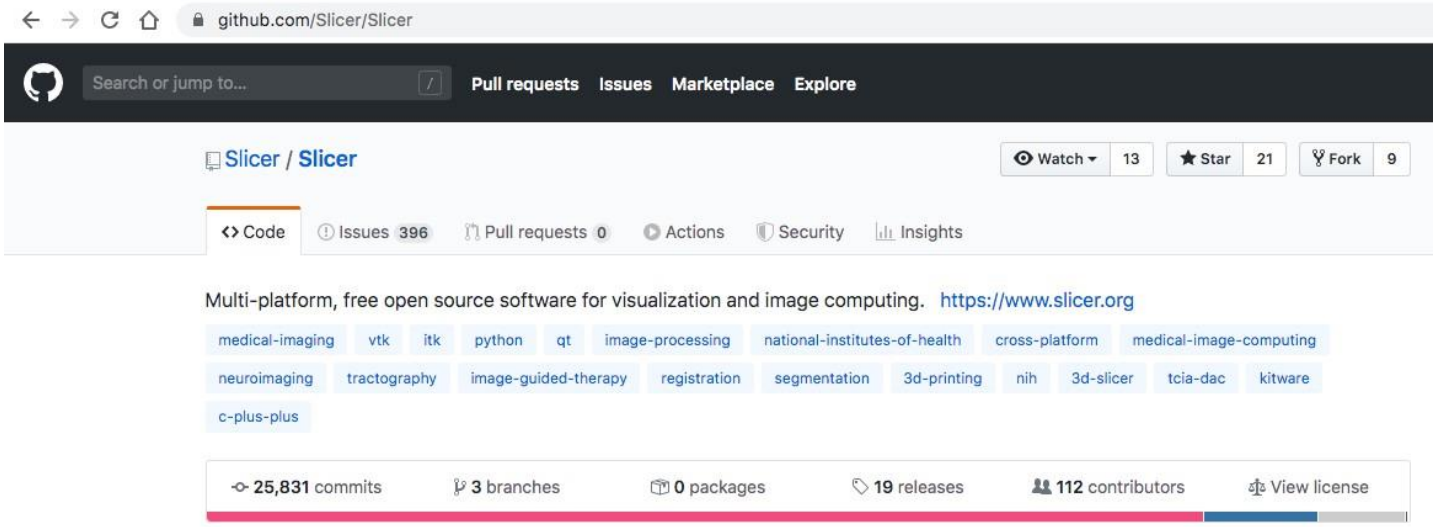
3D Slicer versión 4.11

Parte 1

Visión general de los módulos de Slicer



3D Slicer



- 3D Slicer es una plataforma de código abierto para el análisis y la visualización de datos de imágenes médicas.
- 3D Slicer se compila y se prueba a diario en plataformas Windows,Mac y Linux
- El código fuente está disponible gratuitamente enGitHub en
- http://github.com/Slicer/Slicer

Módulos Slicer

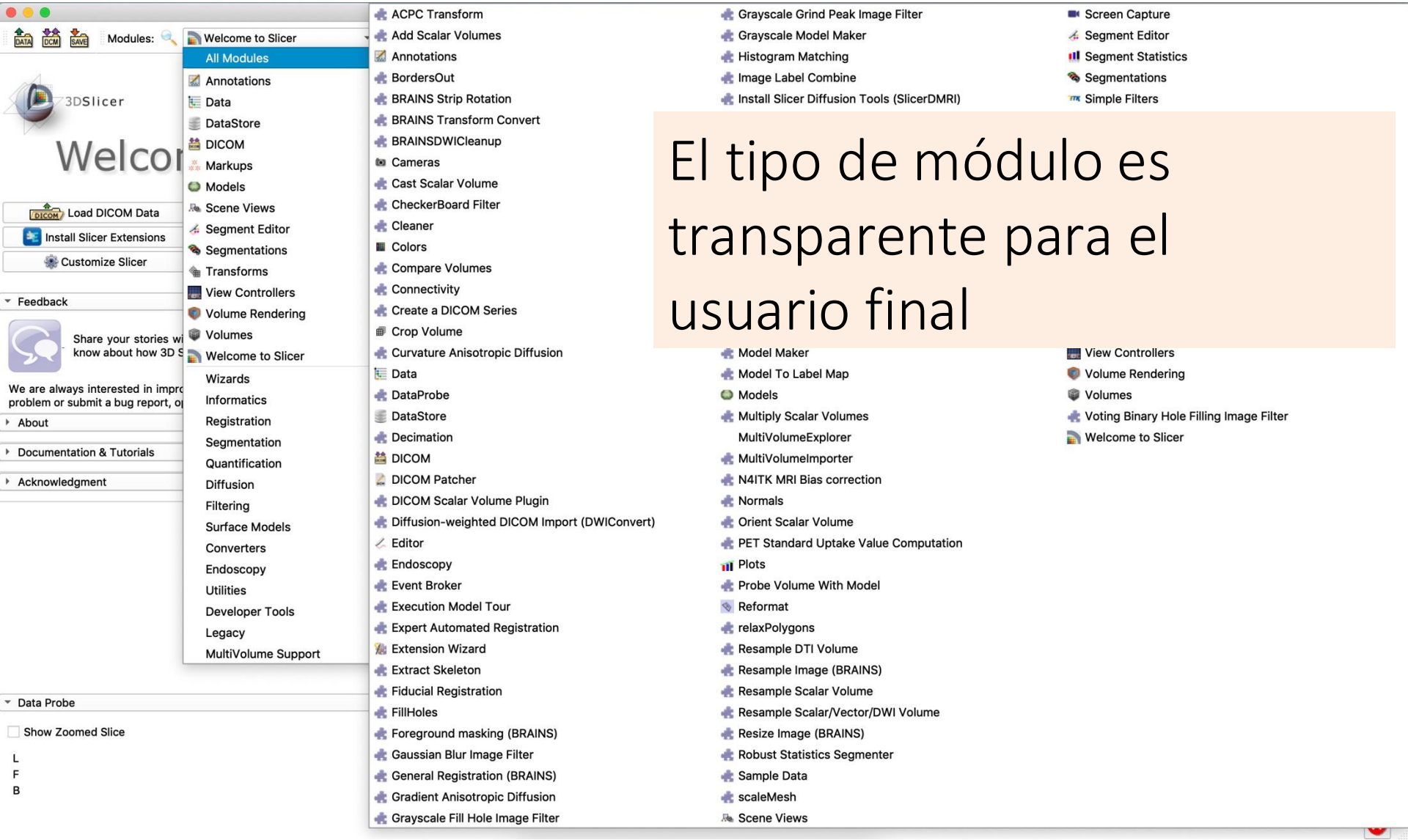
3D Slicer admite tres tipos de módulos:

- Interfaz de línea de comandos (CLI): ejecutor independiente con argumentos de entrada/salida limitados.
- Módulos cargables (C++ Plugins): optimizados para cálculos pesados.

Es el enfoque de este tutorial

- Módulos con script (Python): recomendados para la creación rápida de prototipos y el desarrollo de flujo de trabajo.

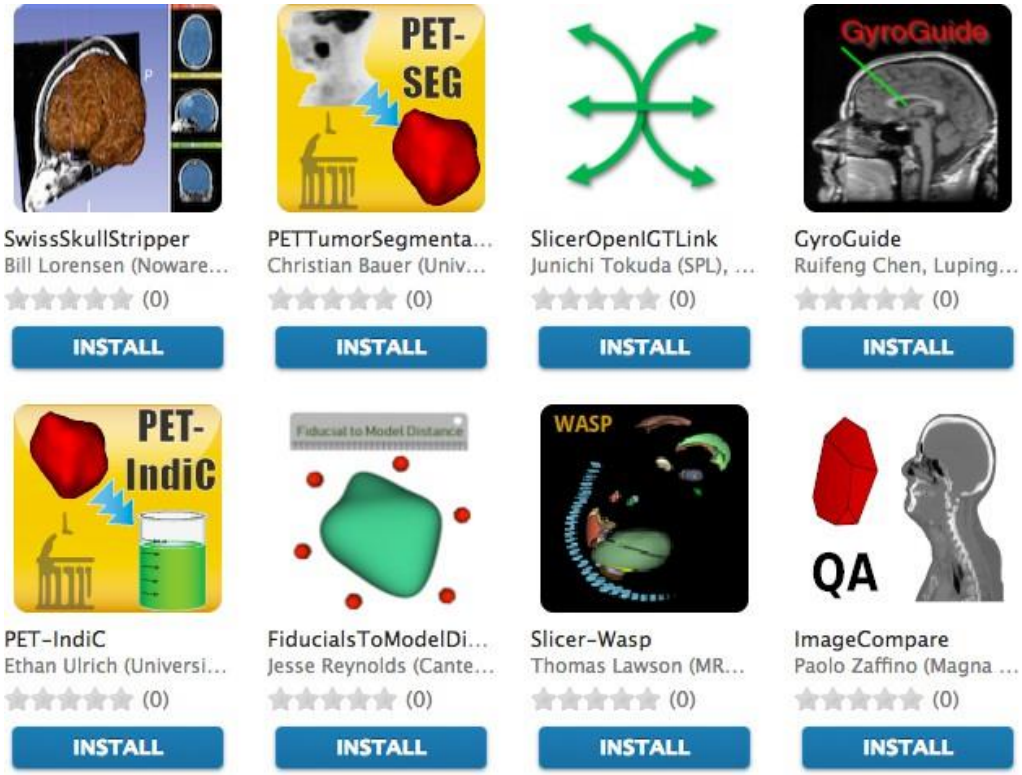
Módulos Slicer



El tipo de módulo es transparente para el usuario final

Extensiones Slicer

Una extensión Slicer es un paquete de entrega que incluye uno o varios módulos de Slicer

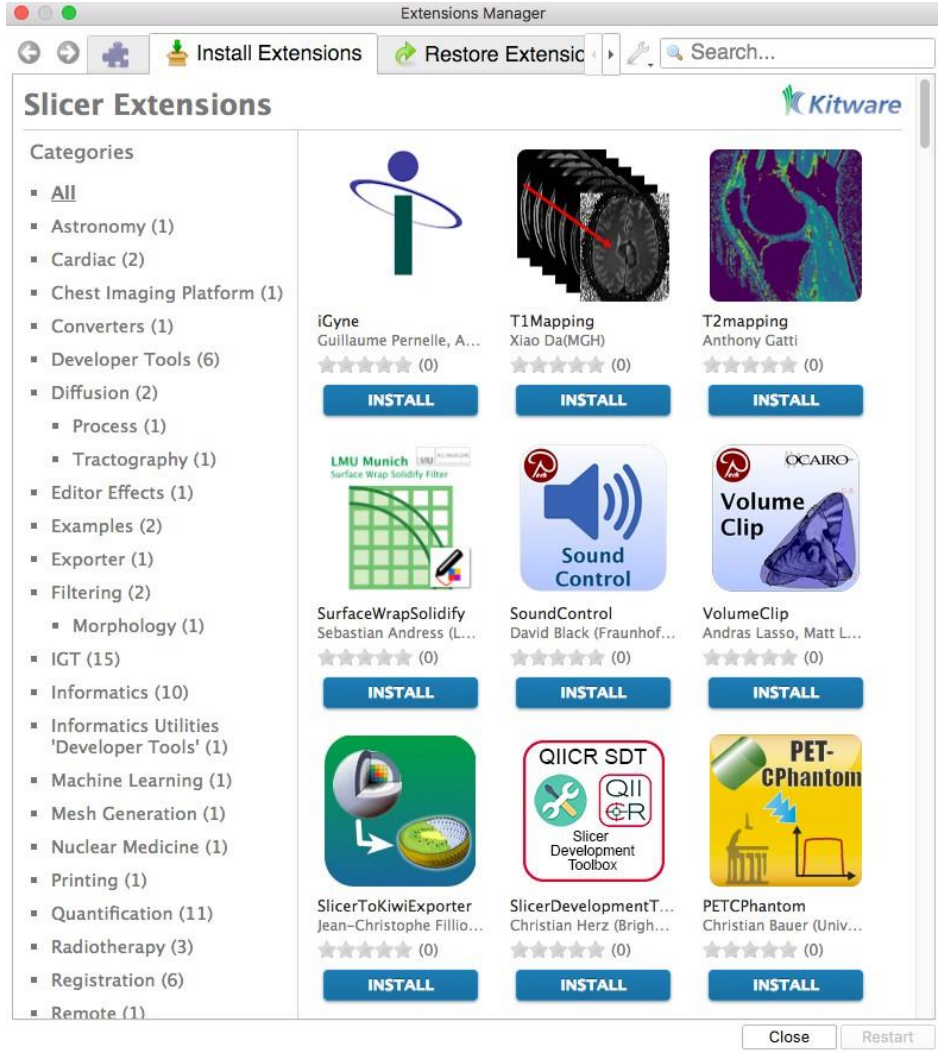


Administrador de extensiones de Slicer

- El administrador de extensiones de Slicer proporciona una plataforma de "tienda de aplicaciones" para el ecosistema de 3D Slicer

- El Extension Manager facilita la creación e instalación de extensiones de Slicer.

- La versión 5 de Slicer incluye más de 130 extensiones.



Parte 2

Familiarizarse con el entorno Python en 3D Slicer

Python en Slicer

Slicer v.4.11 funciona con Python3 y un conjunto de bibliotecas estándar

NumPy	NumPy, el cual es el paquete fundamental para la computación científica con Python.
VTK	VTK es una biblioteca de código abierto para la manipulación y visualización de datos científicos.
ITK	ITK es una biblioteca de código abierto para el análisis de imágenes.
TK	CTK es una biblioteca de código abierto para el cálculo de imágenes biomédicas.
PythonQt	PythonQT es un enlace de Python para Qt.
Qt	Qt es un entorno de multiplataforma utilizado como conjunto de herramientas gráficas.

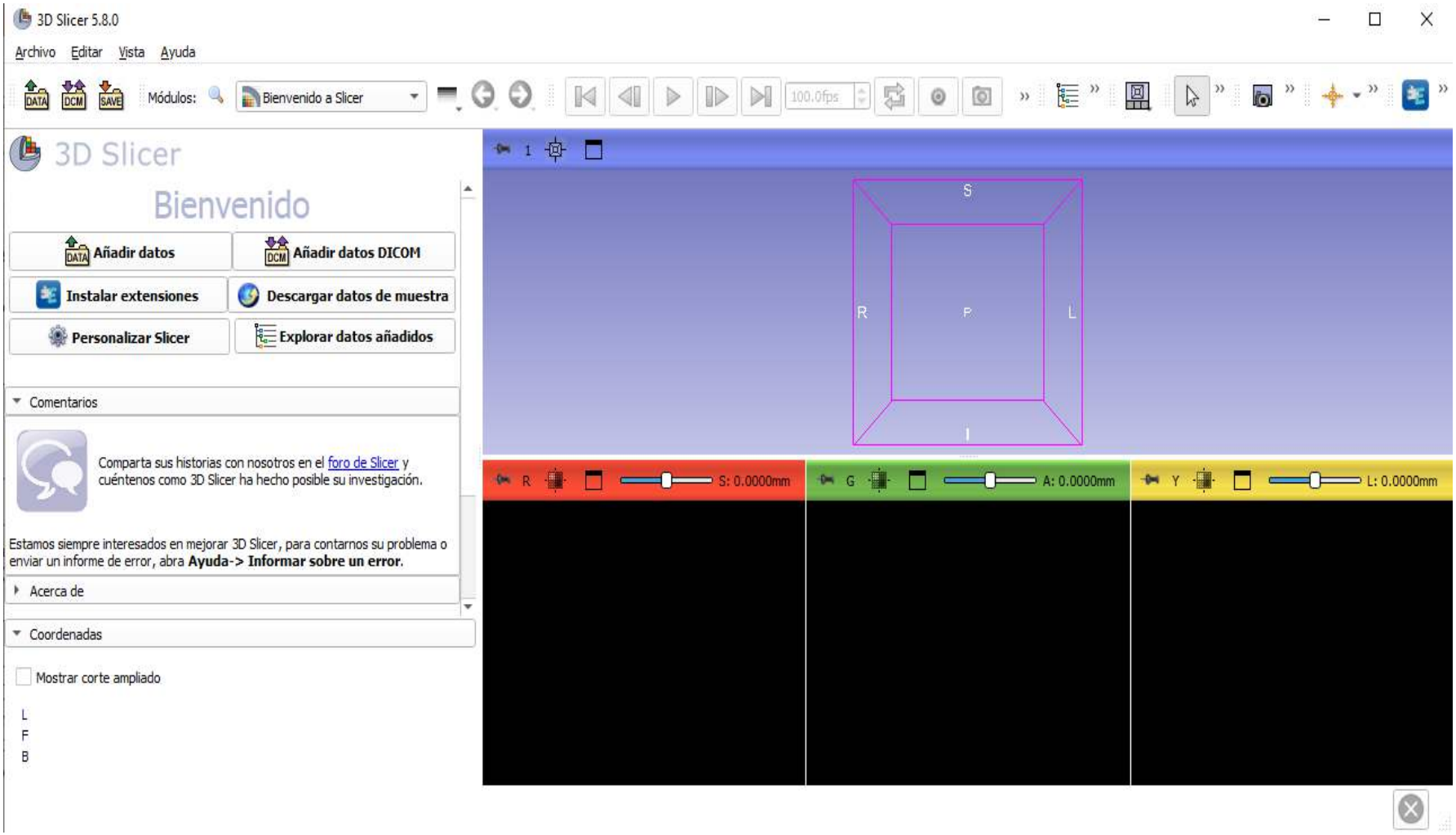
Python en Slicer

- El índice de paquetes de Python (PyPi) da acceso a más de 200.000 paquetes adicionales de Python (http://pipy.org).

- El comando “pip install” en Slicer permite a los desarrolladores instalar las herramientas de computación científica más comunes (por ejemplo, TensorFlow, SciPy, PyTorch, Pandas, etc.).

- Slicer puede utilizarse como núcleo de Jupyter notebook

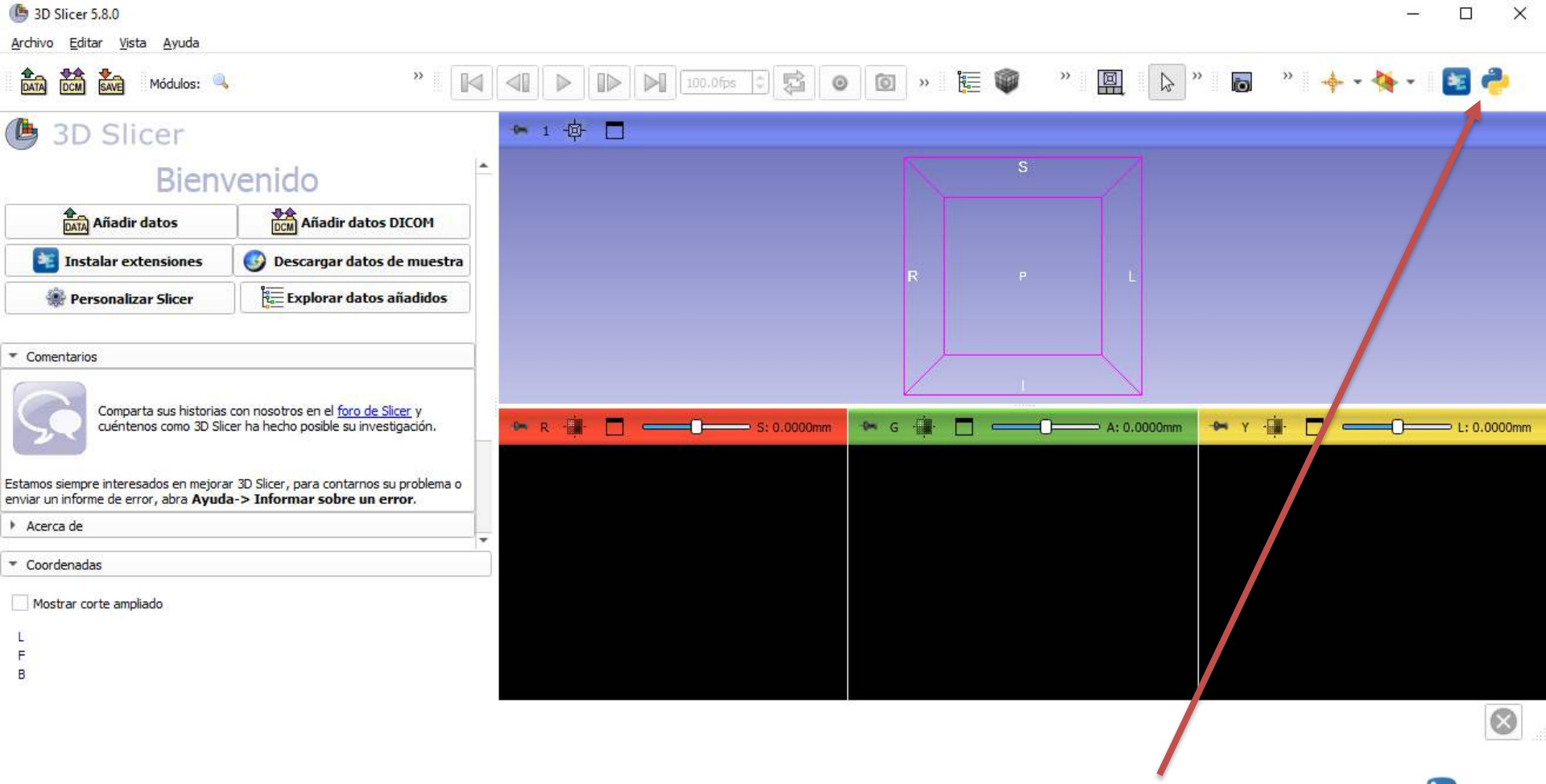
- PyCharm y otras herramientas de desarrollo de Python se pueden utilizar con Slicer



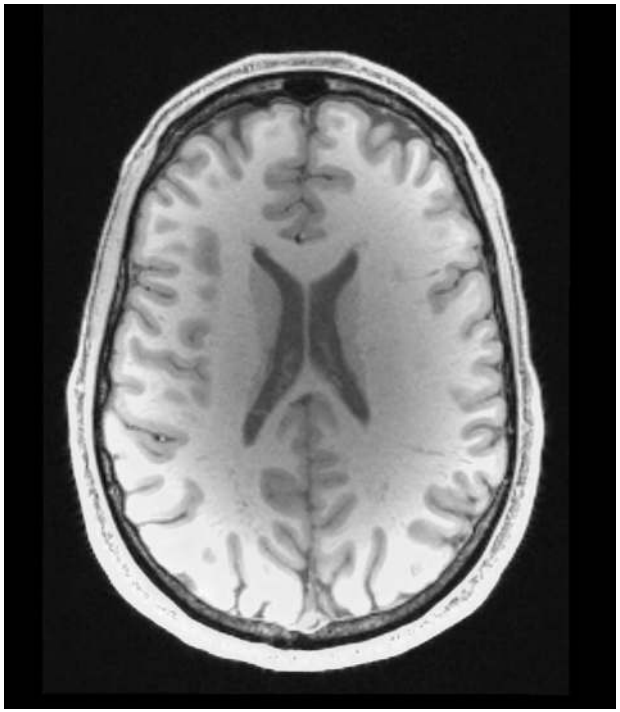
La version 5 de Slicer integra Python3, VTK5 and ITK5

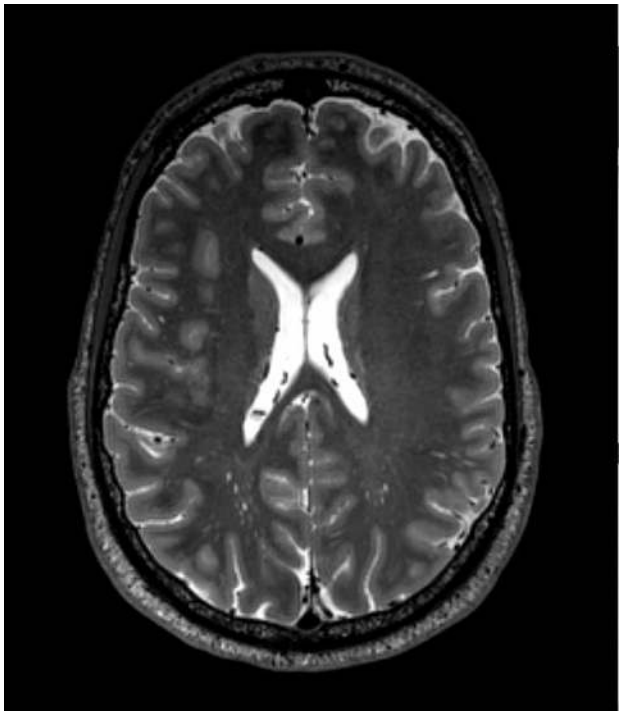
Consola de Python en Slicer

La consola de Python esta basada en Qt y permite un acceso directo a Slicer MRML Nodos, librerias (NumPy, VTK, ITK, CTK) y Qt.



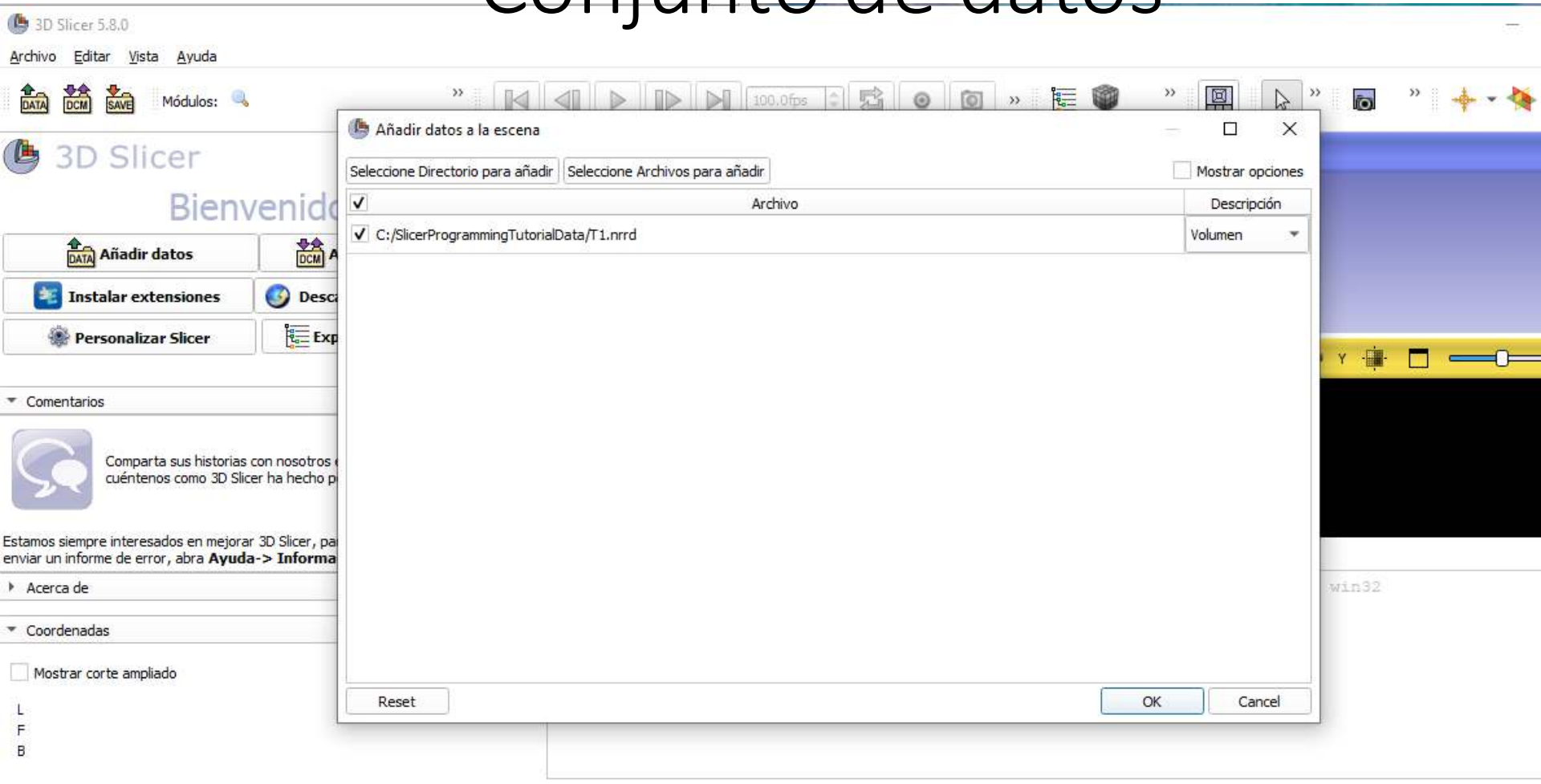
Para acceder a la Consola de Python, haga clic en el icono python en el menú de la barra superior de Slicer





El conjunto de datos del tutorial de programación Slicer incluye una resonancia magnética ponderada en T1 y otra en T2 de un sujeto sano.

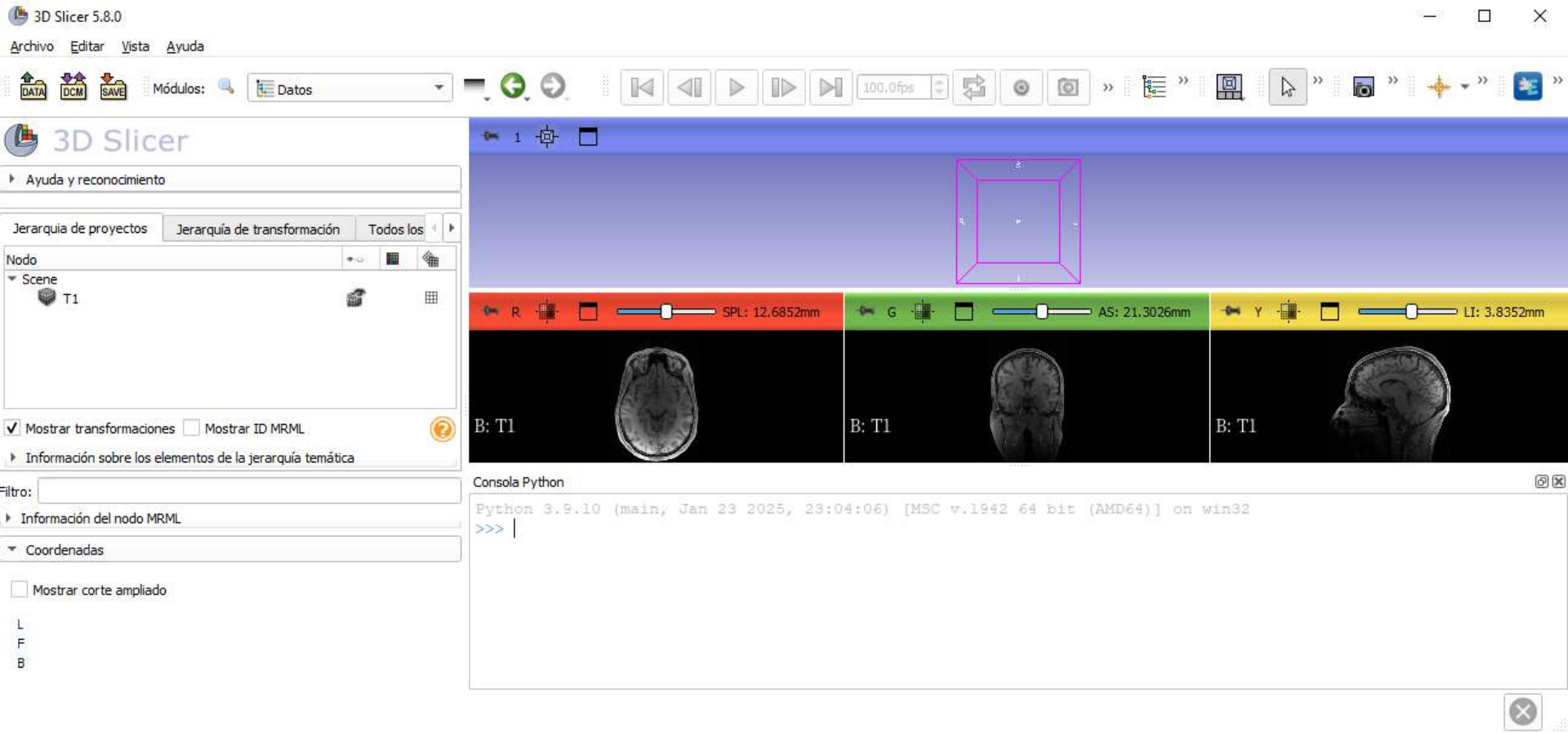
Conjunto de datos



Arrastre y suelte el archivo T1.nrrd

Haga clic en Aceptar para cargar el archivo en Slicer

Tutorial dataset



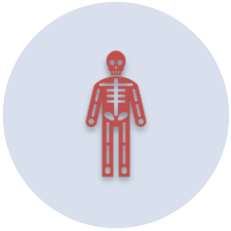
Panorama general

- Slicer es un software gratuito y de código abierto
- Hay miles de imágenes médicas sofisticadas disponibles en Internet que puede visualizar y analizar con 3D Slicer.

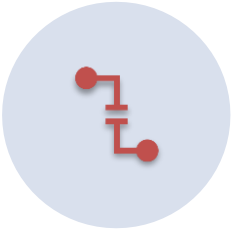
Modelo de datos de Slicer

El modelo de datos de Slicer se basa en la estructura de datos de escena de Slicer.

Una escena Slicer es una colección de imágenes, anotaciones, modelos 3D, transformaciones espaciales, fiduciales y cámaras.



El lenguaje MRML (Medical Reality Markup Language) es un lenguaje basado en XML que se utiliza para serializar el contenido de la escena Slicer en disco (scene.mrml).



Cada elemento de una escena se denomina nodo MRML

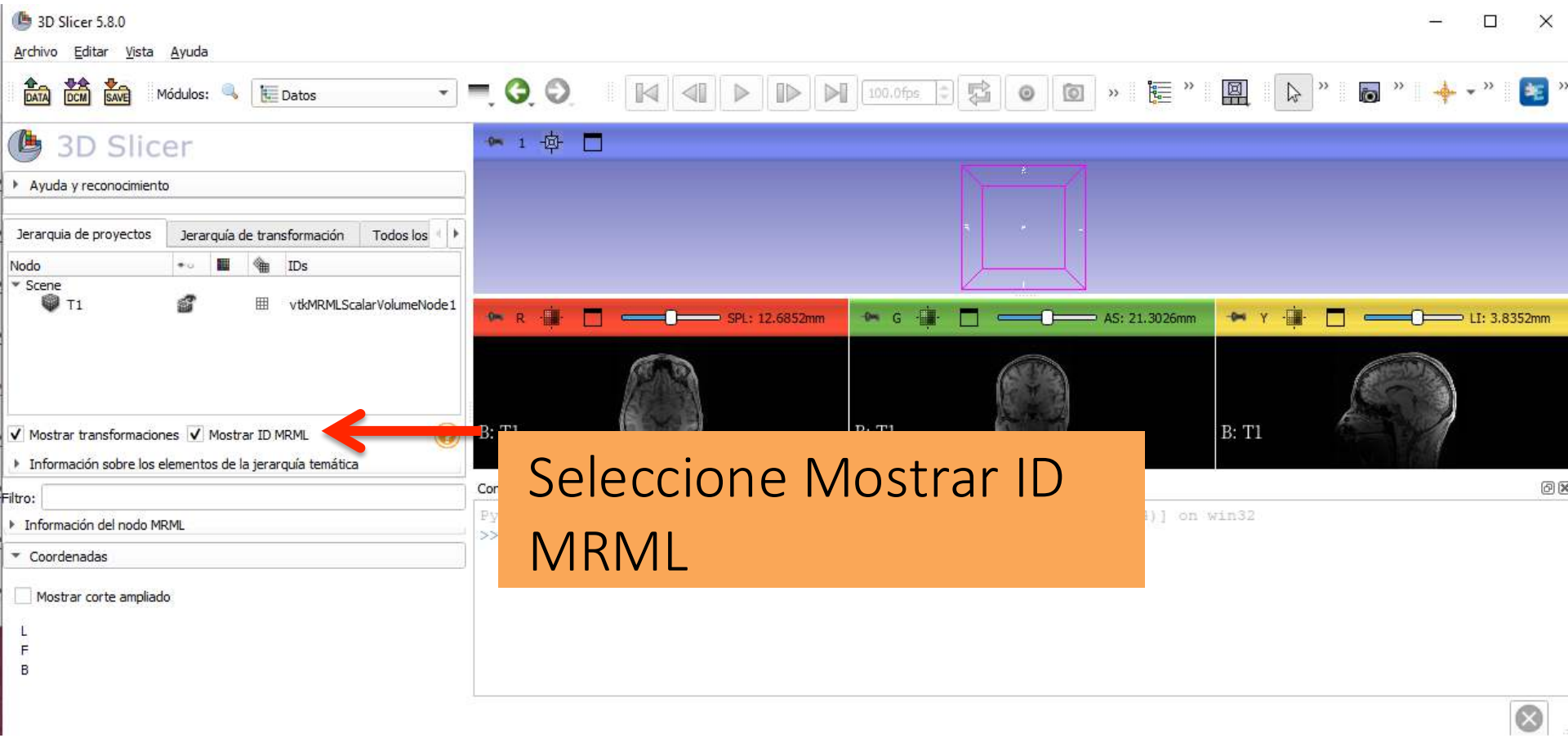
Nodos MRML de Slicer: Tipos Básicos

Nodo de datos: Almacena los datos en bruto.

Nodo de visualización: Describe cómo deben visualizarse los datos.

Nodo de almacenamiento: Describe cómo deben almacenarse los datos en el disco.

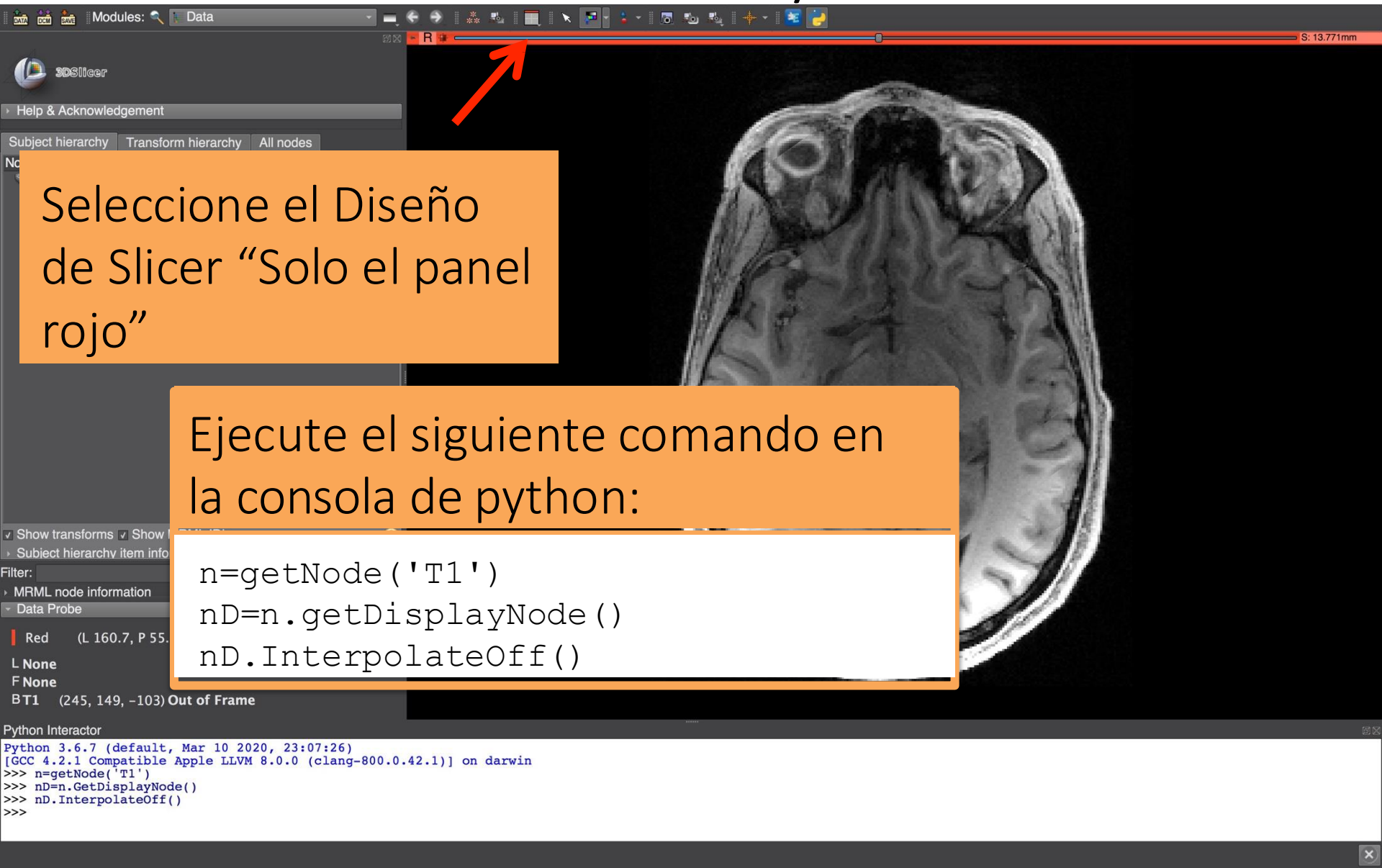
Conjunto de datos del tutorial



Slicer Data Model

Slicer muestra el identificador del nodo T1 vtkMRMLScalarVolumeNode1

Acceso a los nodos MRML desde la consola Python

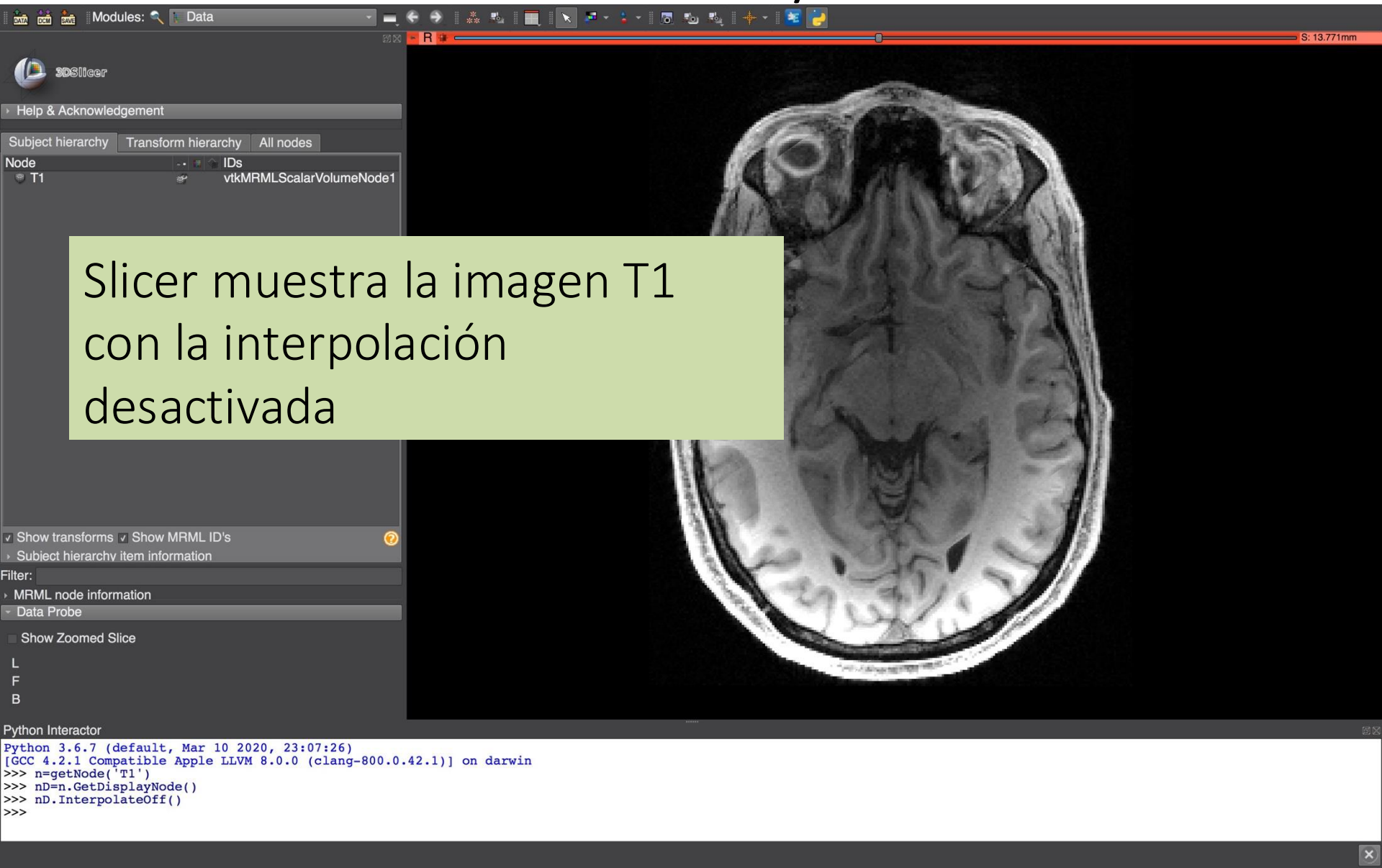


Seleccione el Diseño de Slicer “Solo el panel rojo”

Ejecute el siguiente comando en la consola de python:

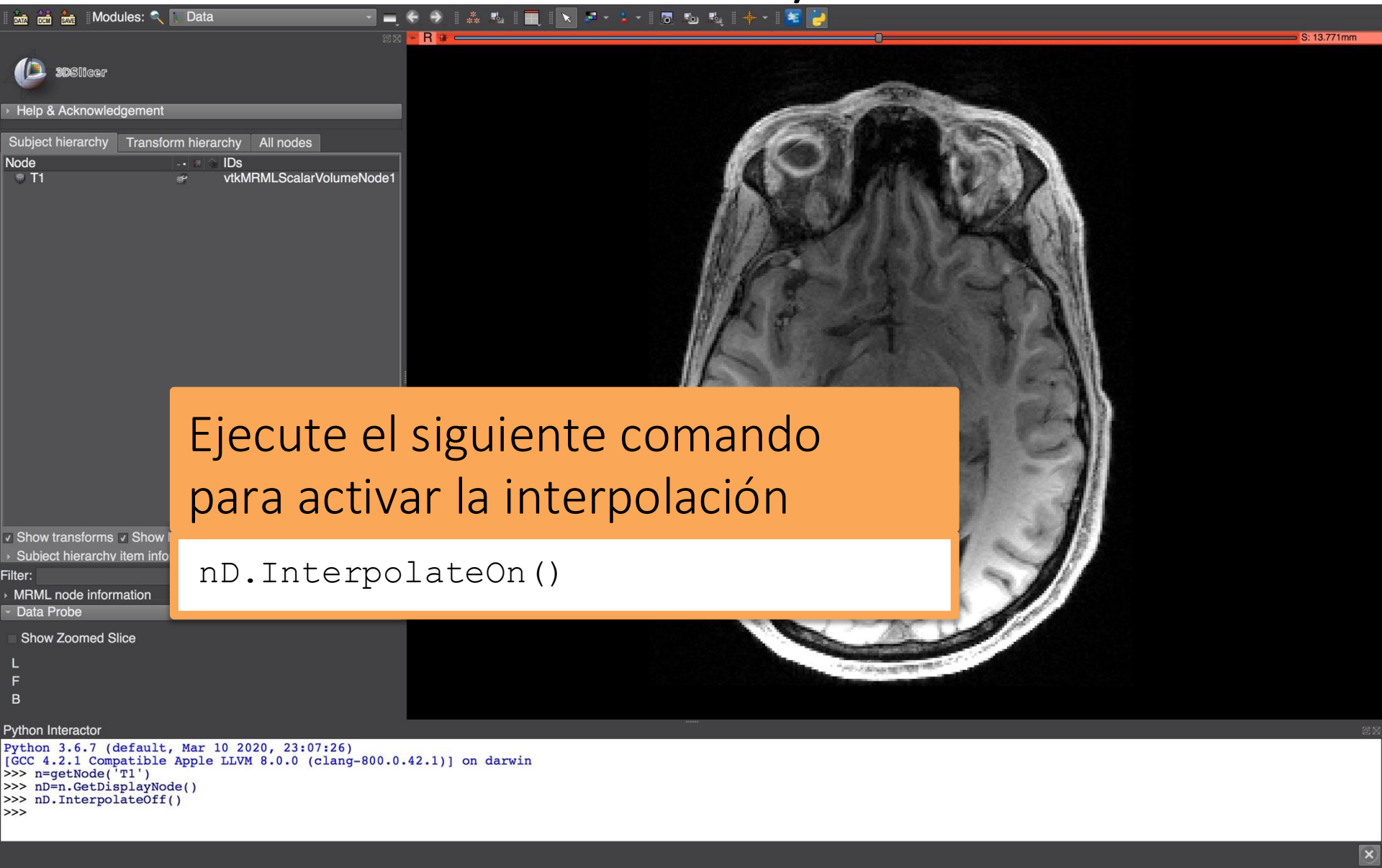
```
n=getNode('T1')
nD=n.getDisplayNode()
nD.InterpolateOff()
```

Acceso a los nodos MRML desde la consola Python



Slicer muestra la imagen T1 con la interpolación desactivada

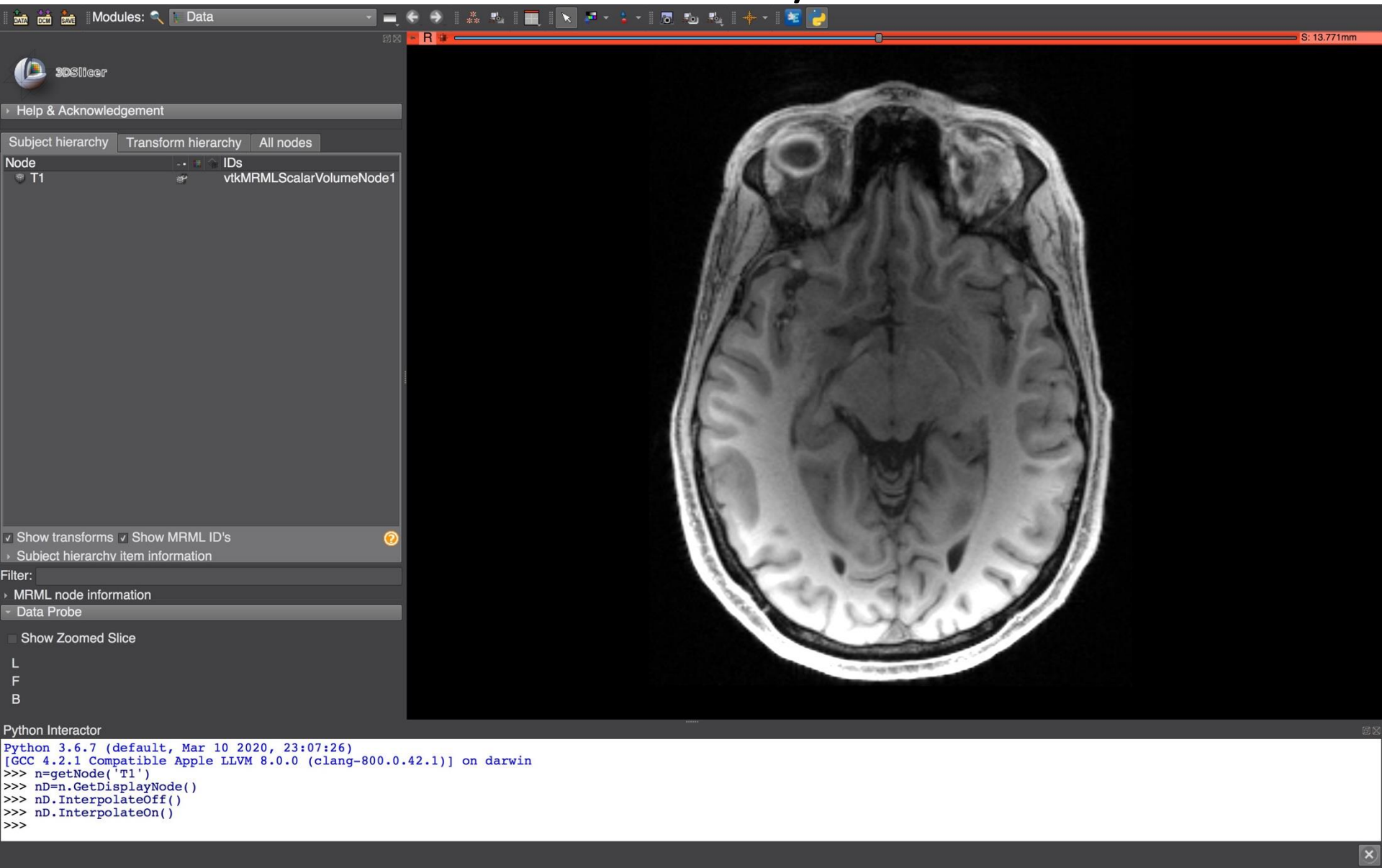
Acceso a los nodos MRML desde la consola Python



Ejecute el siguiente comando para activar la interpolación

```
nD.InterpolateOn()
```

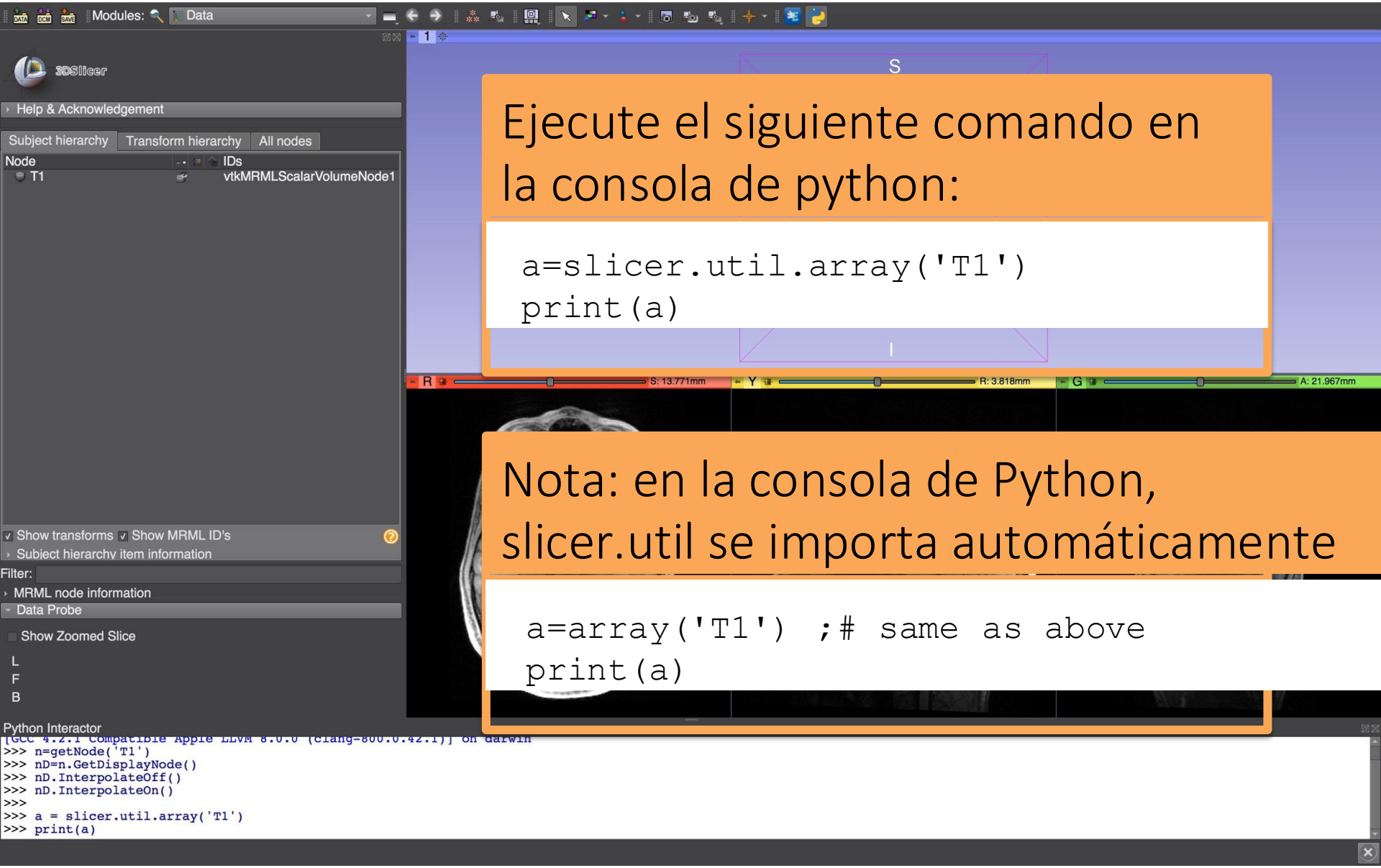
Acceso a los nodos MRML desde la consola Python



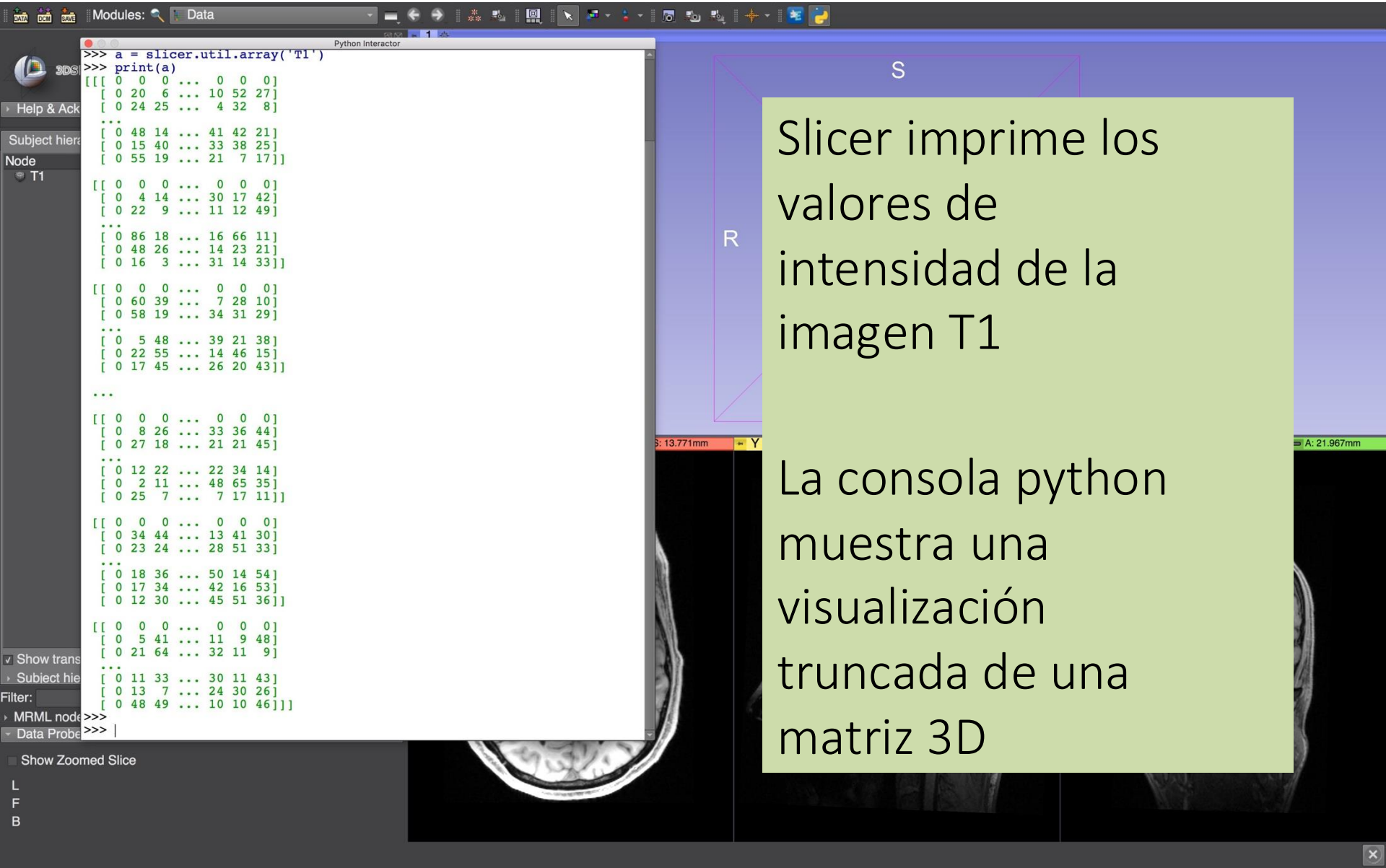
Acceso a los vóxeles de un volumen

- El paquete `slicer.util` da acceso a los volúmenes como matrices multidimensionales NumPy
- Los volúmenes pueden modificarse utilizando métodos estándar de NumPy

Acceso a los vóxeles de un volumen



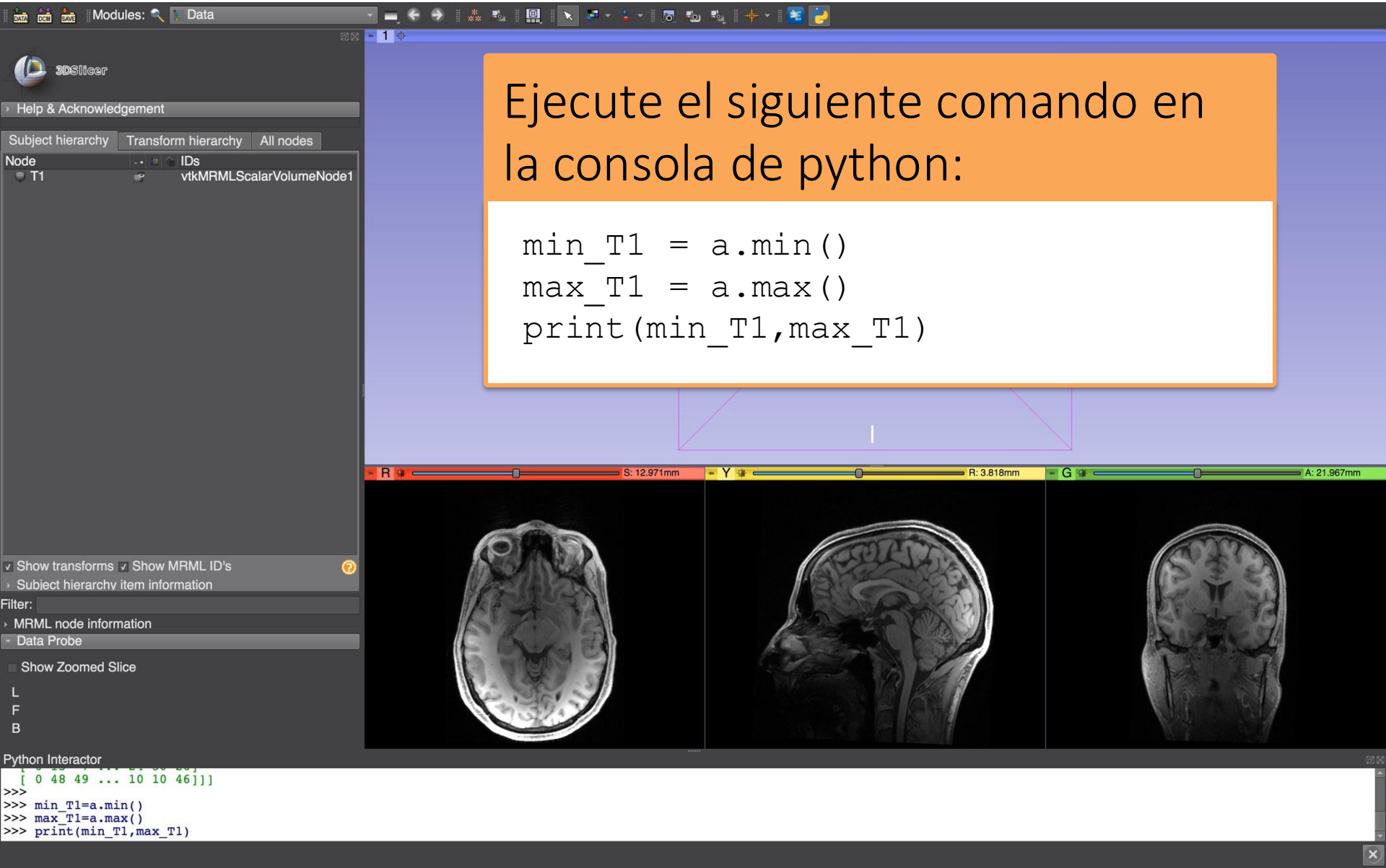
Acceso a los vóxeles de un volumen



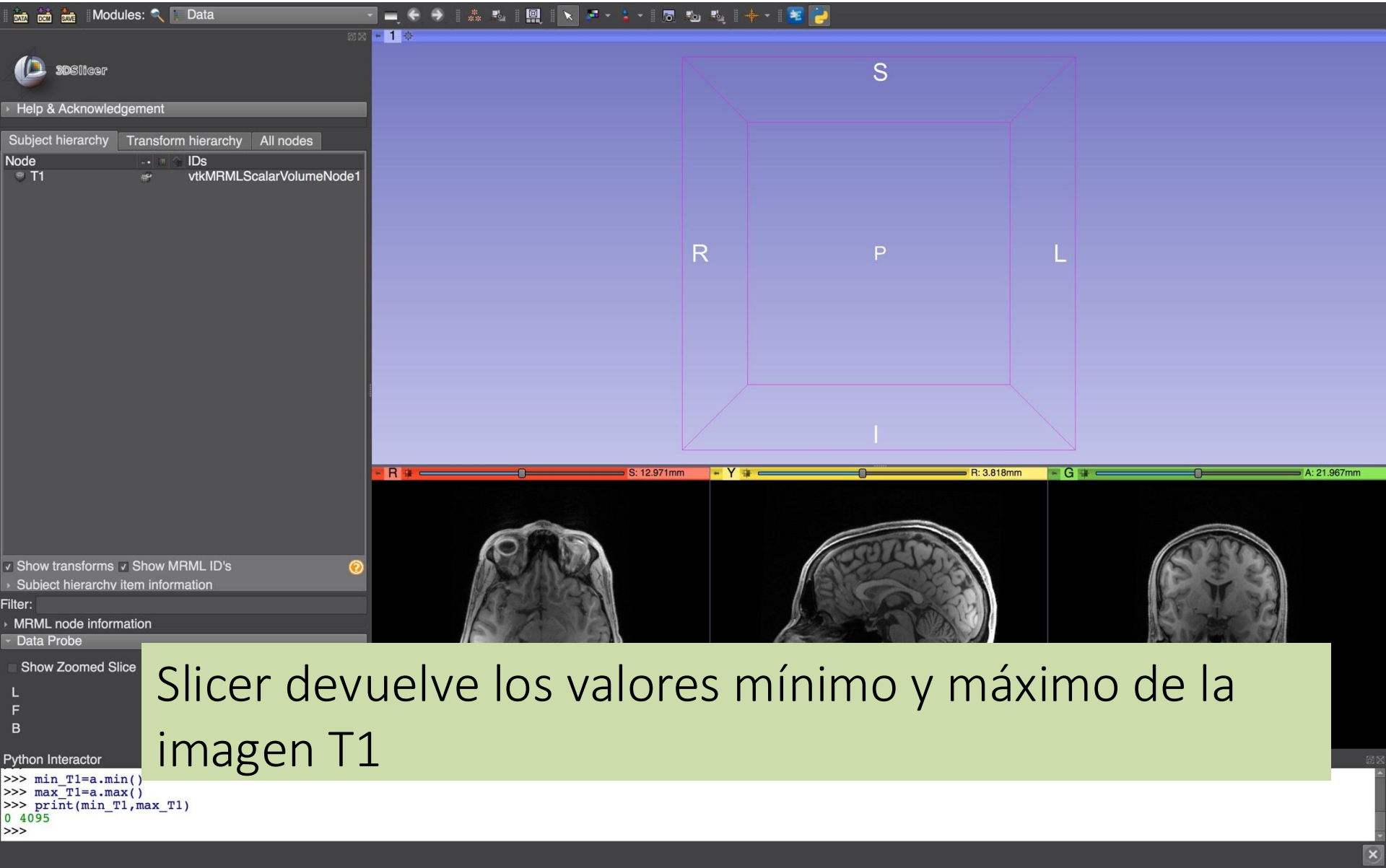
Slicer imprime los valores de intensidad de la imagen T1

La consola python muestra una visualización truncada de una matriz 3D

Acceso a los vóxeles de un volumen

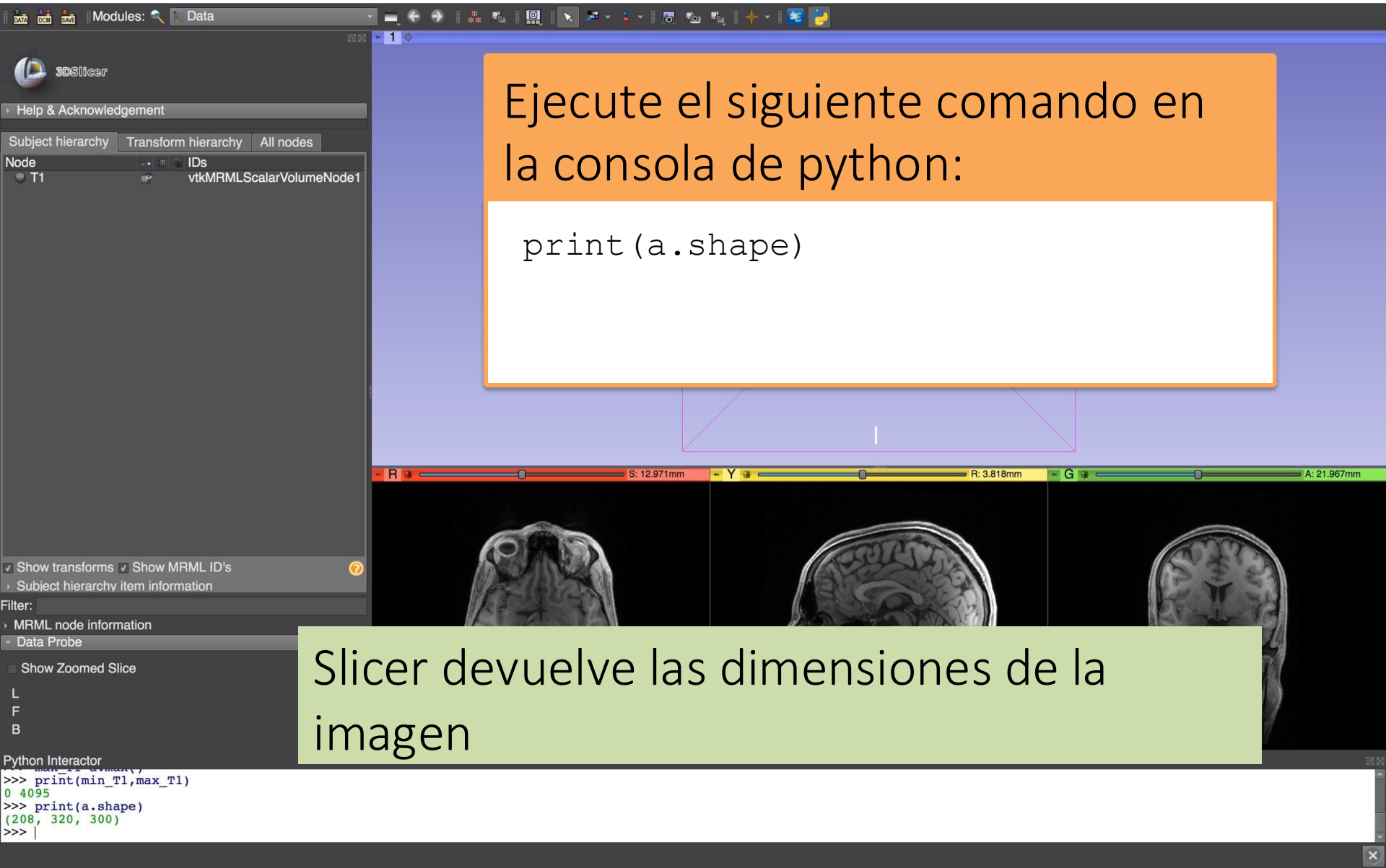


Acceso a los vóxeles de un volumen

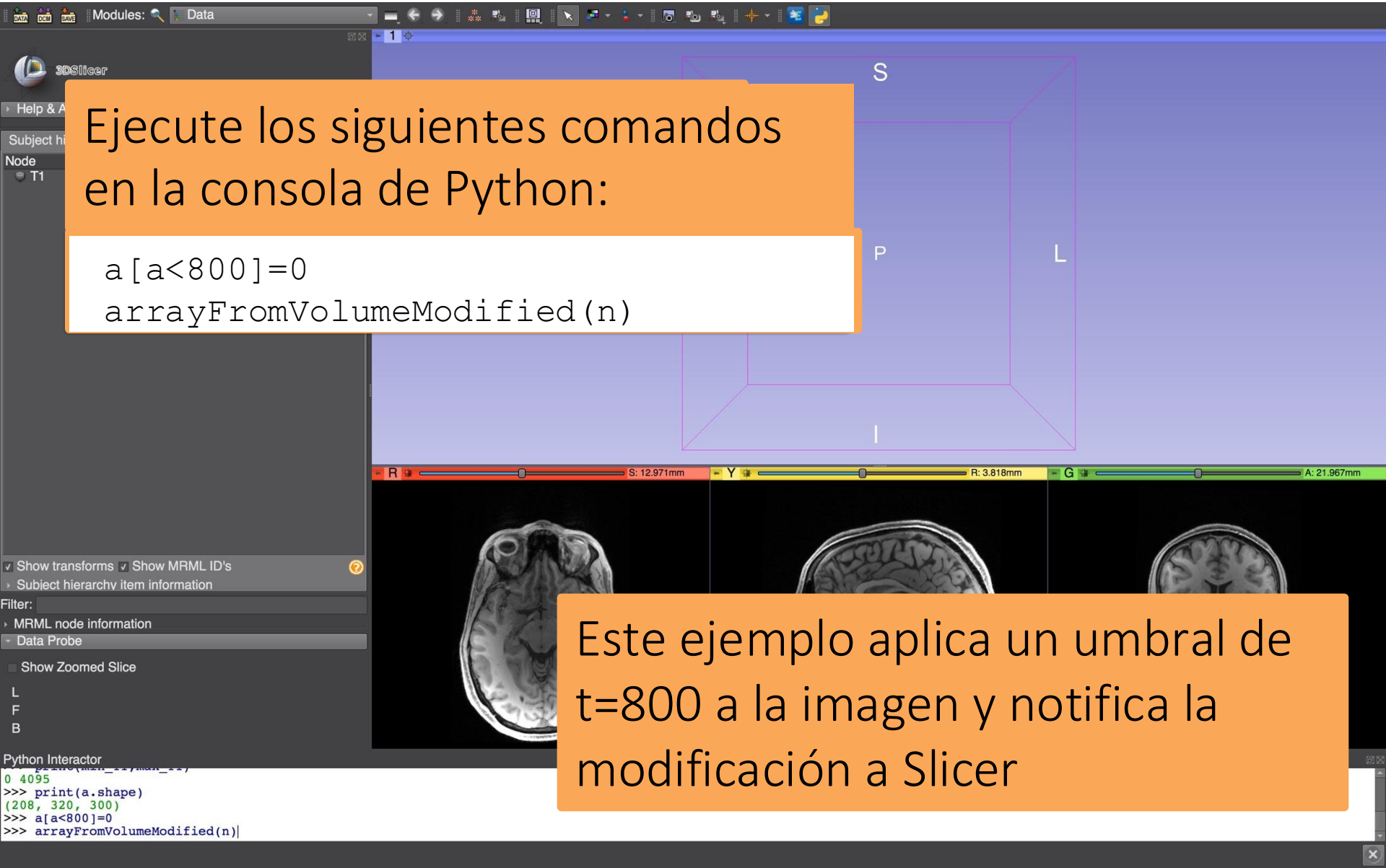


Slicer devuelve los valores mínimo y máximo de la imagen T1

Modificación de vóxeles en un volumen



Modificación de vóxeles en un volumen

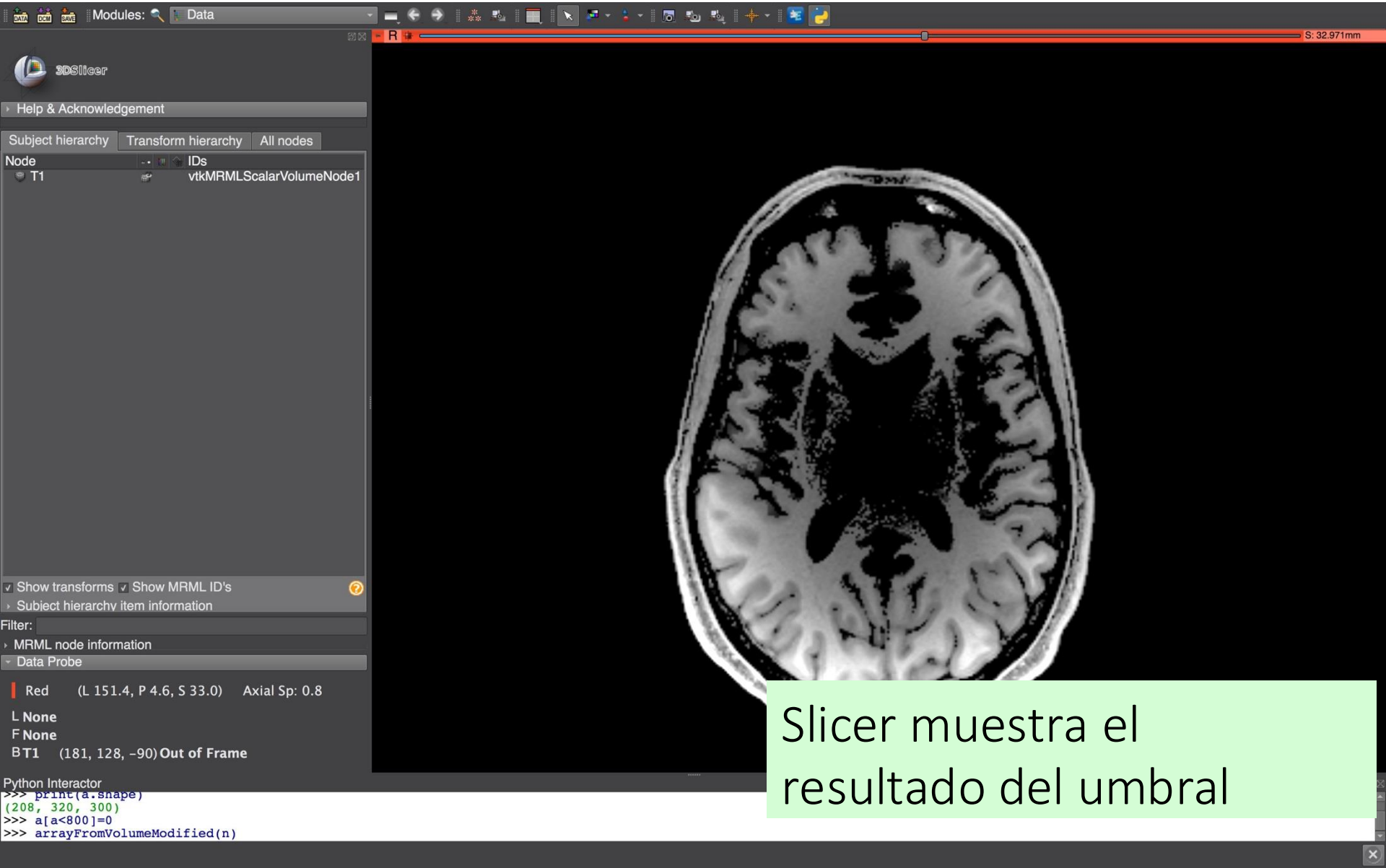


Ejecute los siguientes comandos en la consola de Python:

```
a[a<800]=0
arrayFromVolumeModified(n)
```

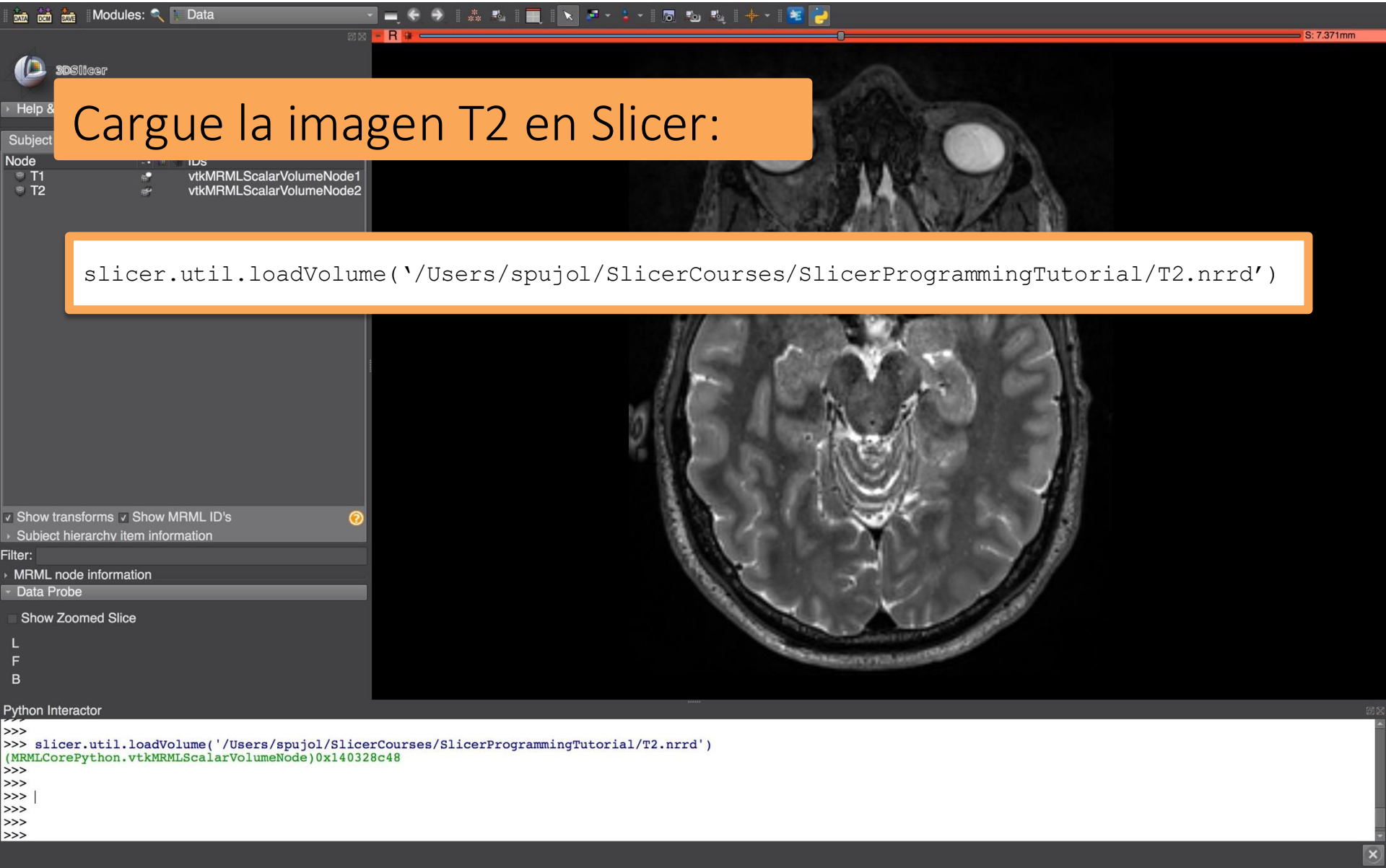
Este ejemplo aplica un umbral de t=800 a la imagen y notifica la modificación a Slicer

Modificación de vóxeles en un volumen

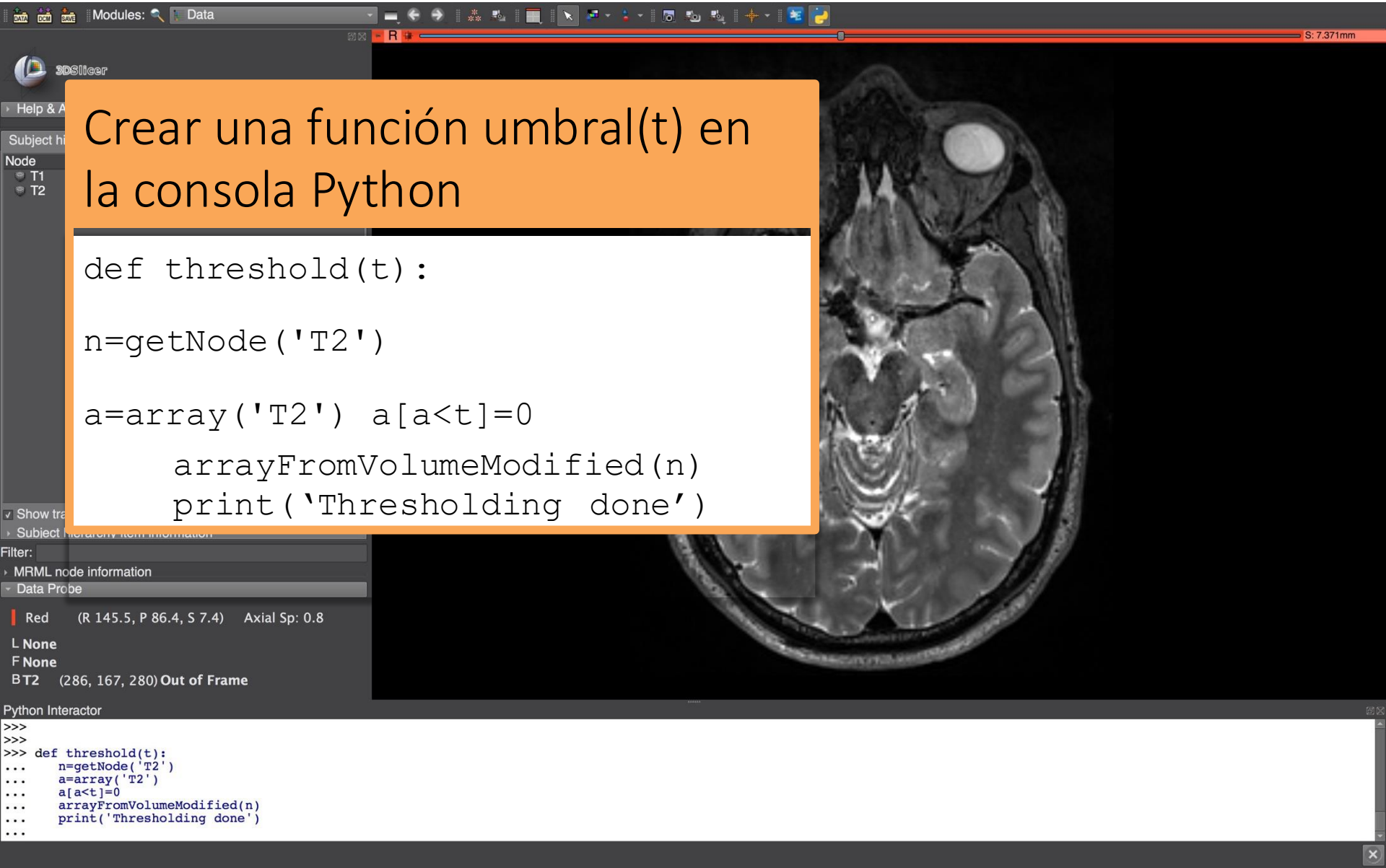


Slicer muestra el resultado del umbral

Cargar el volumen T2



Función Python: umbral



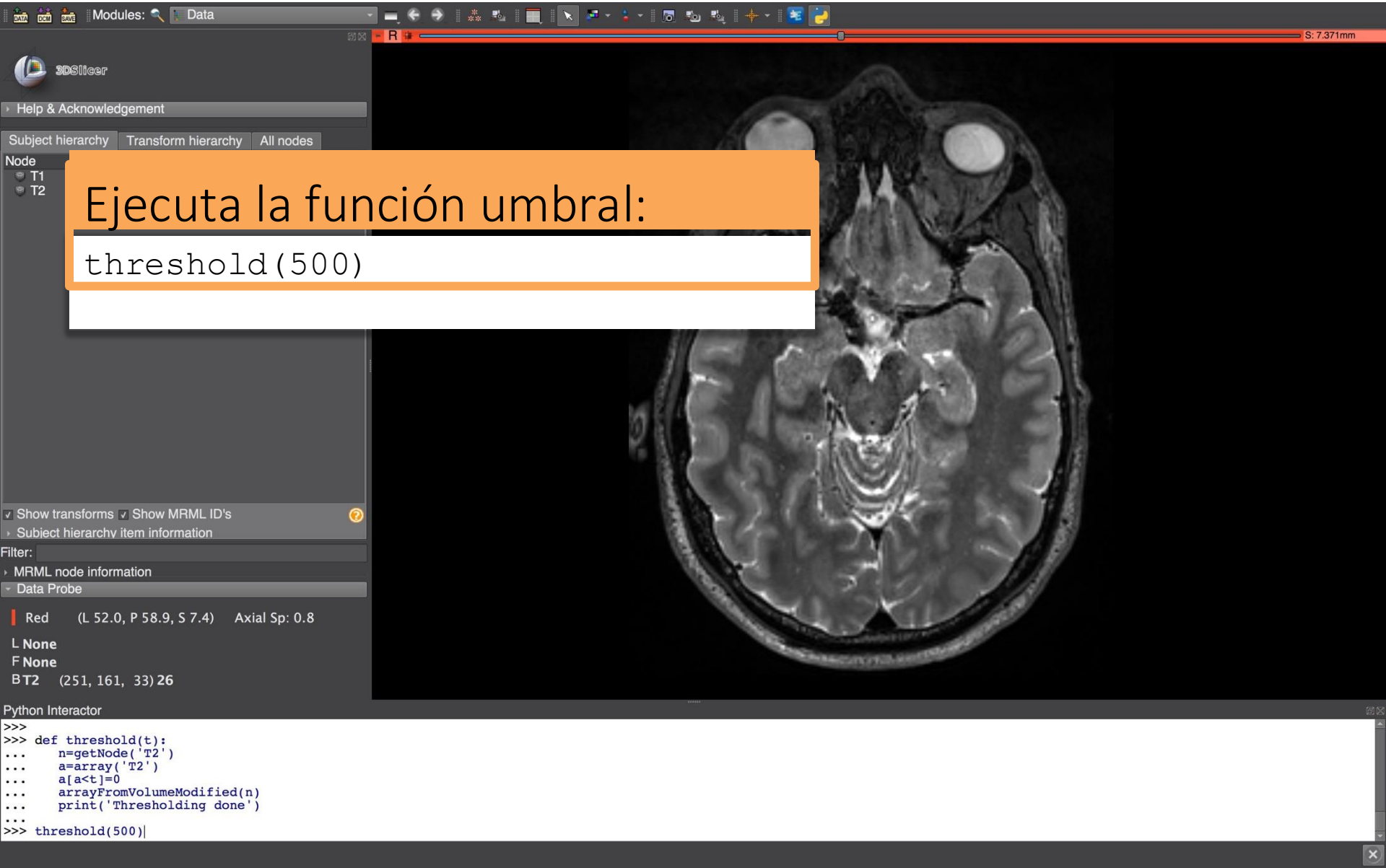
Crear una función umbral(t) en la consola Python

```
def threshold(t):

n=getNode('T2')

a=array('T2') a[a<t]=0
     arrayFromVolumeModified(n)
     print('Thresholding done')
```

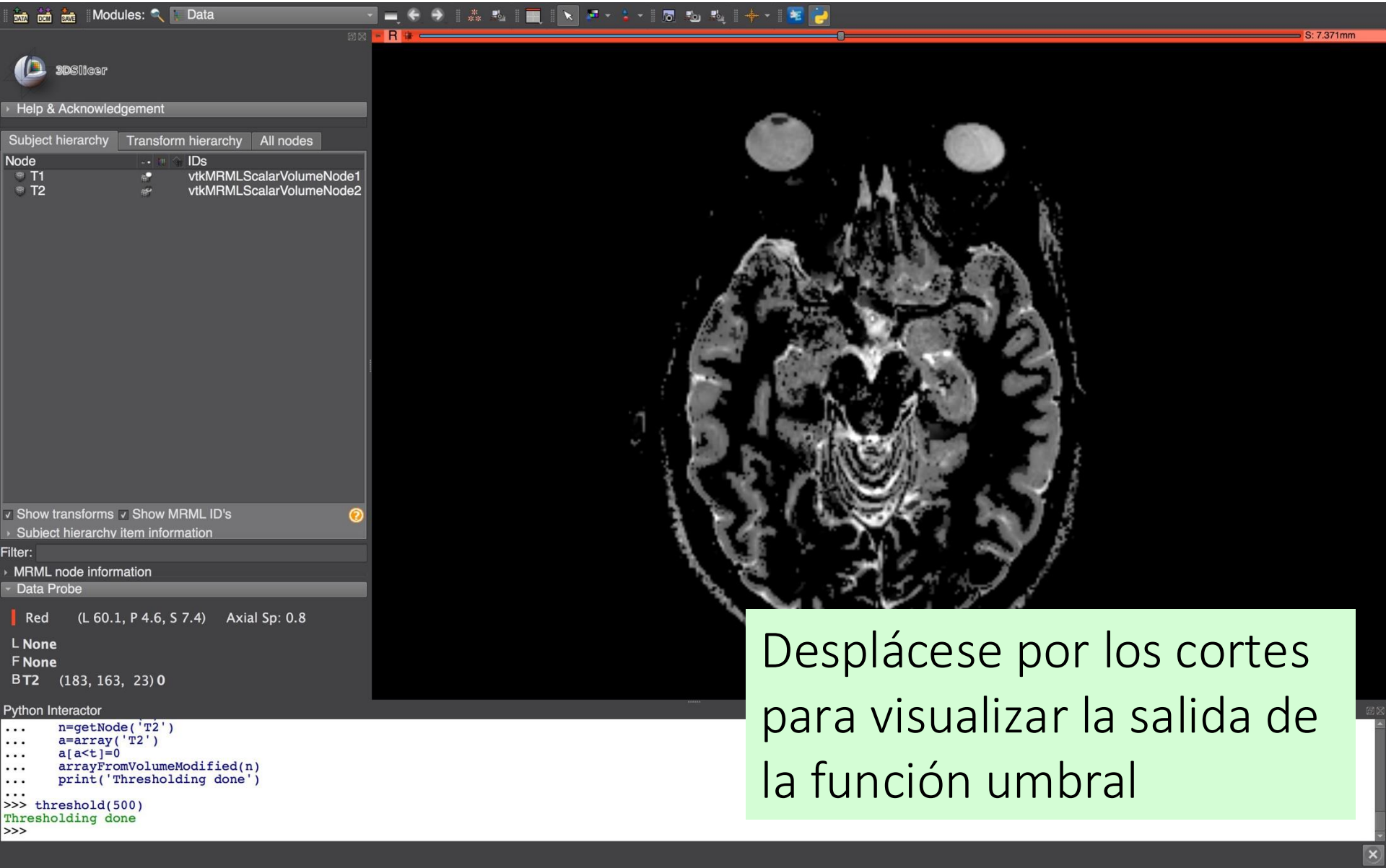
Función Python: umbral



Ejecuta la función umbral:

```
threshold(500)
```

Función Python: umbral



Panaroma General

- Slicer facilita el acceso para analizar y modificar tipos de datos complejos
- Slicer es compatible con una amplia gama de paquetes informáticos científicos de Python.
- Slicer es un entorno de investigación para realizar experimentos con imágenes médicas

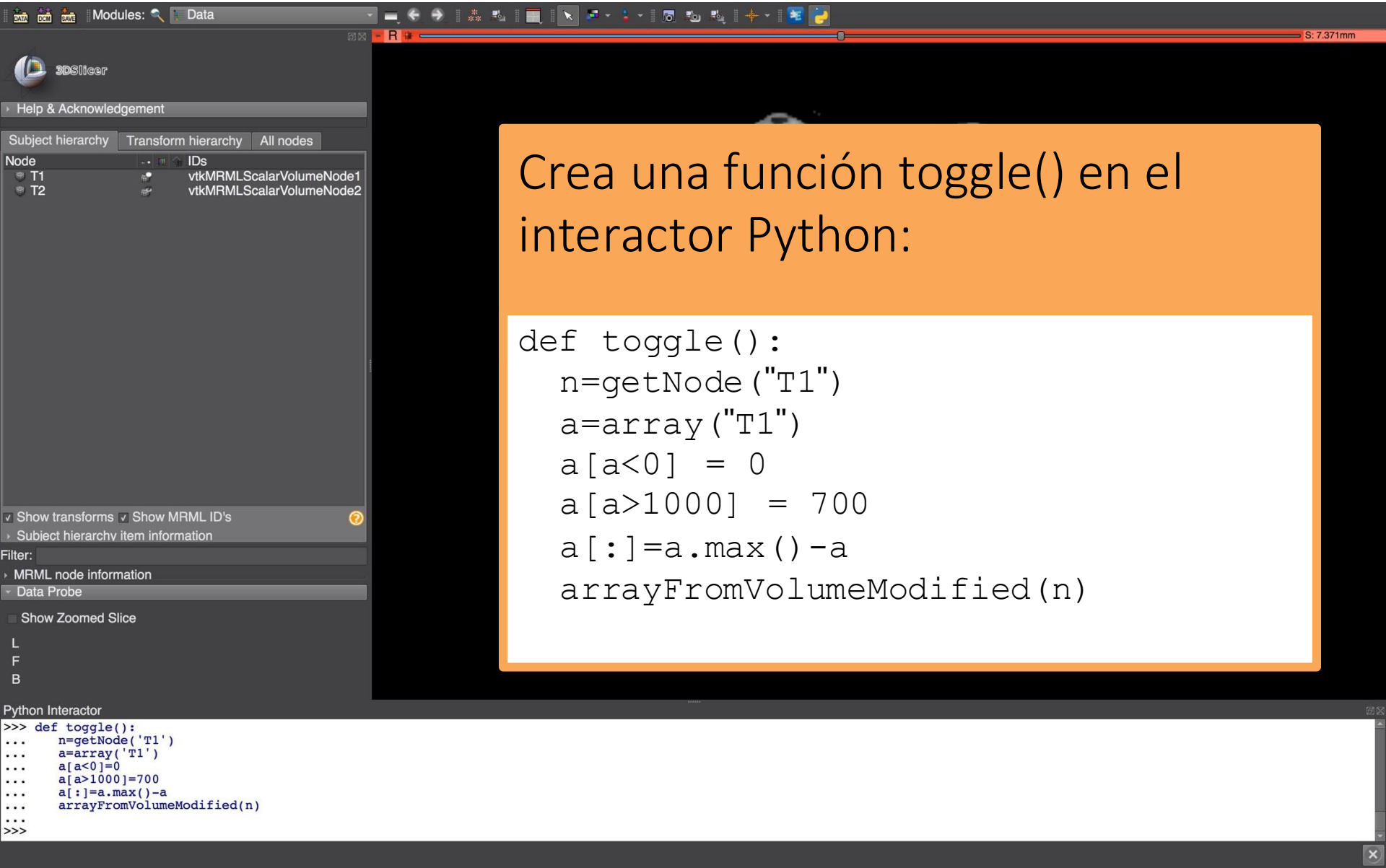
Parte 3

Familiarizarse con Qt en Slicer

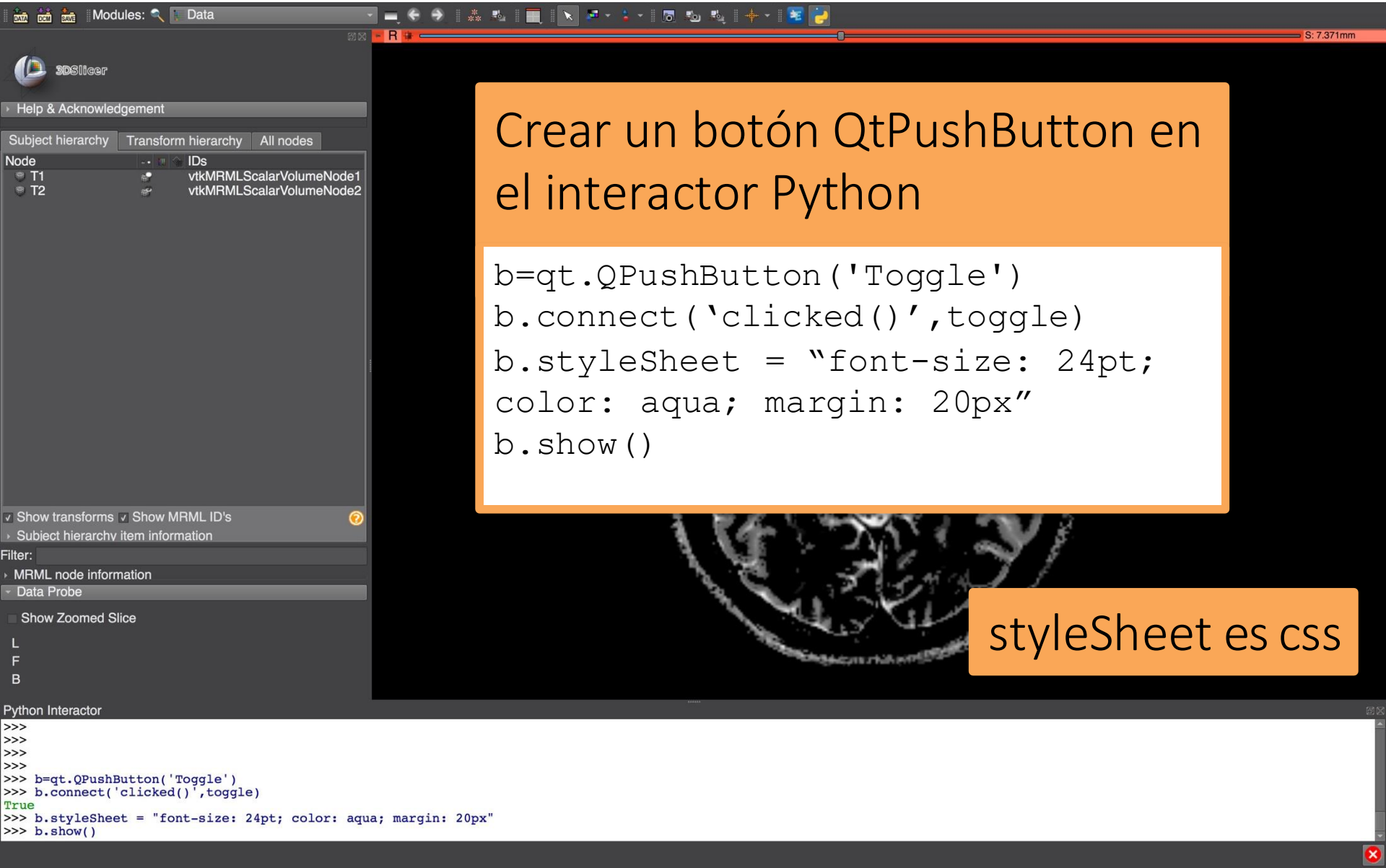
Qt & PythonQt

- Qt es la herramienta principal de Slicer para crear widgets, cuadros de diálogo, entradas de texto, etc.
- PythonQt expone la mayoría de las funcionalidades de Qt y es accesible a través del interactuador Python en Slicer.
- Las interfaces de usuario se pueden crear sobre la marcha para una rápida creación de prototipos y depuración.

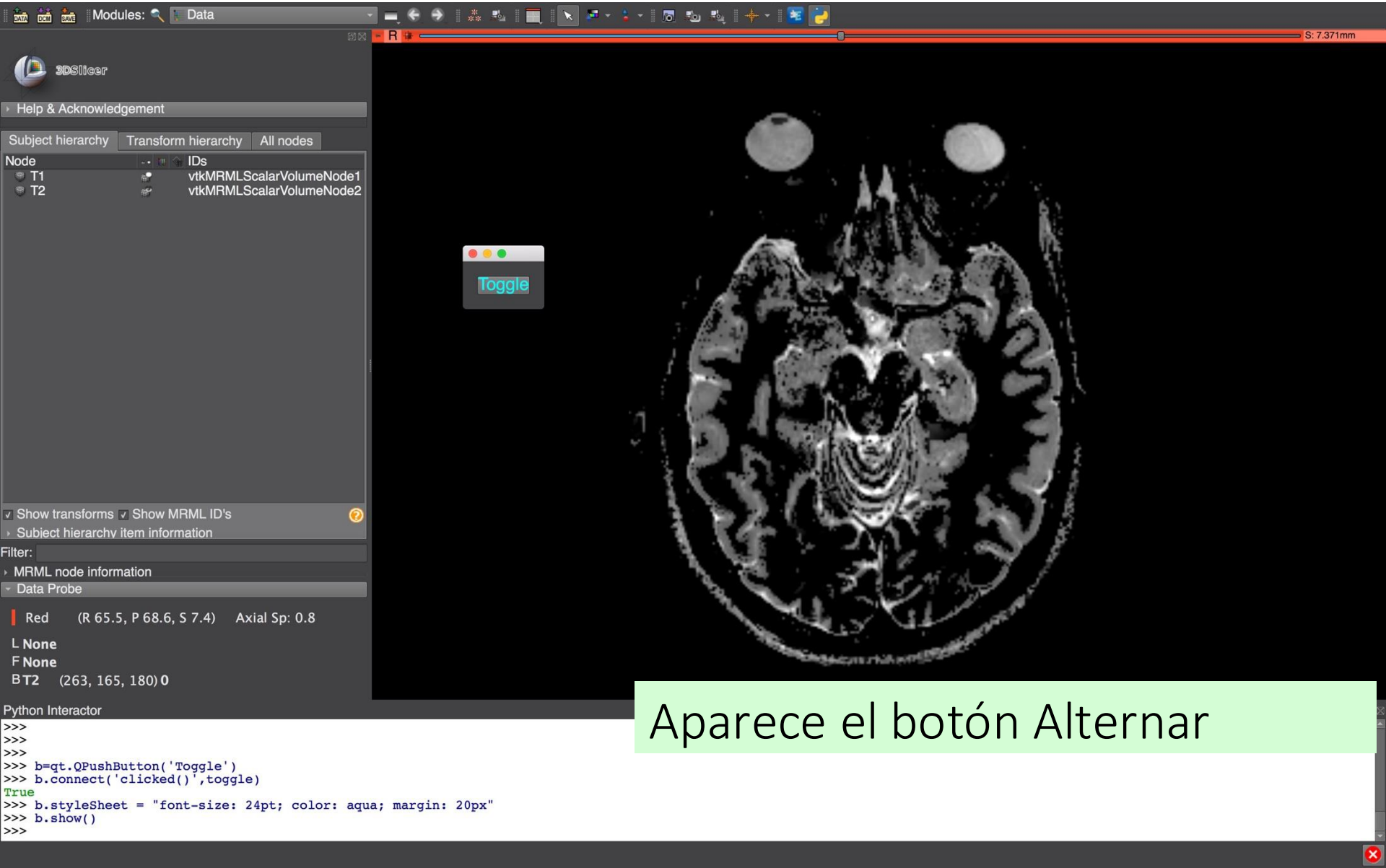
Función Python: toggle



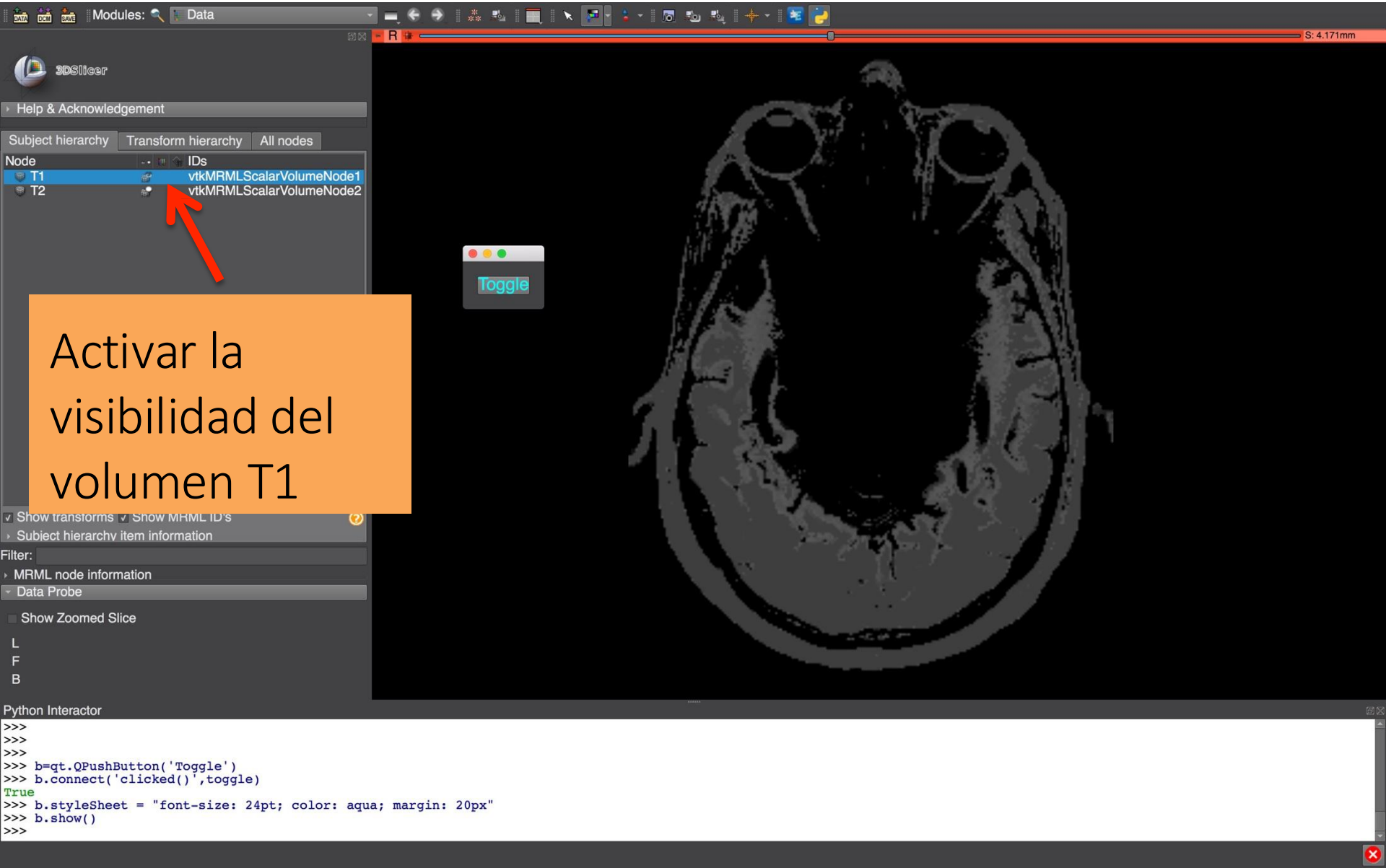
Creación de un botón Qt



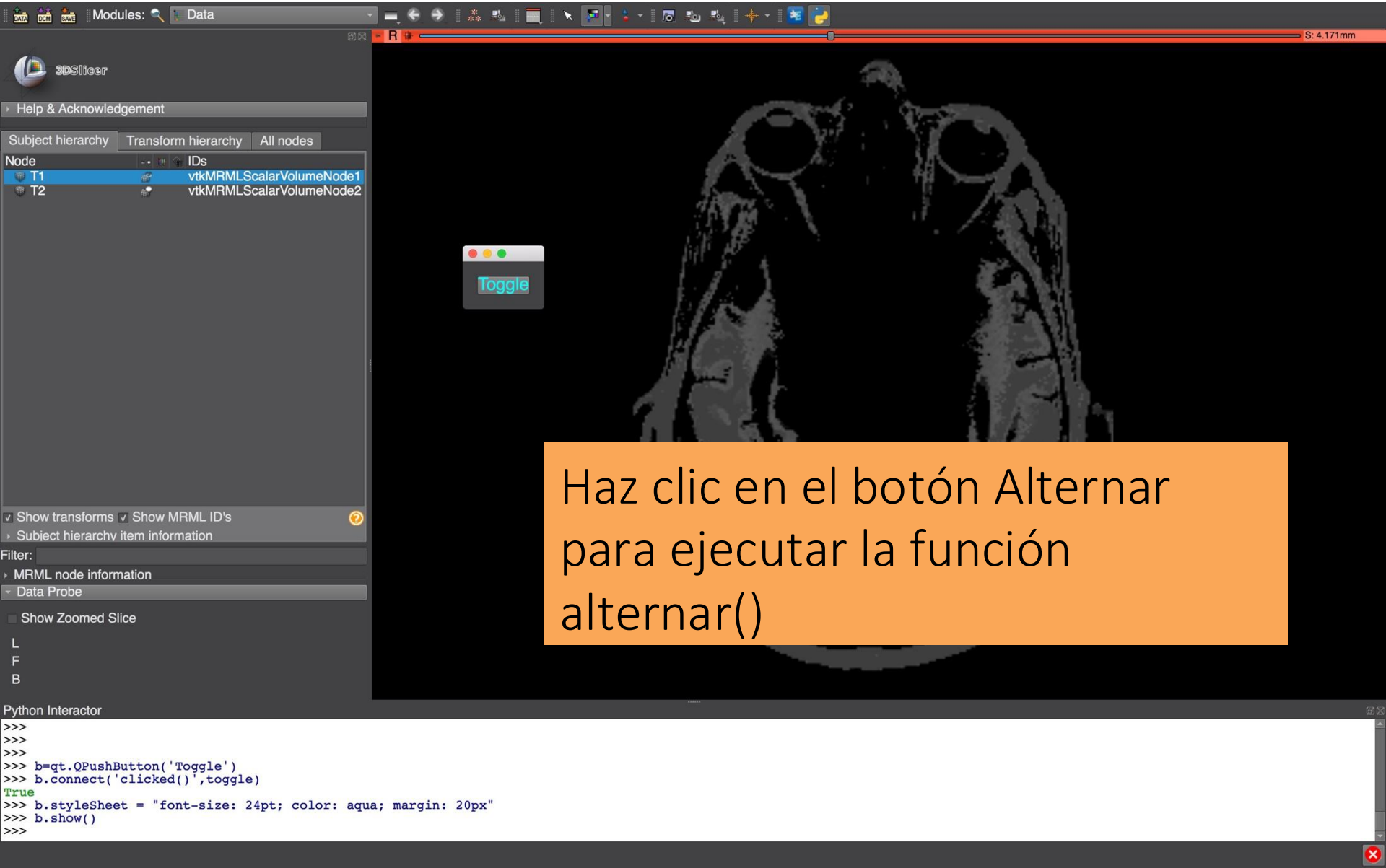
Creación de un botón Qt



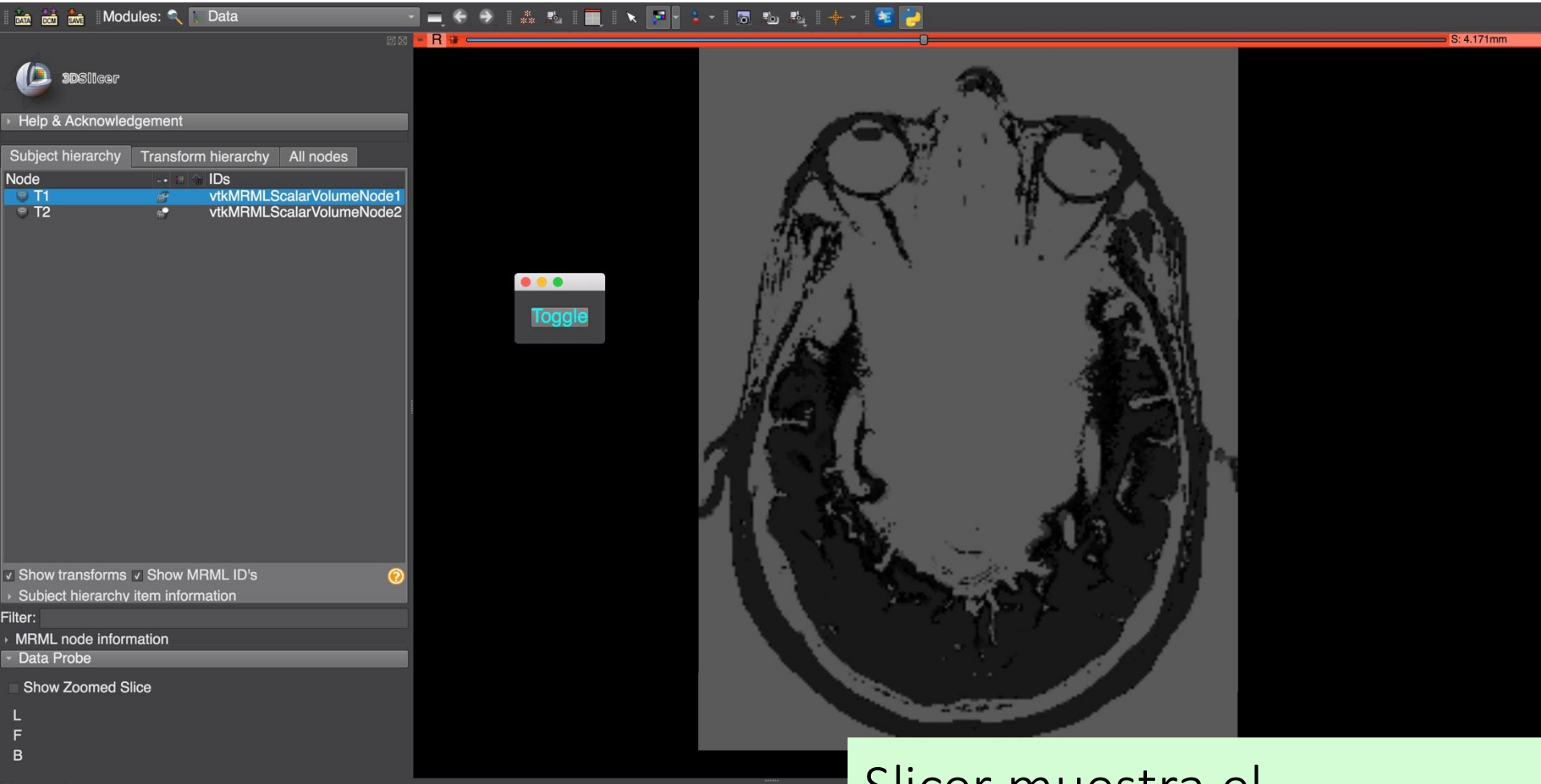
Creación de un botón Qt



Creación de un botón Qt



Creación de un botón Qt



Slicer muestra el resultado del tratamiento de la imagen

Ejemplos de módulos con secuencias de comandos

- El tutorial muestra cómo crear una interfaz sencilla en Python.
- Slicer integra muchos módulos de secuencias de comandos sofisticados, como estadísticas de segmentos, datos de muestras, módulo de endoscopia, etc.
- Para más información, consulte el repositorio de secuencias de comandos de Slicer:

https://www.slicer.org/wiki/Documentation/Nightly/ScriptRepository

Conclusión

- Slicer permite a los desarrolladores crear interfaces complejas y ágiles para los usuarios
- La plataforma de software ofrece posibilidades de personalización ilimitadas
- Slicer da acceso a bibliotecas subyacentes avanzadas a través de un paquete multiplataforma que es fácil de desplegar a los usuarios

Agradecimientos

Neuroimage Analysis Center
(NIBIB P41 EB015902)

Sylvain Bouix, Ph.D.
Psychiatry Neuroimaging Laboratory