



This lecture will be recorded



C O M P A S

064-0026-00L: COMPAS II

Introduction to Computational Methods for Digital  
Fabrication in Architecture

```
mesh.mesh.vertices[...]  
if callable:  
    raise Exception('Callback is not callable.')
```

```
for k in range(100):  
    mesh.vertex_coordinates(key) for k  
    mesh.vertices():  
    if key is fixed:  
        continue  
    p = key_xyz[key]  
    nbs = mesh.vertex_neighbours(key, ordered=True)  
    c = center_of_mass_polygon([key_xyz[nbr] for nbr  
    # update  
    attr = mesh.vertex[key]  
    attr['x'] += d * (c[0] - p[0])  
    attr['y'] += d * (c[1] - p[1])  
    attr['z'] += d * (c[2] - p[2])  
if callback:  
    callback(mesh, k, callback_args)
```

```
def smooth_mesh_length(mesh, lmin, lmax, fixed=None, kmax=100):  
    if callback:  
        if not callable(callback):  
            raise Exception('Callback is not callable.')
```

```
fixed = fixed or []  
fixed = set(fixed)
```

```
for k in range(kmax):
```

slides and code

*[https://](https://tiny.cc/compas-ii)**tiny.cc/compas-ii***

1100

1525

1545

Relays changed  
Started Cosine Tape (Sine cl  
Started Mult + Adder Test.



Relay #70  
(moth) in re

First actual case of bug

~~1630~~ 1630

Antangent started.

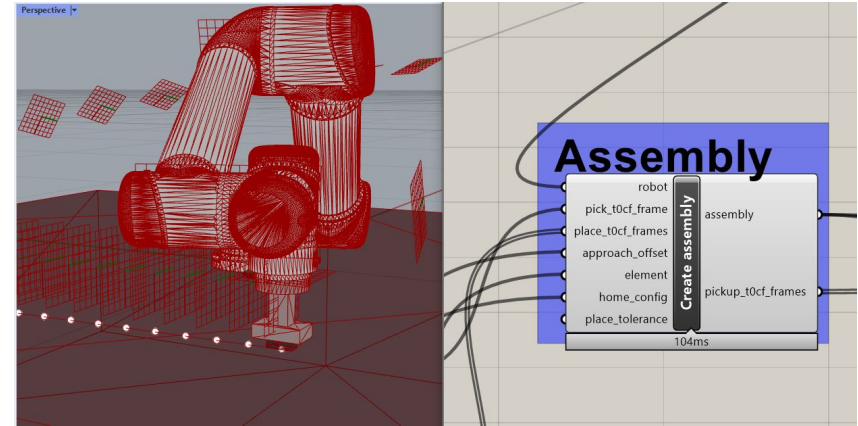
1700

closed down.

```
conda env update --file environment.yml --prune
```

# Review of last lecture assignment

- Building up on the experience of the assignment 04, explore the network-based process
- Using `07_pick_and_place_graph.ghx`, plan pickup trajectory and at least 8 elements
- Store the full assembly to a file called `assembly.json` using the provided serialization

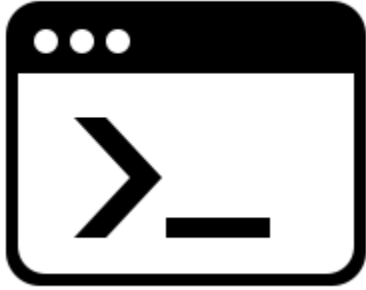


*Unclear assignment goals?*

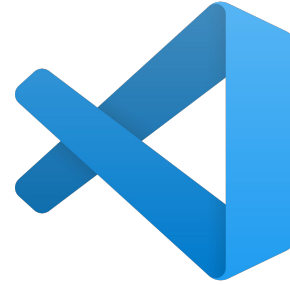
*How to show different elements of the assembly?*

Today's goal

Explore **assembly sequencing** and dissect the pick and place exercise



`docker-compose up -d`



Right-click → Compose Up

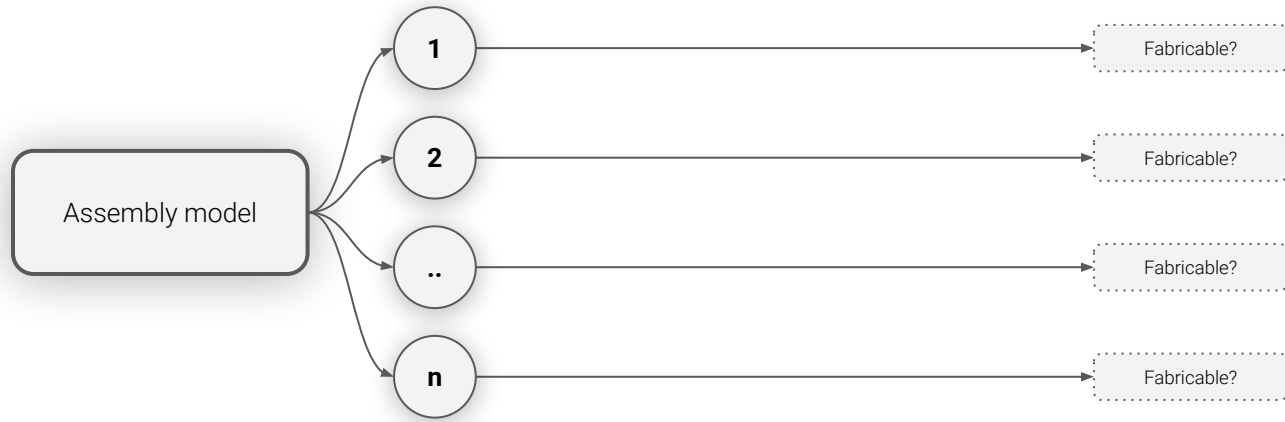


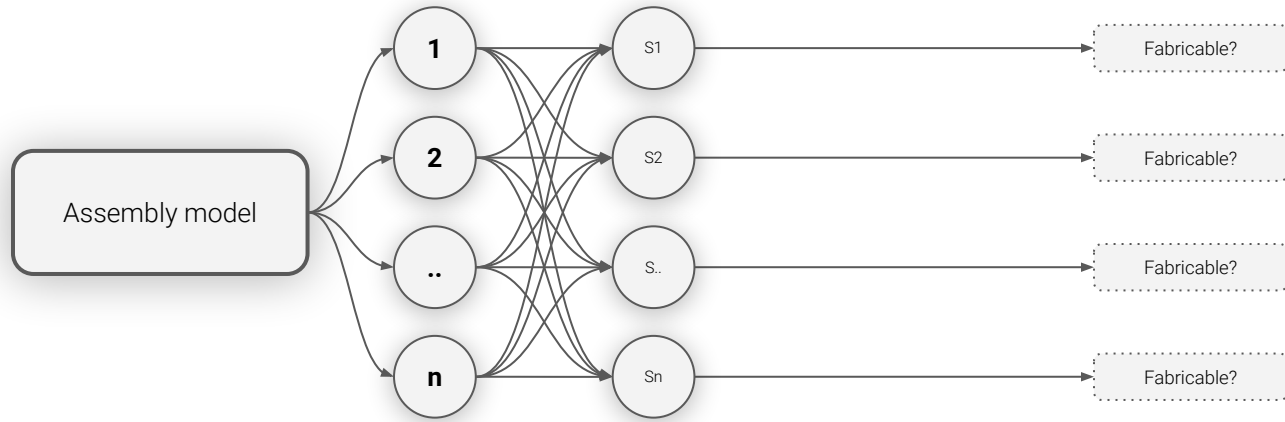
## **docker/ur5-planner**

Lightweight MoveIt UR5 planner  
without any user interface

## **docker/moveit**

MoveIt UR5 planner with user  
interface via browser (noVNC)

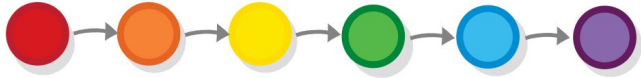




# Sequence types

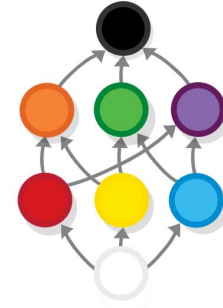
## Total orders (fully linear sequences)

- Simple to describe
- Work for simple processes



## Partial orders (e.g. dependency graph).

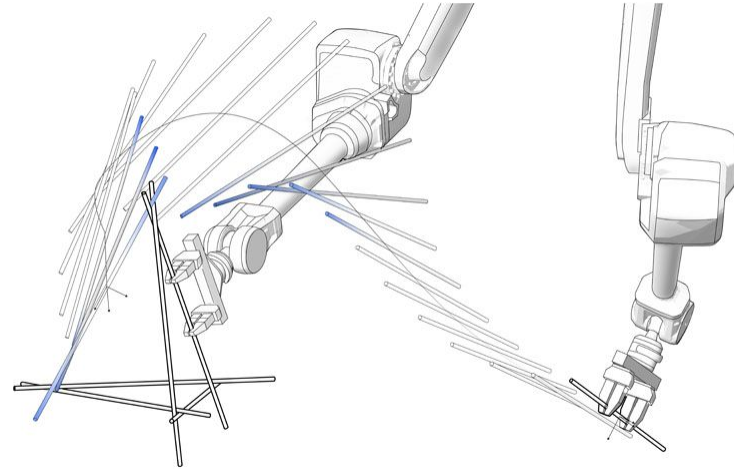
- Allow to express more advanced process (e.g. multiple robots in parallel)
- More involved to describe
- Broader selection of algorithms available



# Impact of building sequence

Sequence affects fabricability in multiple ways:

- Stability during fabrication
- Tolerance build-up
- Robotic accessibility
- Material behavior



Let's dissect the assembly exercise!

# Next week

- No assignment due next week.
- Prepare your computer for COMPAS RRC exercises:
  - Install ABB RobotStudio
  - Explore **getting started** repo: [https://github.com/compas-rrc/compas\\_rrc\\_start](https://github.com/compas-rrc/compas_rrc_start)
  - Unpack the **pack&go** file (COMPAS\_RRC\_IRB-910SC-3\_0.65.rspag)
- Ask for help if needed: Slack, Forum, Office Hours (Fridays, request via Slack)
- Next lecture:
  - Robot control with COMPAS RRC

# Thanks!

