Object dragging in scenegraph-based systems

Tim Weißker

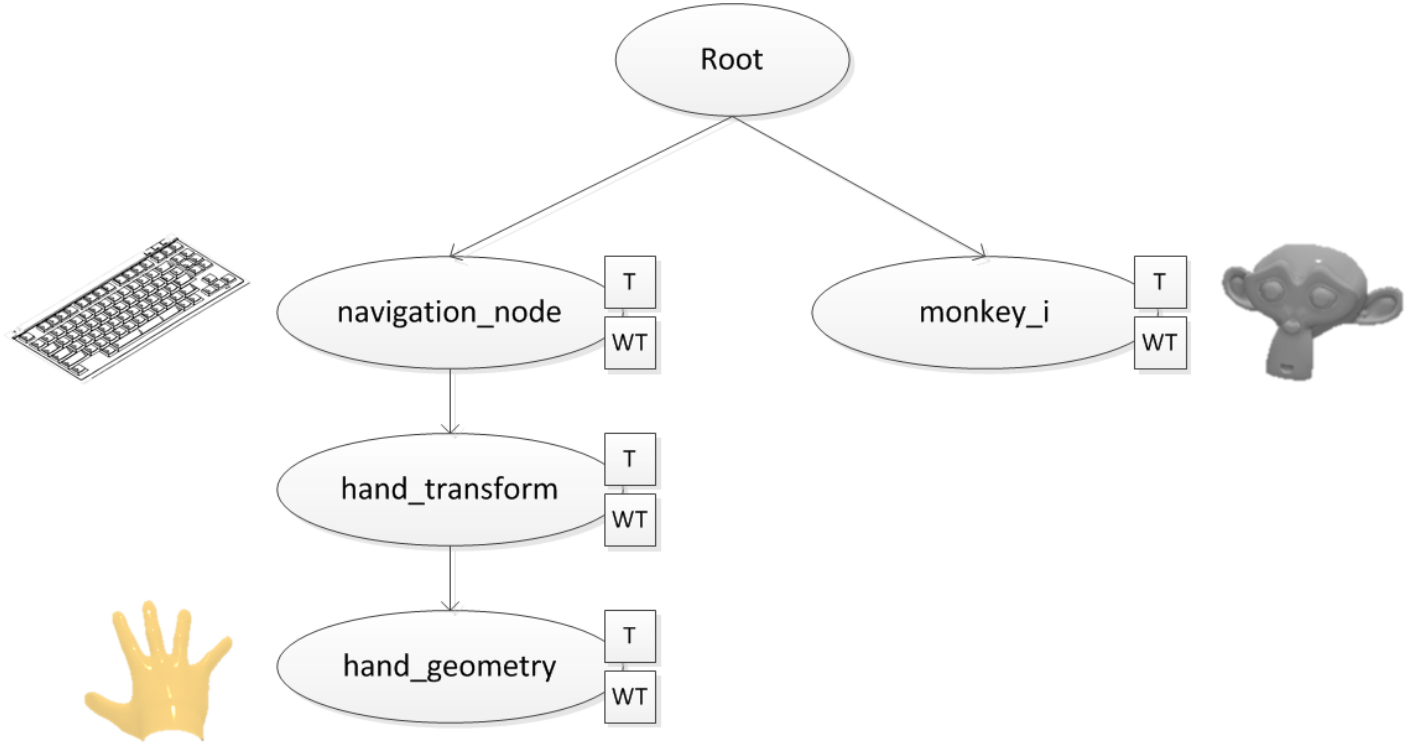

40
20
0
rnd
59.8
app
59.8


40
20
0
rnd 59.9
app 59.8

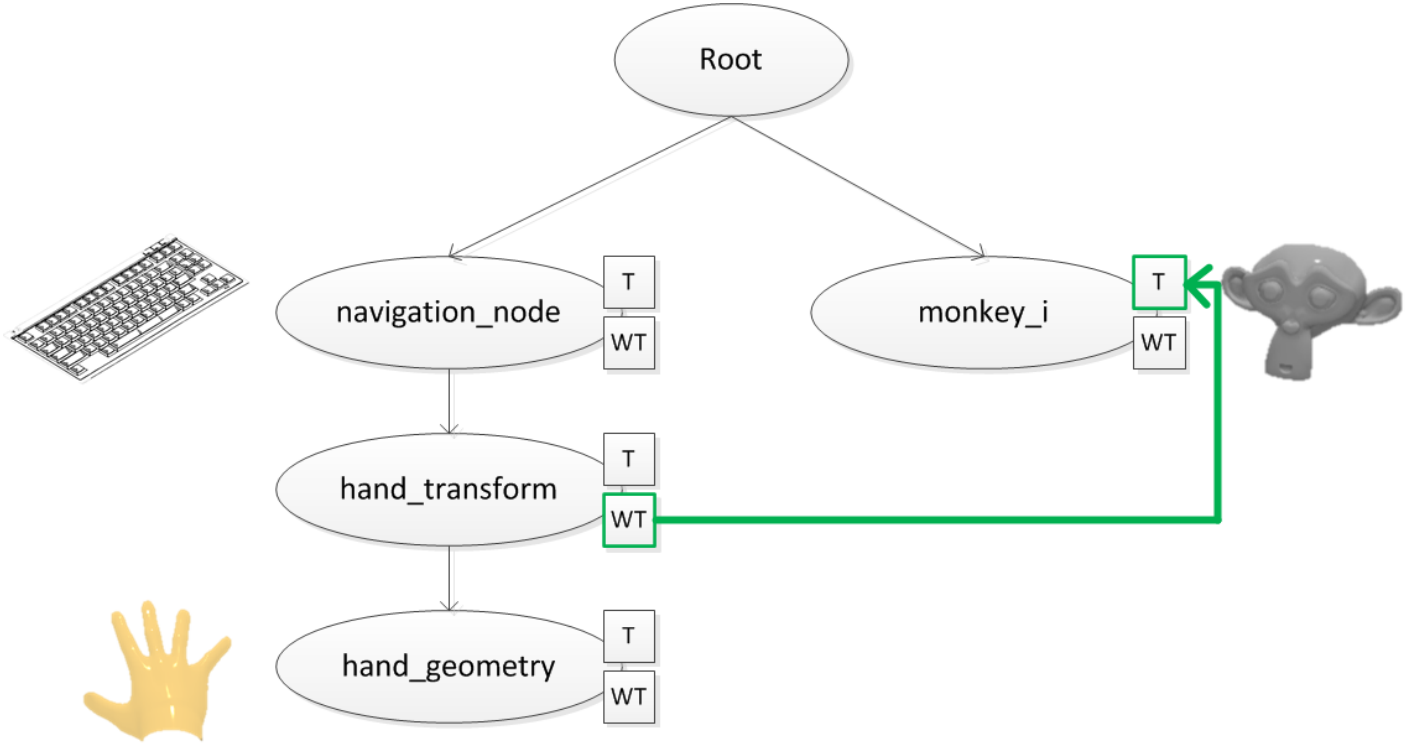

40
20
0
rnd
60
app
59.8

Scenegraph structure

Scenegraph structure



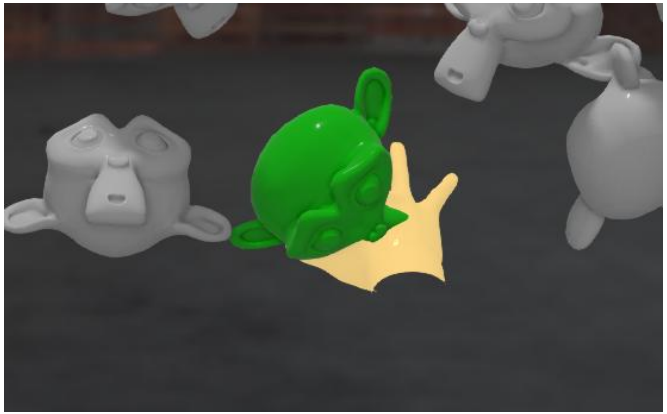
Naïve dragging




40
20
0
rnd
59.9
app
59.8

Preserving the contact point

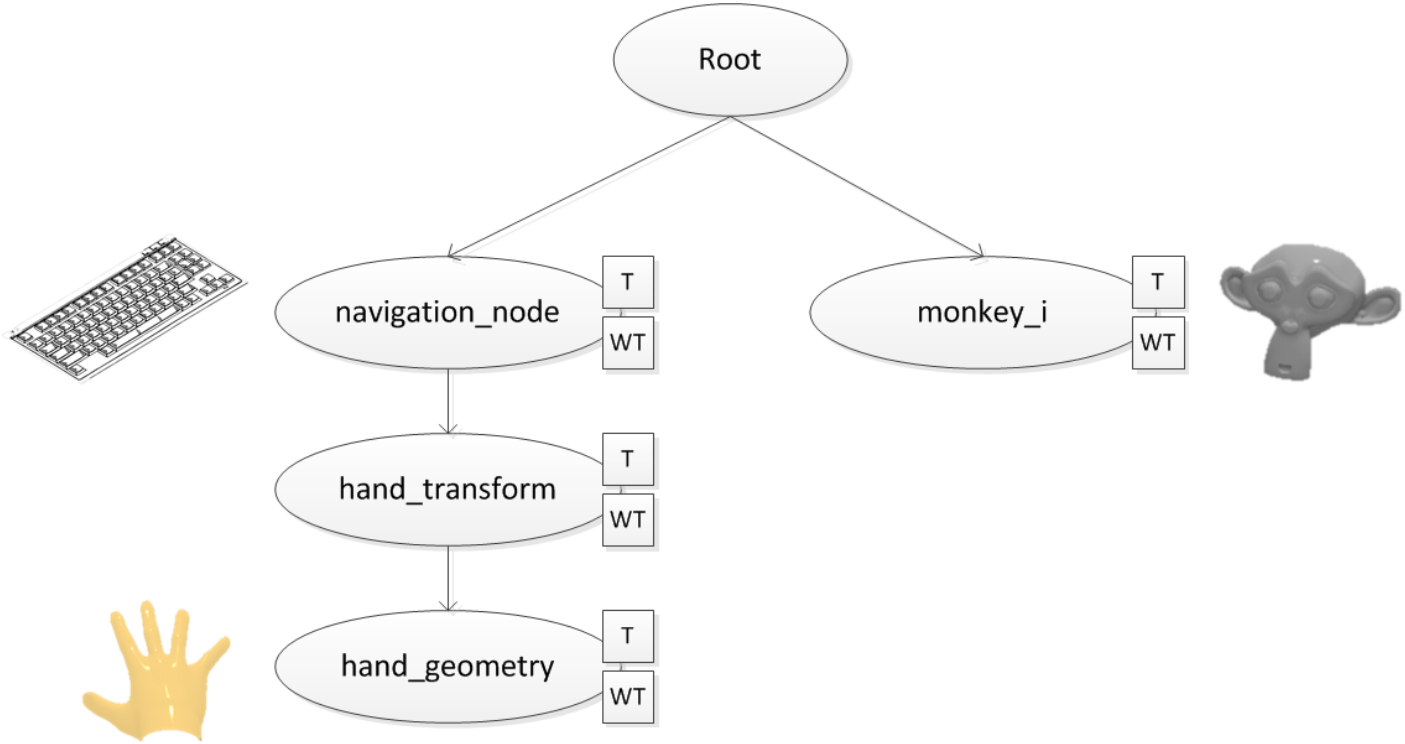
Hand-Monkey offset



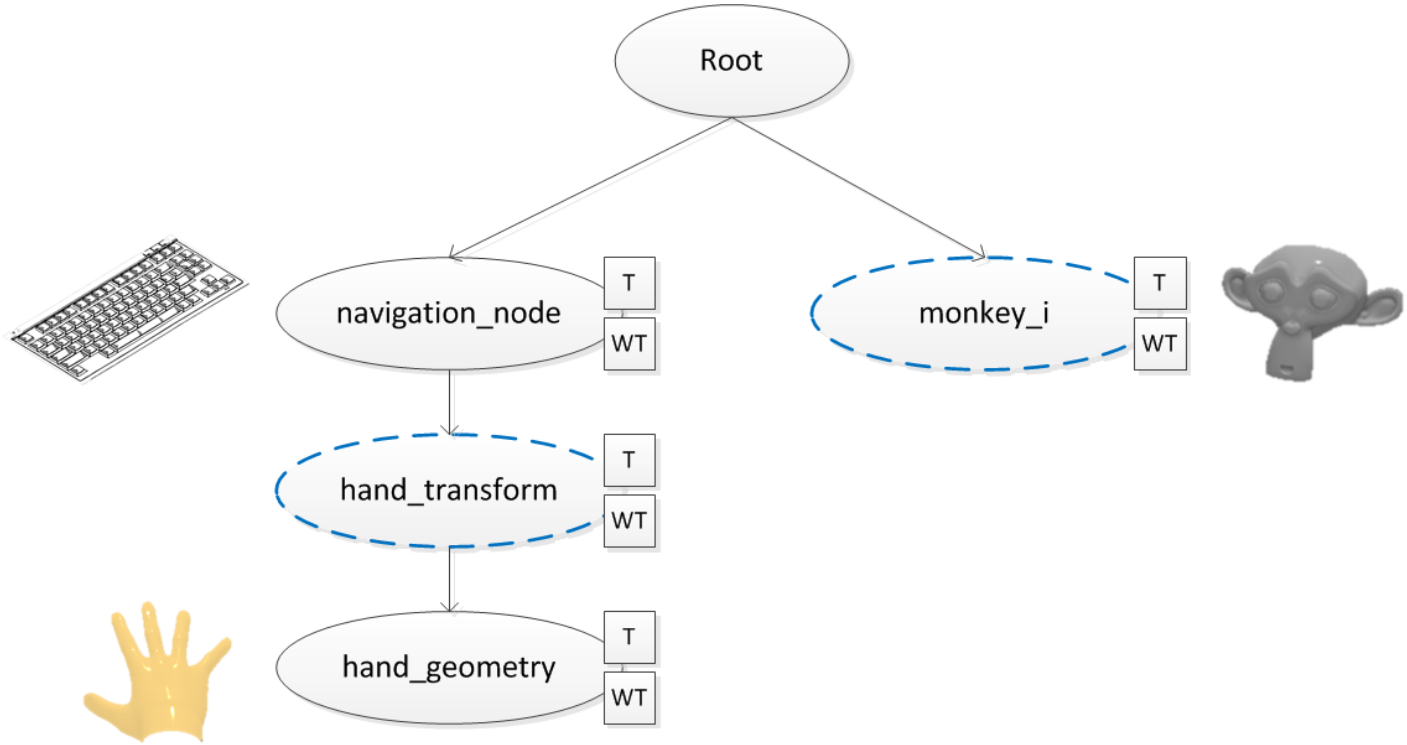
We want to keep the offset between the hand and the monkey during dragging!

How do we compute it?

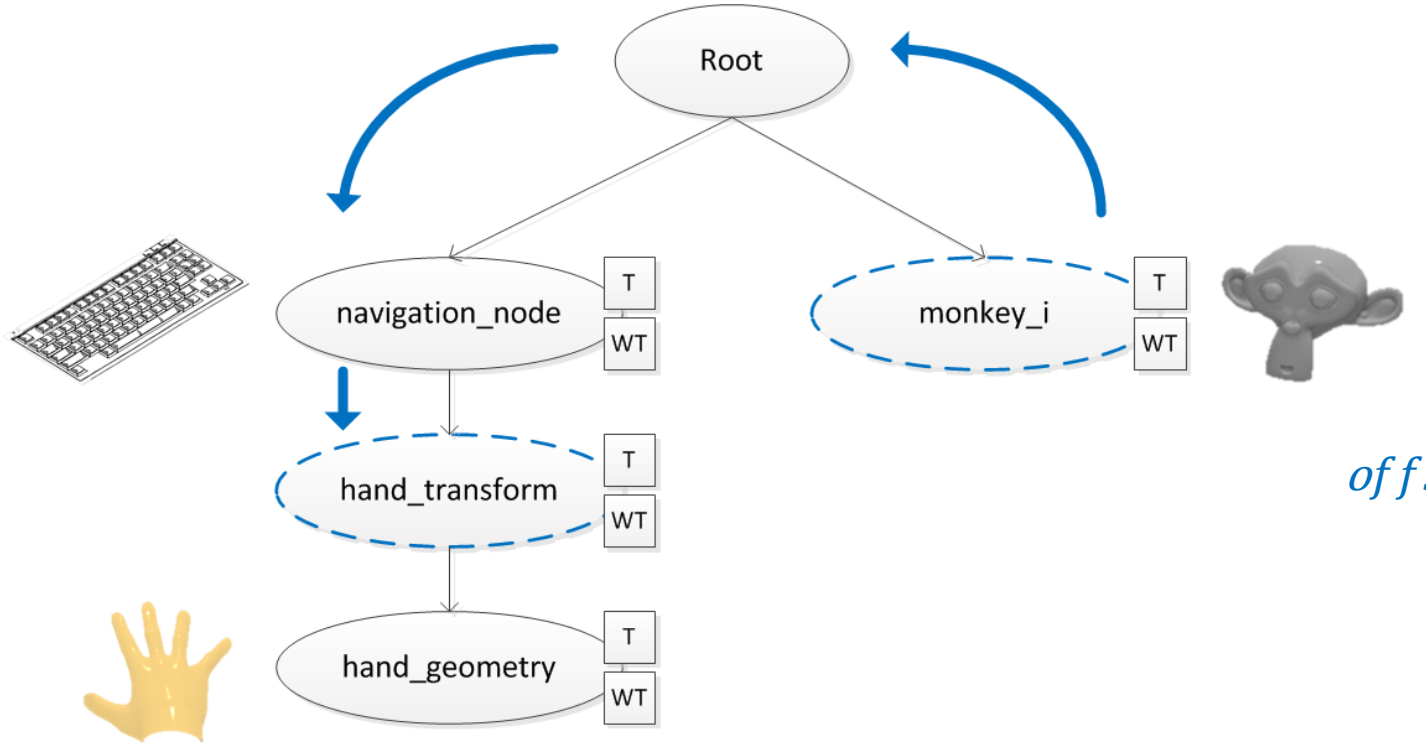
Hand-Monkey offset



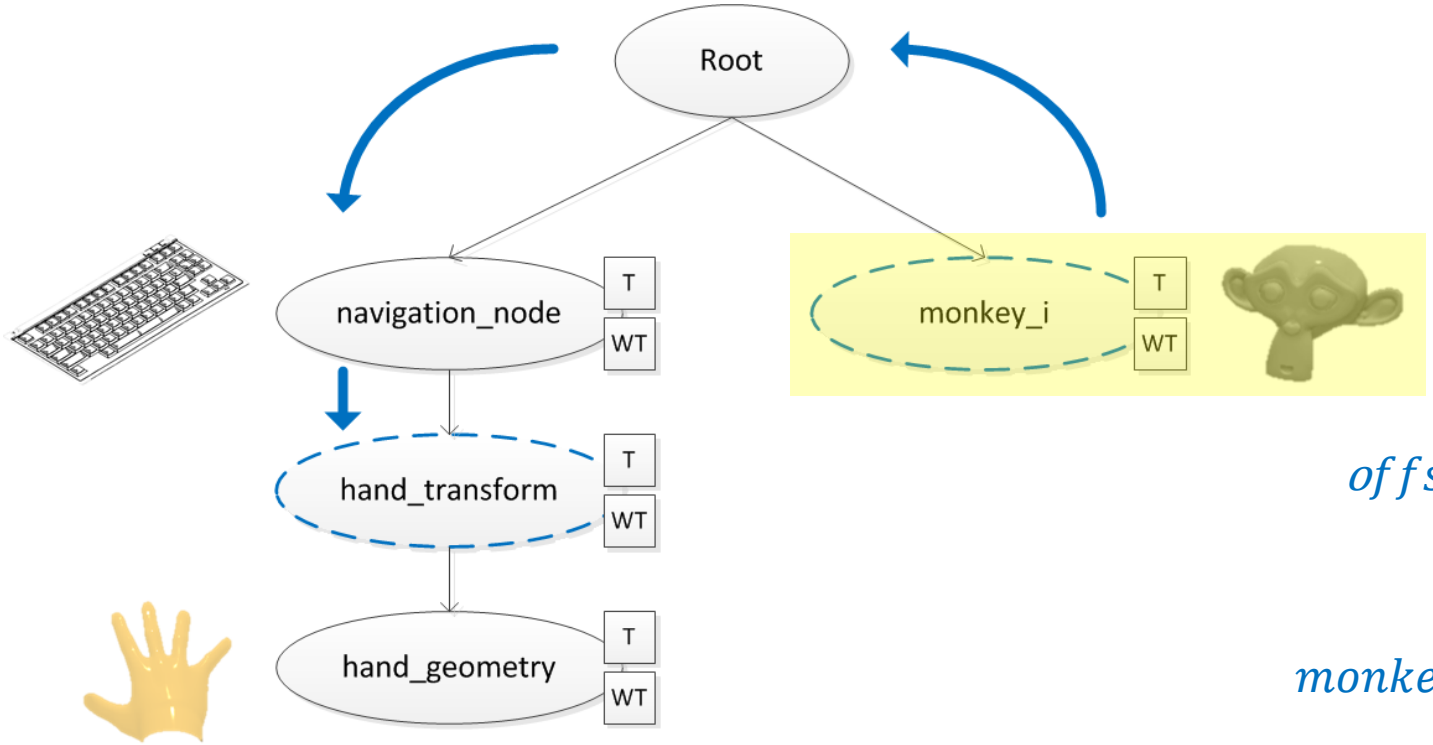
Hand-Monkey offset



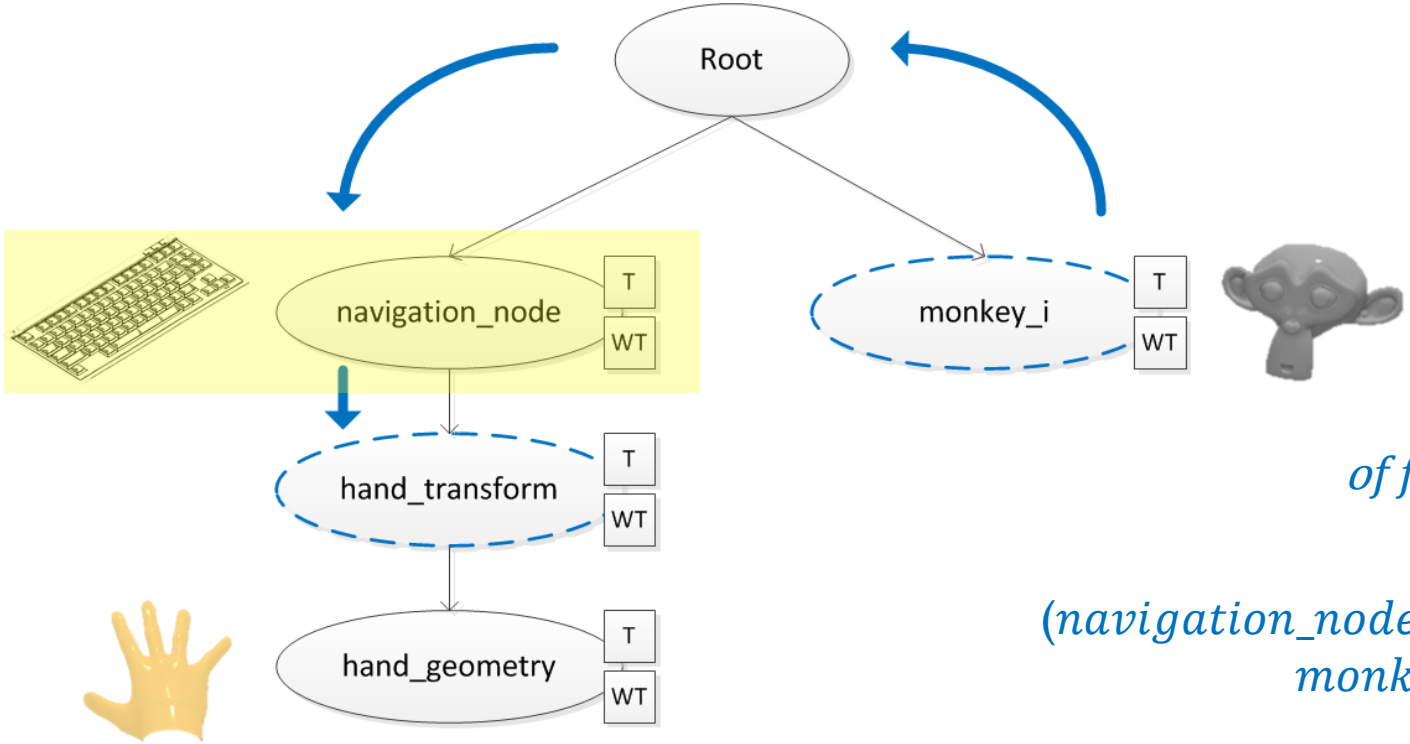
Hand-Monkey offset



Hand-Monkey offset

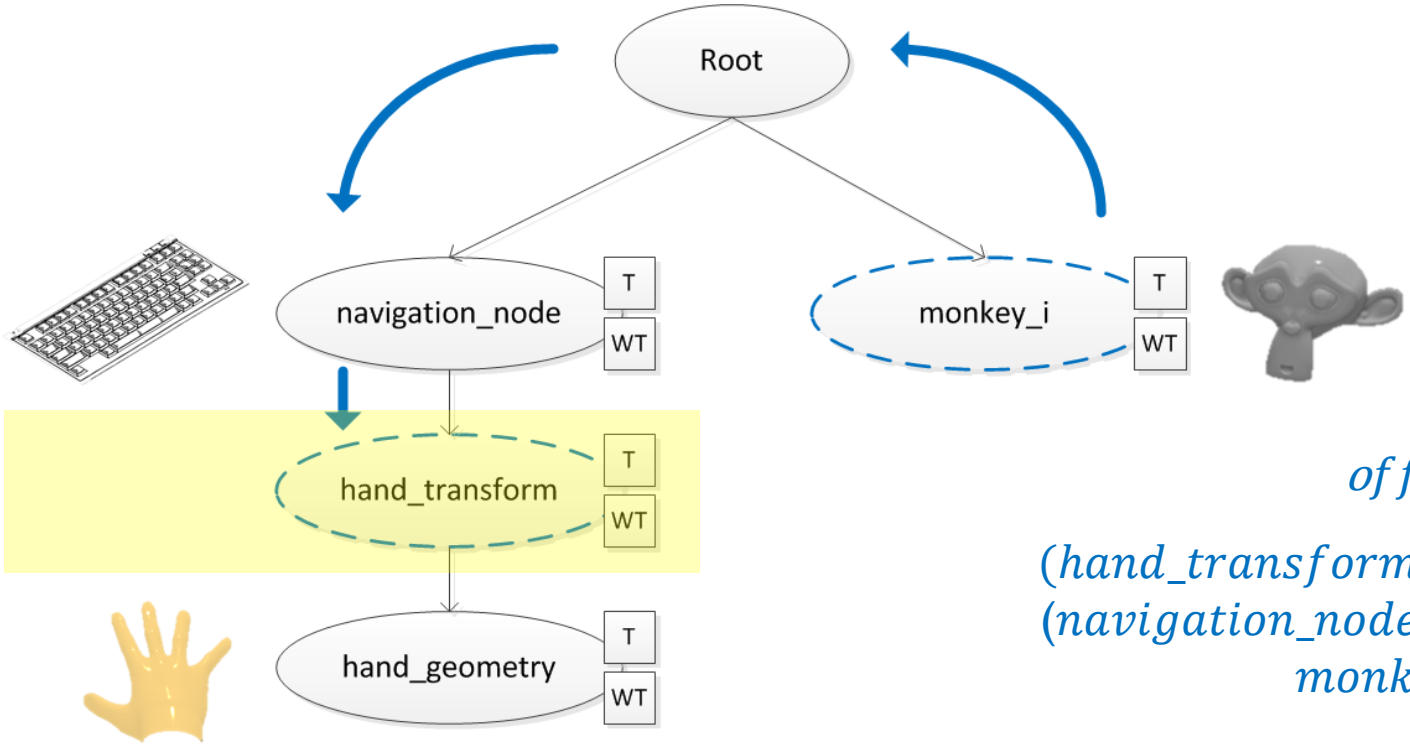


Hand-Monkey offset



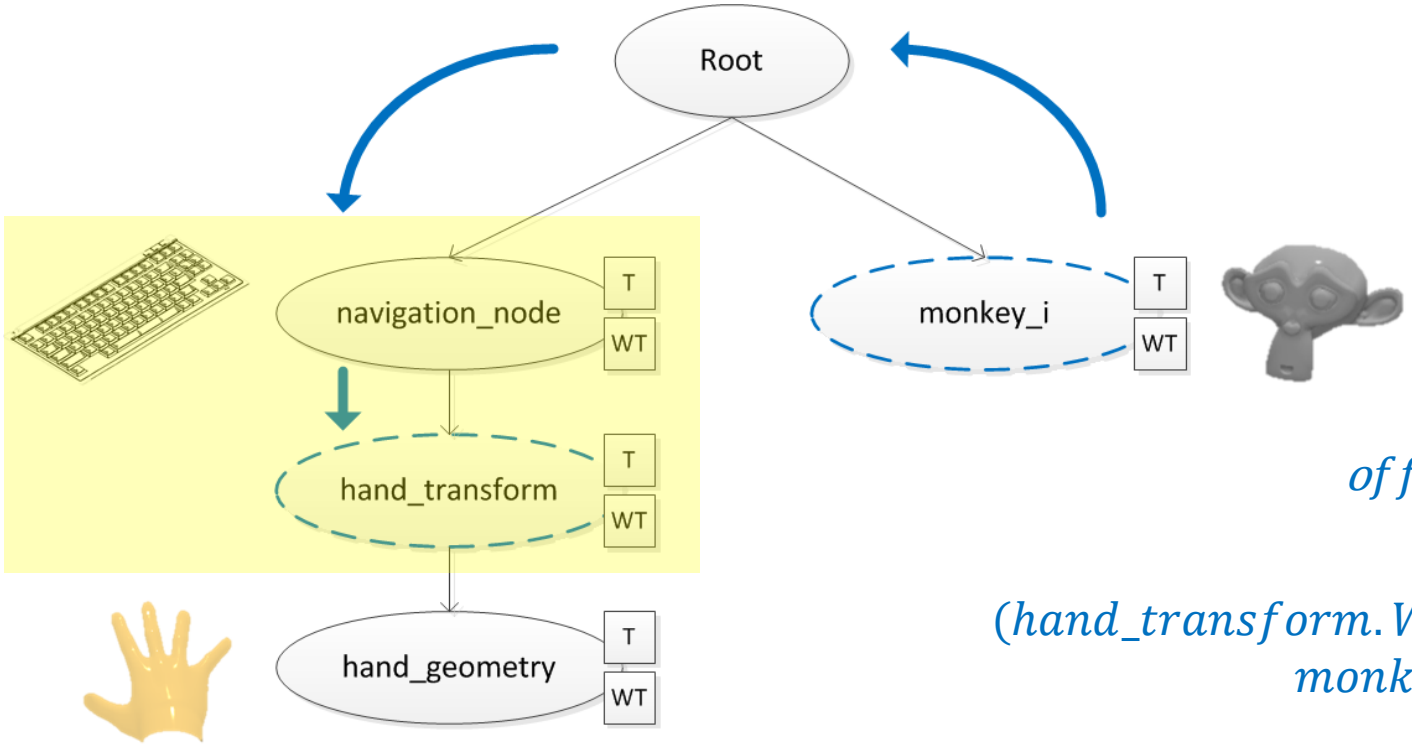
$$offset = (navigation\_node.T)^{-1} \cdot monkey\_i.T$$

Hand-Monkey offset



$$offset = (hand\_transform.T)^{-1} \cdot (navigation\_node.T)^{-1} \cdot monkey\_i.T$$

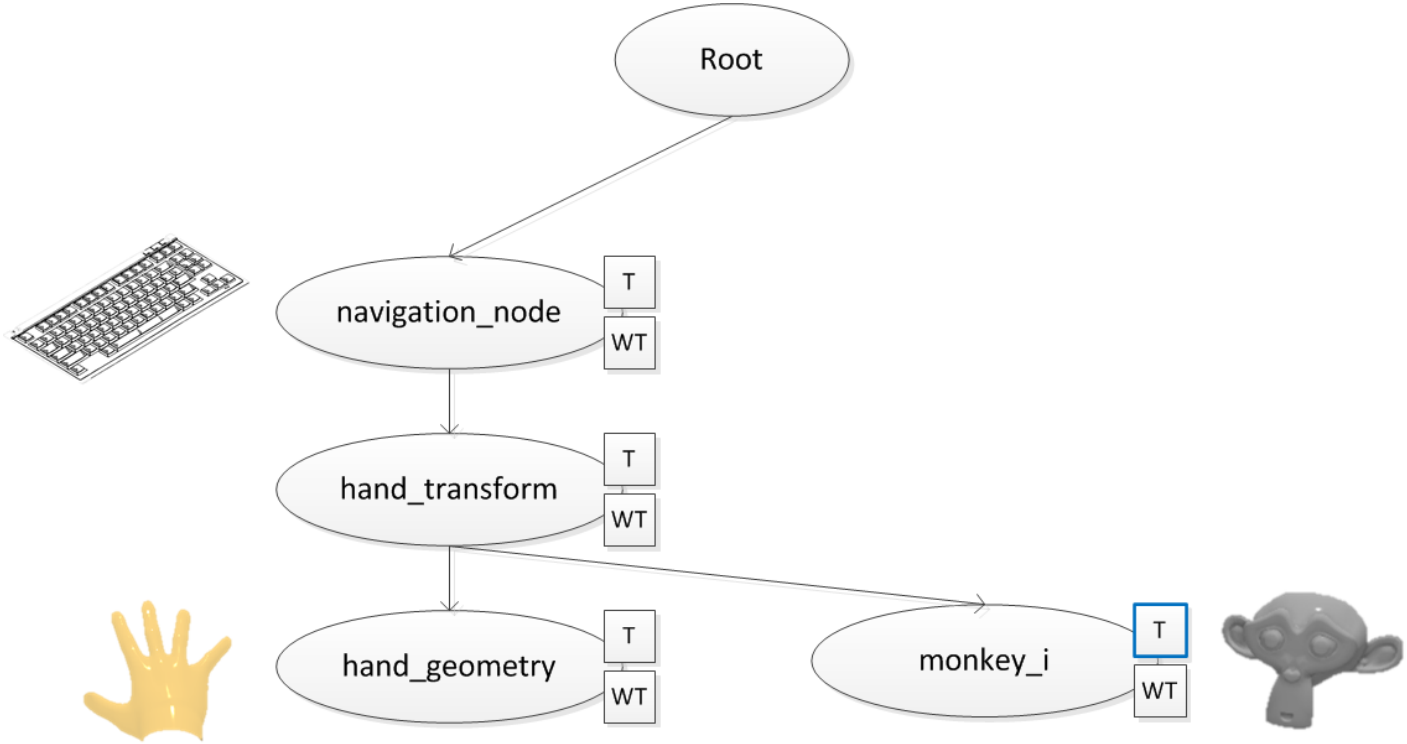
Hand-Monkey offset



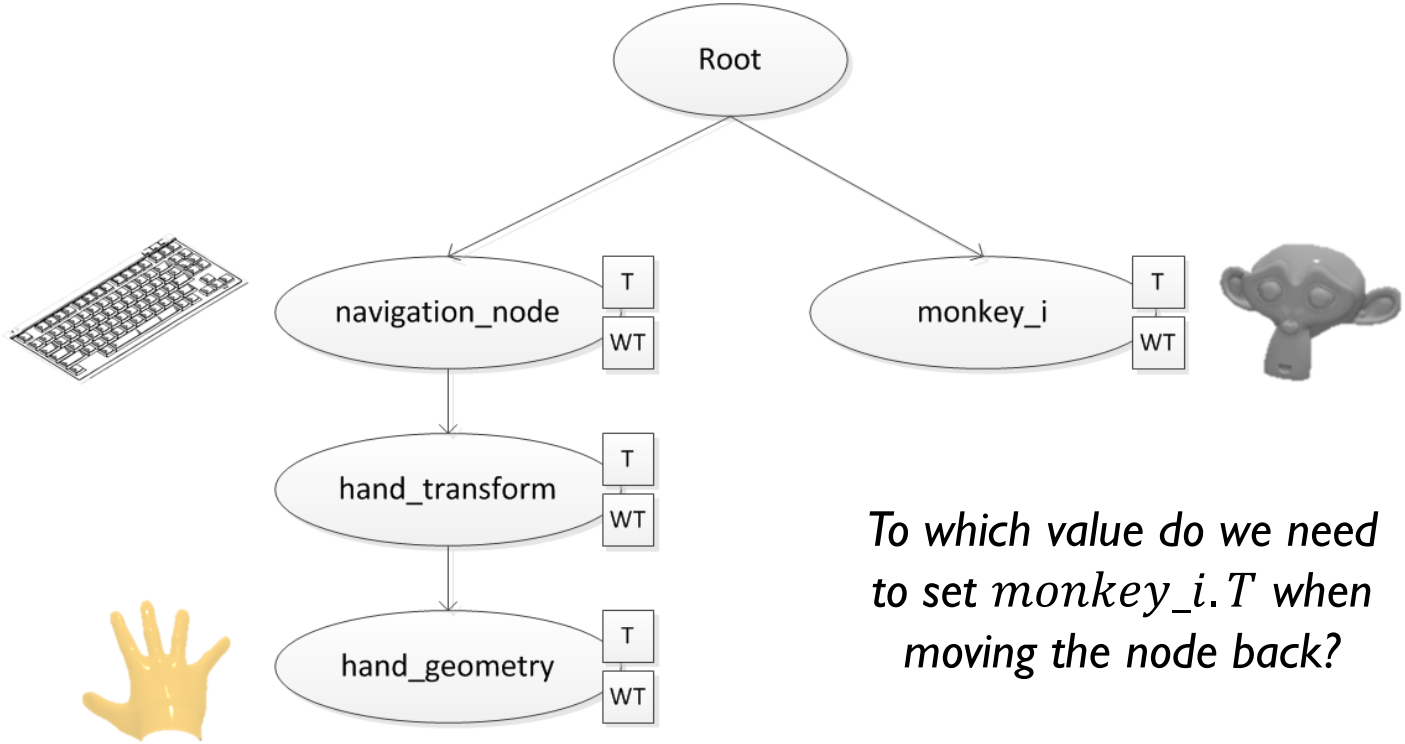
$$offset = (hand\_transform.WT)^{-1} \cdot monkey\_i.T$$

Dragging strategy 1

Dragging Strategy 1



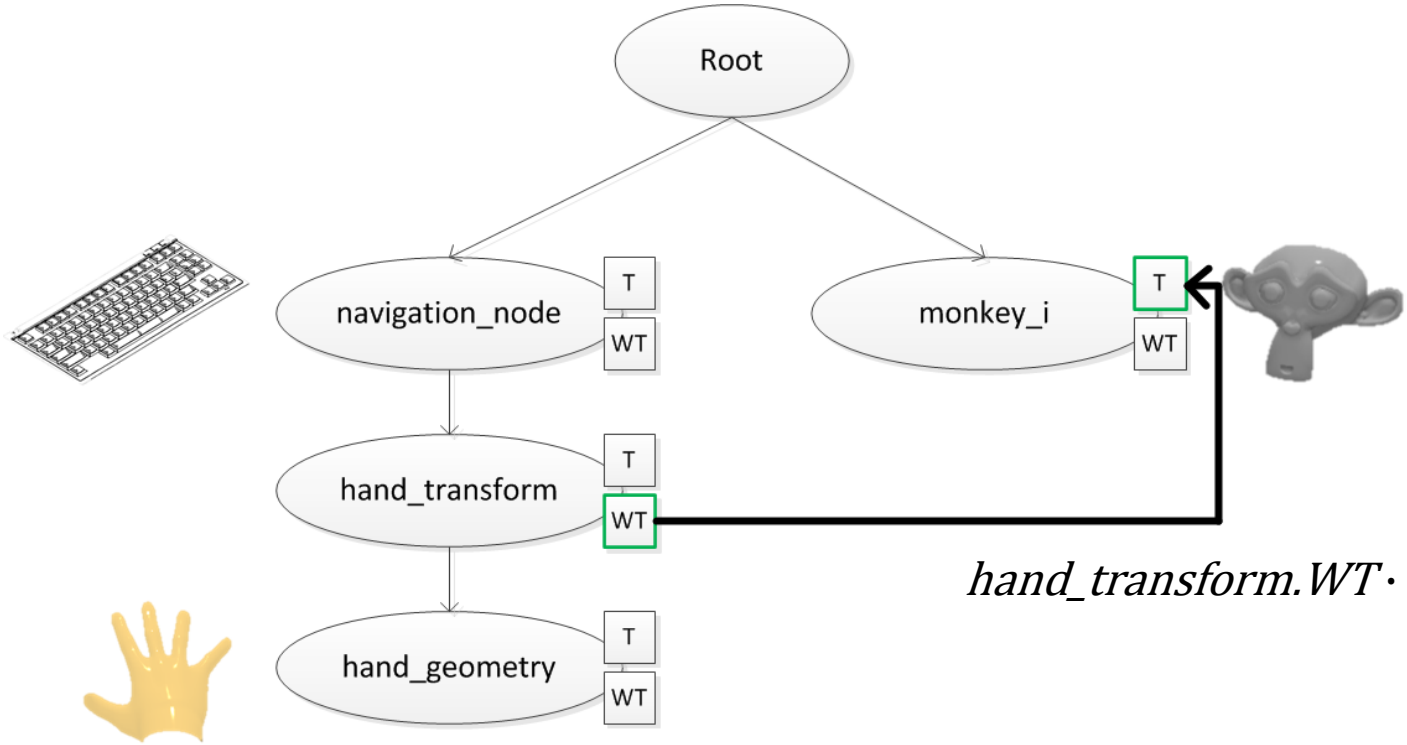
Dragging Strategy 1



To which value do we need to set $monkey\_i.T$ when moving the node back?

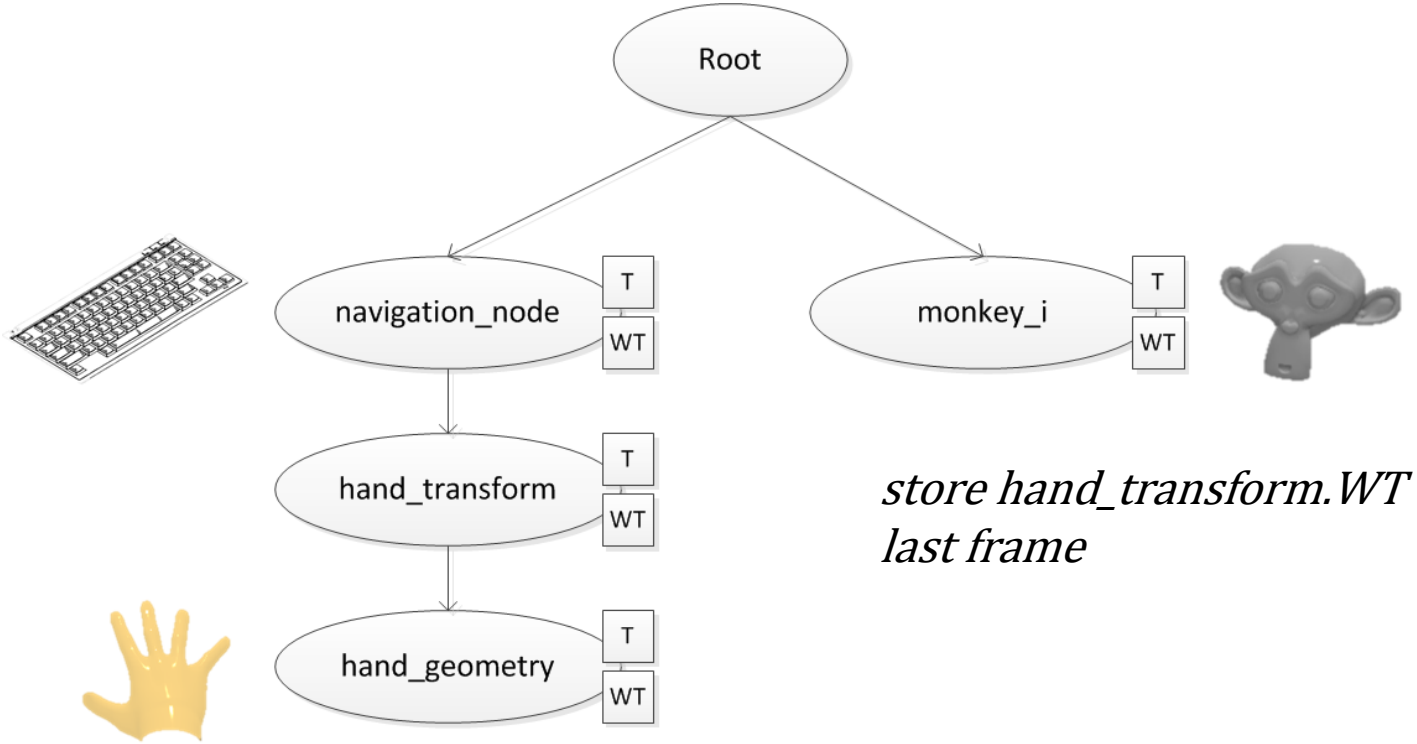
Dragging strategy 2

Dragging Strategy 2

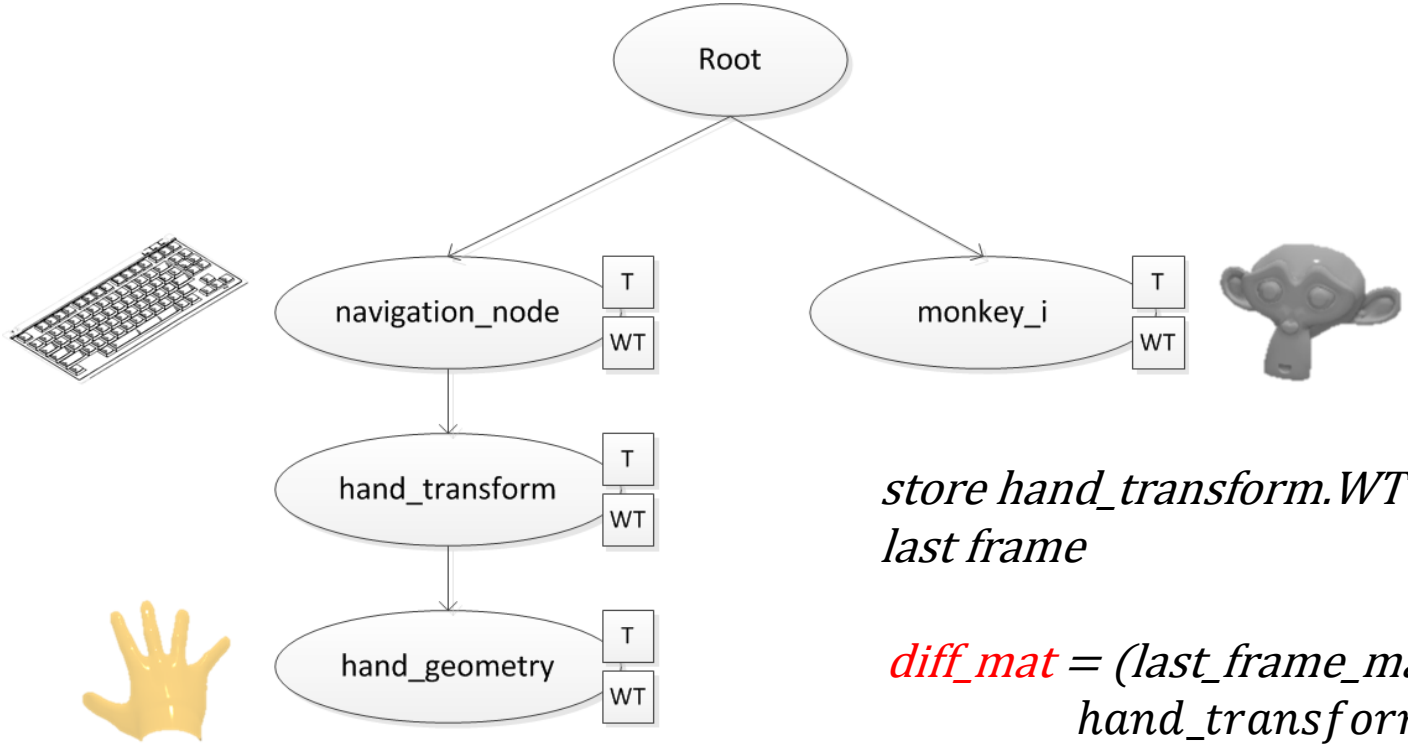


Dragging strategy 3

Dragging Strategy 3



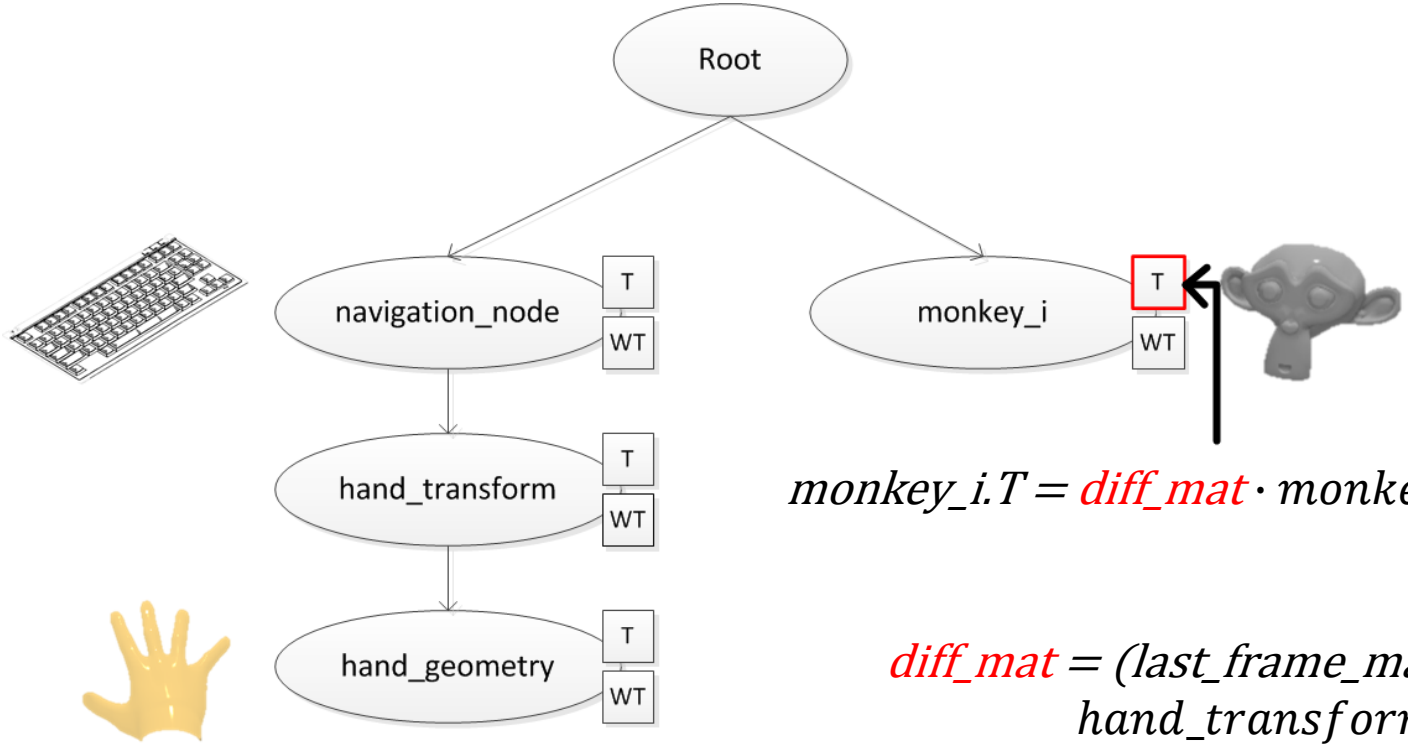
Dragging Strategy 3



store hand_transform.WT of last frame

$$diff\_mat = (last\_frame\_mat)^{-1} \cdot hand\_transform.WT$$

Dragging Strategy 3



$$monkey\_i.T = diff\_mat \cdot monkey\_i.T$$

$$diff\_mat = (last\_frame\_mat)^{-1} \cdot hand\_transform.WT$$

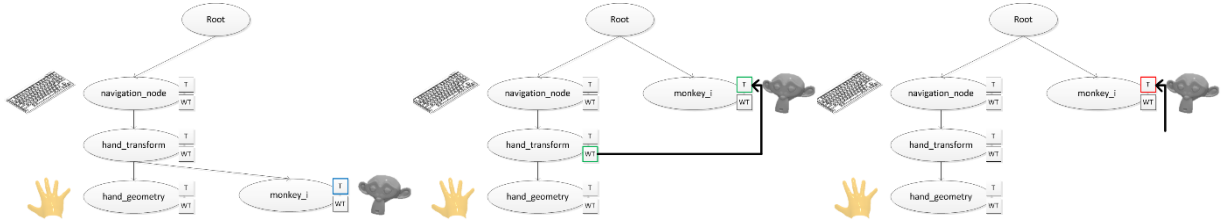
Implementation hints

Dragging methods

```
class ManipulationManager(avango.script.Script):

  def start_dragging(self):
    #...

  def object_dragging(self):
    #...

  def stop_dragging(self):
    #...
```

Comparison

Questions?



	Strategy 1	Strategy 2	Strategy 3
`start_dragging()`	• change node order in scenegraph • set node transformation	• compute offset	• store tool matrix
`object_dragging()`		• compute and set transformation	• compute and apply diff matrix • store tool matrix
`stop_dragging()`	• change node order in scenegraph • set node transformation		