Explainability in Robotics: The Green Button Challenge

Freek Stulp, Adrian S. Bauer, Samuel Bustamante,
Florian S. Lay, Peter Schmaus, Daniel Leidner

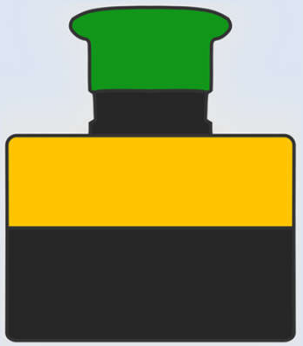
Institute of Robotics and Mechatronics
German Aerospace Center (DLR)

DLR

Green Button Challenge

The Challenge

- Every robot has a green button
  - First press: *What are you doing?*
  - Second press: *And why?*



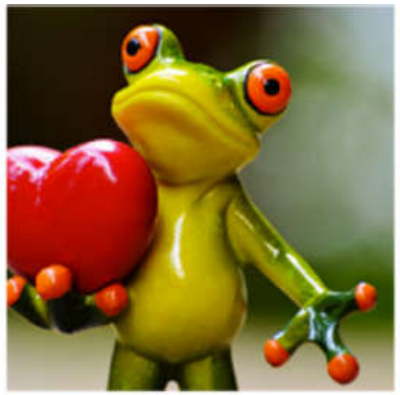
- Mid-term aim
  - Make it a global challenge!
  - Public outreach
  - Strengthen ties to KR community

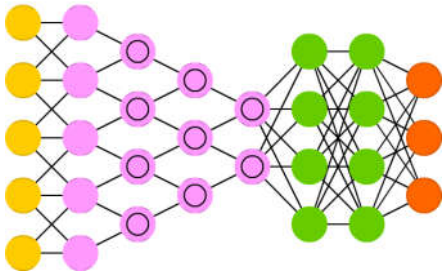
The Hackathon Competition

- 1-week “hackathon”, 5 teams
- Domains: space, healthcare, assembly
- Demo tour

https://www.dlr.de/rm/en/desktopdefault.aspx/tabid-3755/17612_read-63005/

Explainability and Deep Learning

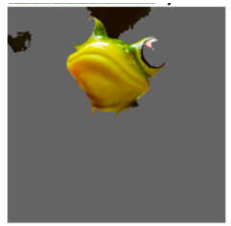


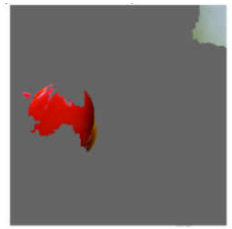


p=0.54

p=0.07

p=0.05

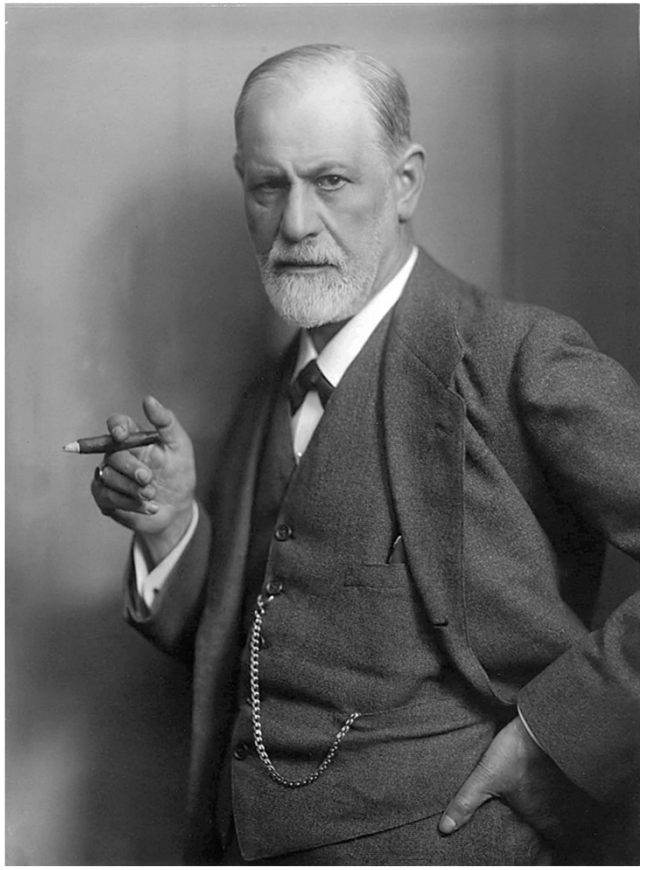




M. Ribeiro, S. Singh, C. Guestrin. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. KDD 2016.
https://www.oreilly.com/learning/introduction-to-local-interpretable-model-agnostic-explanations-lime

Post-hoc: learn first, explain later

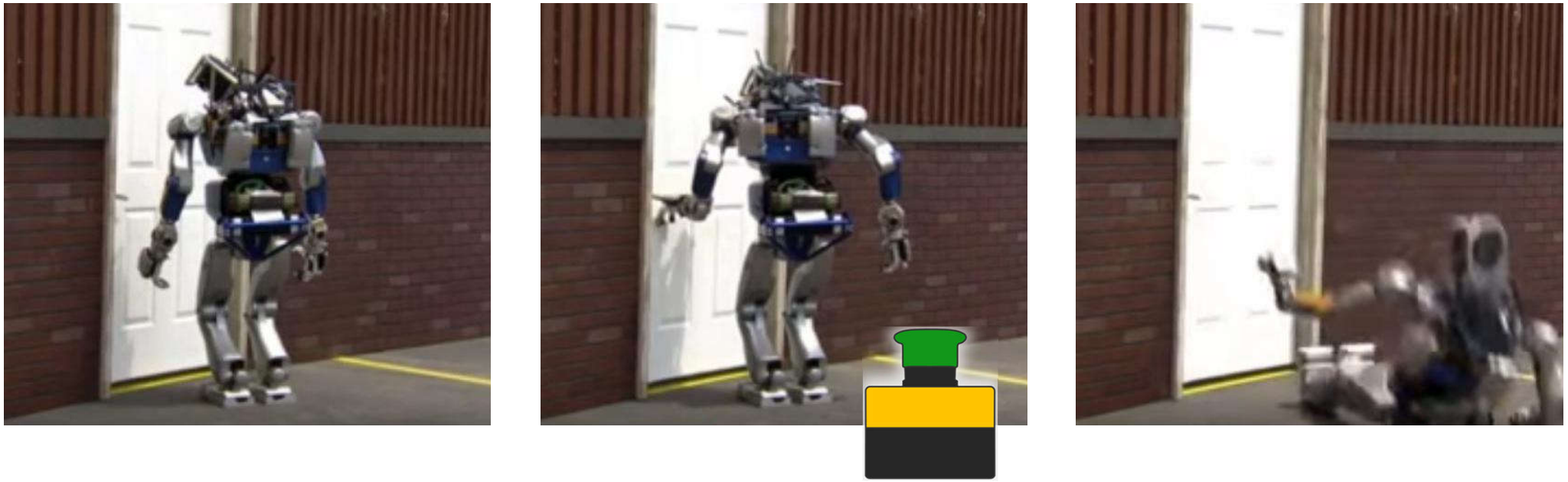
Sigmund Freud



“Psychotherapy for deep networks”

Explainability and Robotic Planning

DARPA Robotics Challenge



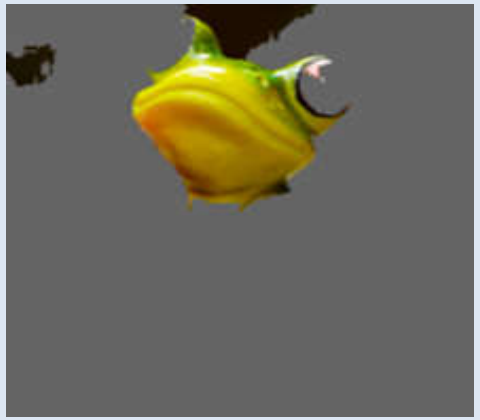
We can explain what the robot is doing and why due to prior knowledge
Ergo, robots need prior knowledge to explain their behavior

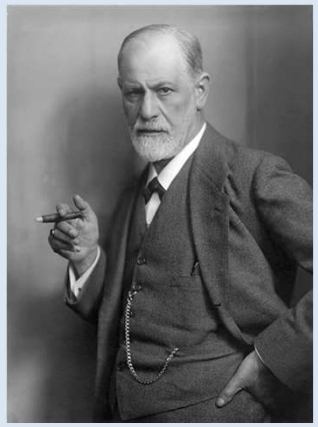
Here, the issue is not that knowledge is implicit,
but that developers do not share their explicit prior knowledge with the robot.

Explainability: Freudian vs. Wittgenstenian

Explainability in Deep Learning

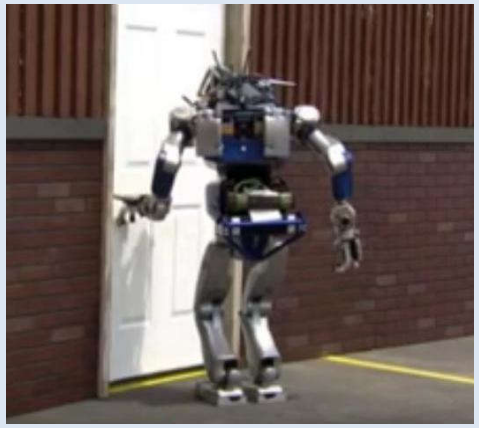
- Make the implicit explicit
  - Post-hoc: learn first, explain later
  - “Psychotherapy for deep networks”





Explainability in Robotic Planning

- Make the explicit explicit
  - KR: human → robot
  - Interpretability: robot → human



“Definitions are rules for the translation of one language into another. Every correct symbolism must be translatable into every other according to such rules.''
Proposition 3.343 of Wittgenstein's *Tractatus Logico-Philosophicus (1921)*

DLR

Motivation for our participation in XLoKR

Stronger ties to KR community!

Up next: Highlight the solution of the winning team at the internal challenge

What‘s in it for Robotics?

- Faster, more flexible planners
- KR with ontologies
- Improved formalization of explainability

What‘s in it for KR?

Robotics provides (cool) applications with high societal and economic impact

Factory of the Future

DLR

Action Templates for Hybrid Robotic Planning

Action Template: _wiper.collect

Symbolic Representation

```
'''
(:action _wiper.collect:
   :parameters (?t - _tool ?m - _medium
                ?s - _surface ?a1 - _manipulator)
   :precondition (and (picked ?t ?a1)
                      (applied ?s ?m))
   :effect (and (collected ?m ?s))
)
'''
```

Geometric Representation

```
def collect(self, medium, surface, manip):

   path = self.compute_task_trajectory("collect", surface, medium)
   initial_config = robot.get_configuration()
   initial_frame = path[0]

   operations = [
      ("plan_to_frame", manip, self.grasp_frame, initial_frame),
      ("set_stiffness", manip, self.tcp, self.stiffness),
      ("set_force", manip, self.tcp, self.force),
      ("follow_task_motion", manip, path, self.grasp_frame),
      ("plan_to_config", initial_config),
   ]
   return operations
```

hybrid planning

- symbolic planning (PDDL - fast downward planner)
- geometric planning (motion planning)

Leidner, Daniel (2017) Cogntive Reasoning for Compliant Robot Manipulation. Dissertation. **Winner of the George Giralt PhD Award 2018 (best European PhD thesis in robotics)**

Leidner, Daniel und Bartels, Georg und Bejjani, Wissam und Albu-Schäffer, Alin und Beetz, Michael (2018) Cognition-enabled robotic wiping: Representation, planning, execution, and interpretation. Robotics and Autonomous Systems.

Action Templates: Grounding

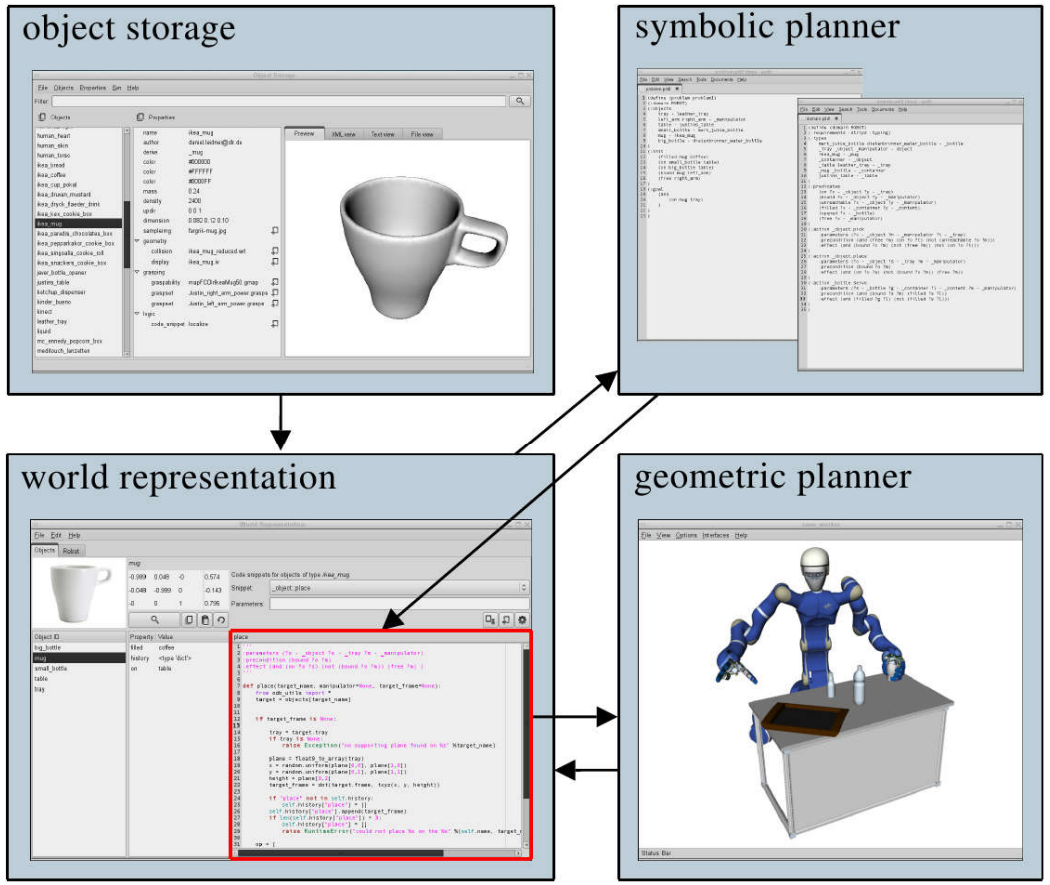
Action Template: _wiper.collect

Symbolic Representation

```
'''
(:action _wiper.collect:
   :parameters (?t - _tool ?m - _medium
                ?s - _surface ?a1 - _manipulator)
   :precondition (and (picked ?t ?a1)
                      (applied ?s ?m))
   :effect (and (collected ?m ?s))
)
'''
```

Geometric Representation

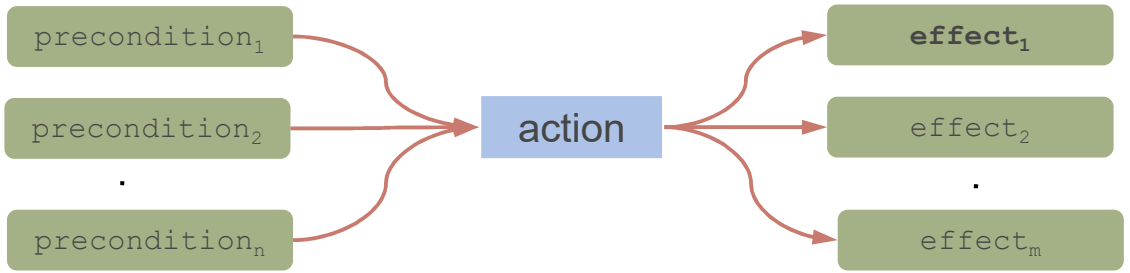
```
def collect(self, medium, surface, manip):

   path = self.compute_task_trajectory("collect", surface, medium)
   initial_config = robot.get_configuration()
   initial_frame = path[0]

   operations = [
      ("plan_to_frame", manip, self.grasp_frame, initial_frame),
      ("set_stiffness", manip, self.tcp, self.stiffness),
      ("set_force", manip, self.tcp, self.force),
      ("follow_task_motion", manip, path, self.grasp_frame),
      ("plan_to_config", initial_config),
   ]
   return operations
```



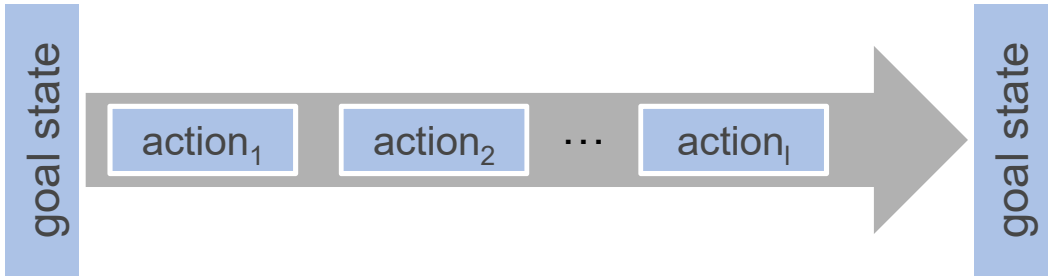
Up next: explain the resulting symbolic plan

Symbolic Actions and Plans

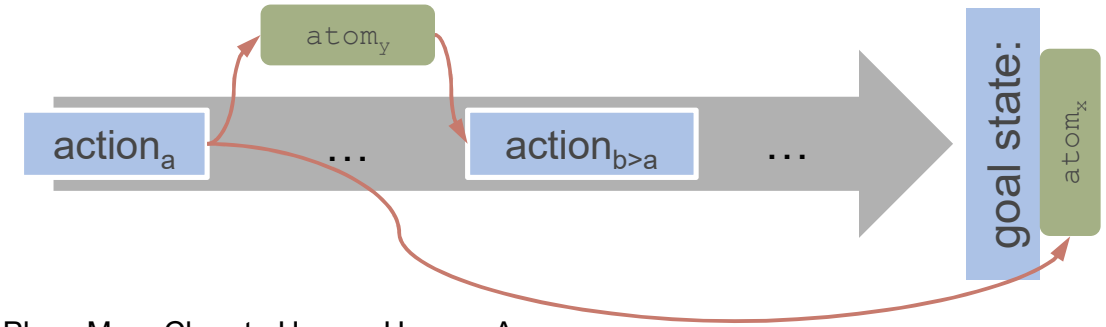
- **Actions** defined in action templates (ATs):
  - header in PDDL: preconditions, effects, parameters
  - body: geometric grounding to robot operations
- **Goal state**
  - conjunction of atoms
  - e.g. `(and (clean panel1) (stored wiper left_holster))`



- **Plan *P***
  - ordered sequence of actions
  - Initial state → *P* → goal state



- **Theorem (inspired by [1])**
  Every action in a plan without superfluous actions has at least one effect that
  1. is a precondition of a later action or
  2. is part of the goal state.



[1] B. Seegebarth, F. Müller, B. Schattenberg, and S. Biundo, “Making Hybrid Plans More Clear to Human Users – A Formal Approach for Generating Sound Explanations”, *Int. Conf. Automated Planning and Scheduling*, 2012

Creating a Causal Graph

P. Bercher, S. Biundo, T. Geier, T. Hoernle, F. Nothdurft, F. Richter, and B. Schattenberger, “Plan, Repair, Execute, Explain – How Planning Helps To Assemble your Home Theater”, in *Proc. 24th Int. Conf. Automated Planning and Scheduling*, 2014, pp. 386-394

T. Chakraborti, K. Fadnis, K. Talamadupula, M. Dholakia, B. Srivastava, J. Kephart, and R. Bellamy, “Planning and visualization for a smart meeting room assistant”, *AI Communications*, vol. 32, no. 1, 2019, pp. 91-99

S. Kambhampati, “Mapping and Retrieval During Plan Reuse: A Validation Structure Based Approach”, in *Proc. 8th Nat. Conf. Artificial Intelligence*, vol. 1, 1990, pp. 170-175

M. Veloso, *“Learning by Analogical Reasoning in General Problem Solving”*, PhD Thesis, 1992

initial state

locked panel1

Stored wiper left_holster

unlock panel1 left_arm

unlocked panel1

rotate panel1 right_arm

rotated panel1

lock panel1 left_arm

locked panel1

grasp wiper left_holster

bound wiper left_arm

free left_holster

clean panel1 wiper left_arm

store wiper left_holster

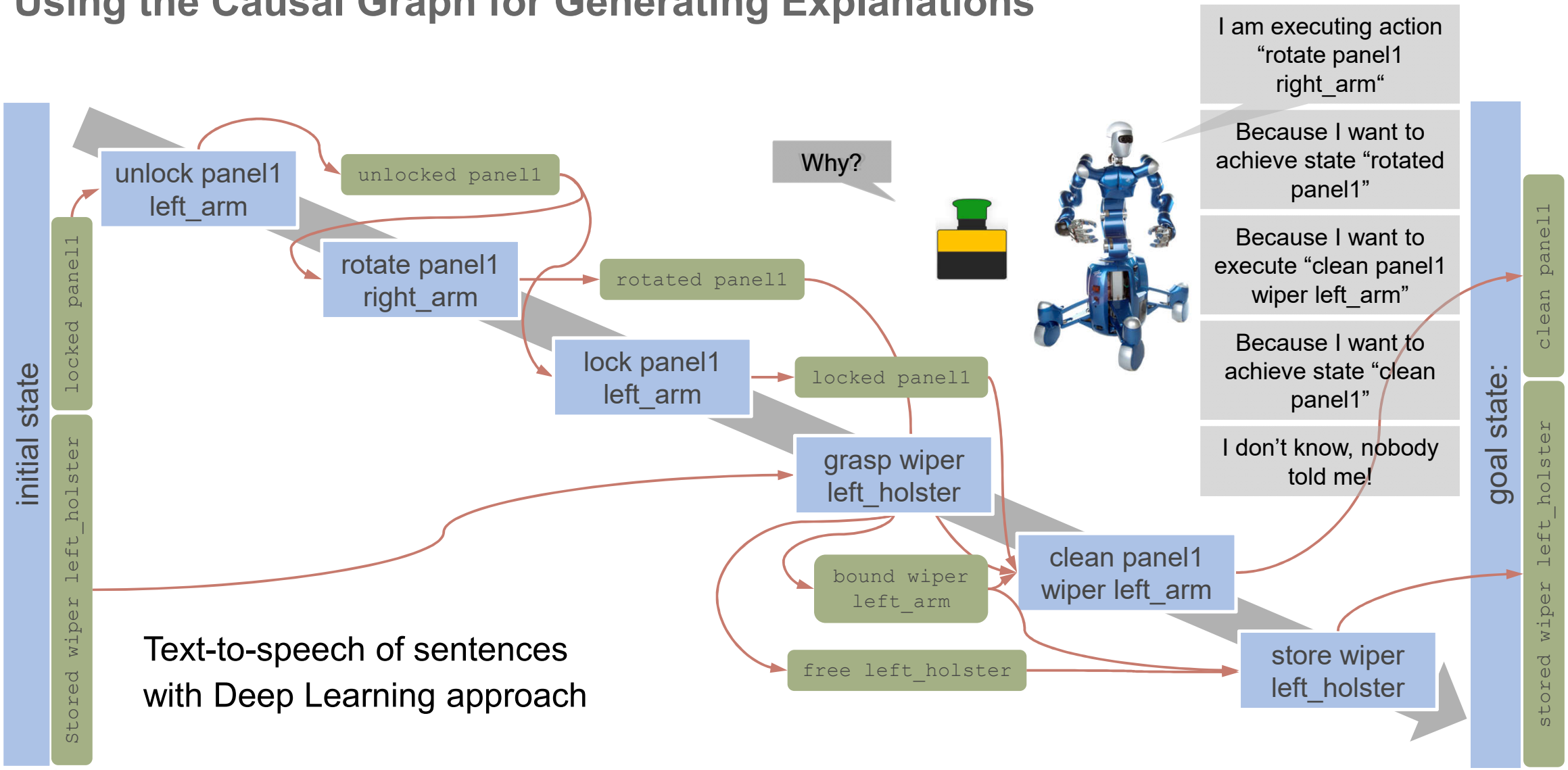
goal state:

clean panel1

stored wiper left_holster

DLR

Using the Causal Graph for Generating Explanations

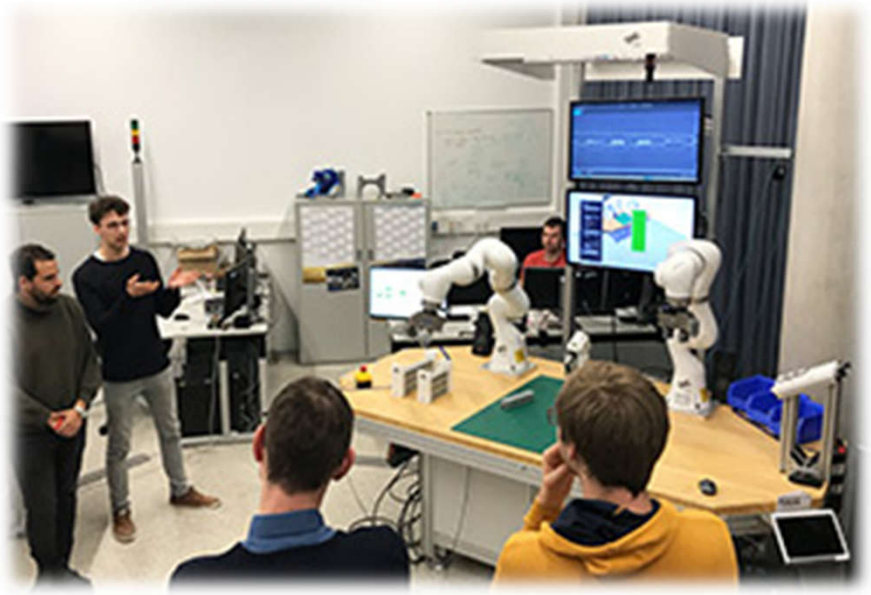


Text-to-speech of sentences
with Deep Learning approach

DLR

Results of other teams

- Explainability in (Hierarchical) State Machines
- Enabling inquiries about specific concepts: ontologies
- “Mumbling”: spontaneous explanations
- Using Deep Learning for text-to-speech
- Making the physical Green Button



Main results after one week

- Raising awareness for challenges in explainability
- Design patterns for explainability of complex systems
- Fun!

Conclusion

Aim: Stronger ties between KR and robotics communities

- Planners, ontologies, formalization
- Interested in working with (*or at!*) our institute? → zoom chat or email

What's in it for KR?

Robotics provides (cool) applications with high societal and economic impact