

README for the json template:

The json template file includes transforms, locations, ovcs and constraints for the task description in the form of ROS messages. This file serves as an input for the scheduling Data in this template can be edited.

List of transforms:

For each workpiece or storage we need to have individual transform. Each transform is defined by:

- Original reference frame: **"frame_id"**: "world"
- Target reference frame: **"child_frame_id"**: "board"
- Translation of ref. frame: **"translation"**
- Rotation of ref. frame: **"rotation"**

Named list of locations:

List of locations which are used within the experiment.

- Name of the location which can be used to referencing: **"object_name"**: "loc_18"
- Timestamp when was the location recorded: **"stamp"**
- 6D pose - "pose" - **"position"** and **"orientation"**

Named list of OVCs:

An OVC models a sequence of actions to be executed at different locations by a manipulator.

Values are from an example of the OVC, where we have to "pick a bolt from a small bolt storage (from loc_1,loc_2,loc_3 or loc_4) and glue it here (loc_7) for at least 30 seconds"

- Name of the OVC: **"name"**: "ovc_1"
- Sequence of locations to be visited: **"location_names"**: [{"values": ["loc_1","loc_2",,"loc_3","loc_4"]}, {"values": ["loc_7"]}]
- Actions to be performed in individual locations: **"action_types"**: ["pick","glue"]
- Time range of the performed action in the given location **"duration_ranges"**: [{"max": 20, "min": 5}, {"max": 100, "min": 30}],
- Set of arms to be considered to execute the task: **"groups"**: ["r1_arm", "r2_arm"]
- Description of the location acquired from the language **"location_description"**: ["small bolt storage","here"],
- List of configurations of the robot for to be visited locations: **"configuration_domains"**: []

List of constraints:

Ordering constraints which should be applied to the

Values from the example: "ovc_22" starts after end of "ovc_21"/"First ovc_21 then ovc_22"/"First glue a small bolt here (ovc_21) then glue a point here (ovc_22)."

- Type of the constraint (1 - StartsAfterEnd): **"ct_type"**: {"val": 1}
- Name of the constraint: **"name"**: "ct_23"
- List of OVCs to which to apply the constraint: **"ovc_name"**: ["ovc_21", "ovc_22"]

- Locations within OVC to which the constraint is applied (-1 - whole ovc, which means that the whole ovc_21 has to first end and then ovc_22 can start; [0,3] would mean that the 4th location within ovc_22 can be visited after 1st location ovc_21 was visited):
"ovc_interval": [-1, -1]
- Additional parameters for the constraint - e.g. time gap (first ...then within 5 seconds...):
"params": []