Project acronym:	ACAT
Project Type:	STREP
Project Title:	Learning and Execution of Action Categories - Update
Contract Number:	600578
Starting Date:	01-03-2012
Ending Date:	28-02-2015

Deliverable number:	D5.6
Deliverable Title:	Performance index evaluation
Type (Internal, Restricted, Public):	PU
Authors:	D. Chrysostomou, C. Schou, R. S. Andersen, O. Madsen, M. Beetz, D. Nyga, G. Lisca, D. Vitkutė- Adžgauskienė, M. Tamošiūnaitė, F. Wörgötter
Contributing Partners:	AAU, UoB, VDU, UGOE

Contractual Date of Delivery to the EC:	29-02-2016
Actual Date of Delivery to the EC:	21-04-2016
Delay agreed by the Commission	

1 Content

1. Executive summary

This document presents an overview of the benchmarking efforts of the ACAT system and the evaluation of the key performance indicators proposed before in the Deliverable 5.2. First, we present briefly the demonstrator scenarios along the lines of which the ACAT system performance is to be measured. Then, we report on the evaluation results of the three categories of key performance indicators, which are now oriented as end-user, language and research benchmarks. Finally, we provide the analysis and results of the Human Interaction trials we performed in the industrial scenario.

2 Introduction

This document presents the evaluation of the benchmarks and key performance indicators for the ACAT system. The goal of this document is to report on the performance of the ACAT system for both examined scenarios according to the key performance indicators. For each of these indicators the method of measuring has already been described in the updated version of D5.2 and a description of two demonstrator scenarios and related instruction sheets is presented in D5.1. These scenarios form the main benchmarks for the entire ACAT system. As these scenarios are thoroughly described in D5.1 and their respective KPIs in D5.2 only a brief overview will be presented in this document for the sake of completion.

3 Main Benchmark Scenarios

In ACAT two main demonstrators form the main benchmarks of the ACAT system. These two scenarios, IASSES and CHEMLAB, are thoroughly described in D5.1, but will be briefly presented in this section, too.

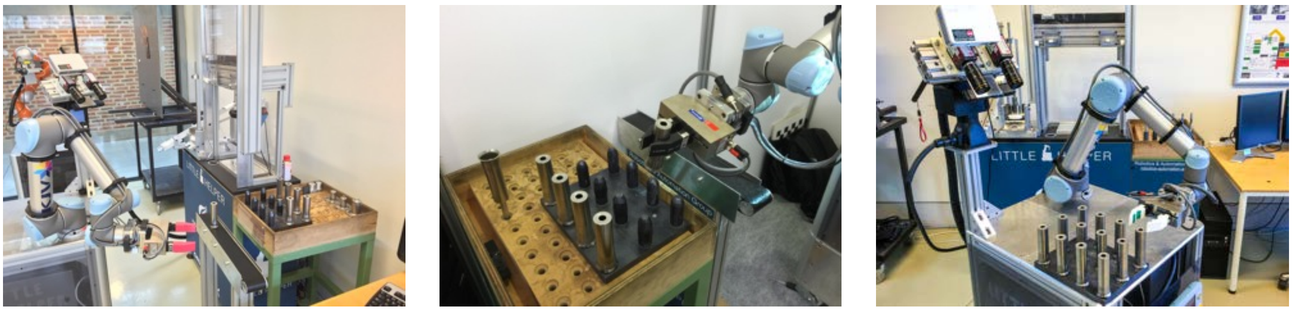
3.1 IASSES scenario

The IASSES scenario will focus on manufacturing tasks from the production of rotors for submersible pumps at the SQ-factory at the Danish company Grundfos. The production environment from Grundfos and thus the selected tasks will be replicated at Aalborg University. Two benchmarks were performed in relation to the IASSES scenario. This scenario is relevant in industrial settings and will be used to perform “end-user oriented” benchmarking.

3.1.1 Rotor cap collection benchmark

The goal of this benchmark is to pick a cylindrical component called a rotorcap from a conveyor belt or the robot platform and place it on a fixture on the robot. In the course of the project, three versions of this benchmark were established. The first one (Figure 1a) was established for the M24 demonstration in the premises of Aalborg University. The second version (Figure 1b) was set up for the demonstration of the ACAT system during the IROS’15 conference which held during M30 of the project. Finally, the last version of this benchmark (Figure 1c) was used to measure the degree of human intervention by performing Human Robot Interaction trials with 12 participants in AAU in M36 of the project.

The instruction sheet of all variations is similar and fairly simple in order to focus on execution speed instead of ambiguity during setup and execution of the task. Therefore, the instruction sheet contains one sentence: “Pick up rotorcap from conveyor (or platform) and place it on fixture”. The parsing of this instruction delivers a list of 3 skills; PickUp, MoveTo, and Place.



(a) used in M24 demonstration of the project (b) used in IROS exhibition in M30 (c) used for HRI tests in M36

Figure 1: Different versions of the Rotor Cap Collection benchmark.

3.1.2 Rotor Assembly Benchmark

In the rotor assembly benchmark the rotor for the electrical motor of a SQFlex submersible pump is assembled from the components shown in Figure 2.

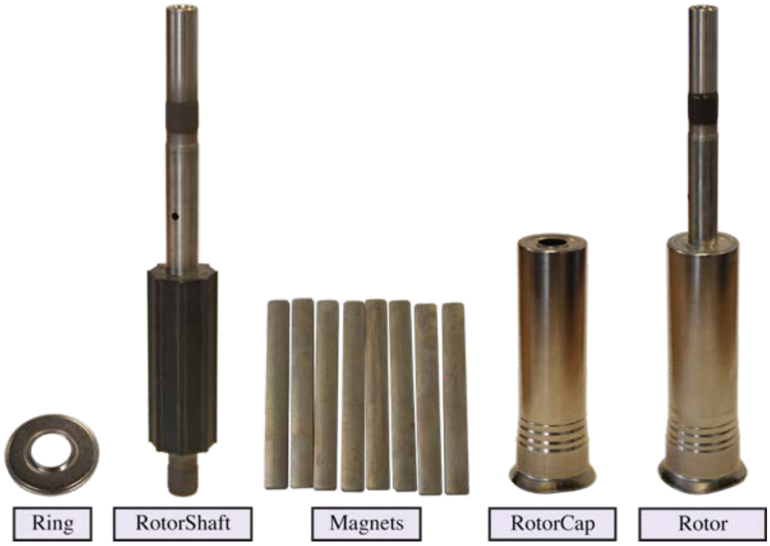


Figure 2: Overview of the components used in the assembly of the SQFlex rotor.

The task is carried out at a workstation containing a hydraulic press, see Figure 3. The main structure of this workstation, along with auxiliary feeders made to assist the robot execution, was also constructed as a replica at Aalborg University, see Figure 3.

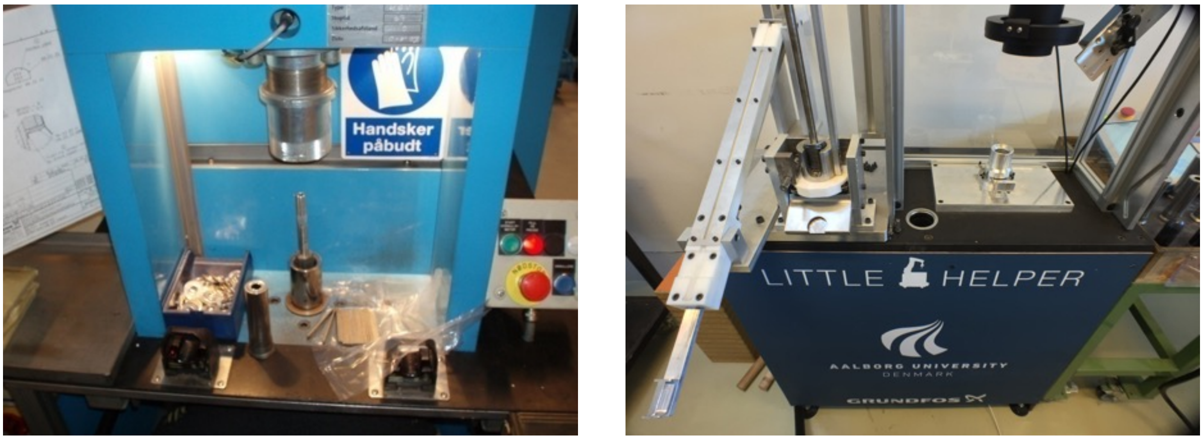


Figure 3: Left: Workstation containing a hydraulic press at which the SQFlex rotor is assembled at Grundfos. Right: Replication of this workstation at Aalborg University.

The manual task of assembling the rotor at Grundfos is carried out as follows:

1. The pressure ring is mounted onto the rotor shaft before it is placed into the press' fixture.
2. Eight magnets are placed on the sides of the rotor shaft. These magnets must be correctly oriented and aligned with the octagonal shape of the core (part of the rotor shaft) in order to fit into the fixture.
3. The rotor cap is placed on top of the fixture with the rotor axle sticking through the center hole. Due to limited clearance above the rotor axle the rotor cap must be tilted when placed on the rotor axle.
4. The press is then activated.

5. Afterwards the pressed rotor is removed from the press. Again the clearance above the pressed rotor is limited, and the rotor must be tilted in order to be removed.
6. The pressed rotor is placed in a moveable fixture plate which holds a number of units.

These instructions are written for human operators accompanied by explanatory images. Naturally, the majority of them require bimanual manipulation of the objects as well. Therefore, we adapted the instruction sheet to a robot-friendly version as follows:

1. Press button to release pressure ring from ring dispenser.
2. Pick up pressure ring and drop it in press tube.
3. Take rotor axle from trolley and insert it into press tube.
4. Move slider to release magnet from magnet dispenser.
5. Do step 4 for 7 more times.
6. Pick magnet and insert into press between axel and rim of press tube.
7. Turn axel by 45 deg.
8. Repeat steps 6 and 7 for 7 more times.
9. Take rotor cap from fixture and put over the rotor axel.
10. Slide cover down to closed position.
11. Slide cover up to open it.
12. Take (finished) rotor and put it into box.

Note: Due to its industrial nature, the task of rotor assembly was the final target of the integration and benchmarking efforts of the ACAT system. The CHEMLAB scenario is much wider open and only more general benchmarks are given (see below).

3.2 CHEMLAB Scenario

The selected scenario is the process of DNA extraction from a sample. The process involves the handling of liquids (pouring, decanting, etc.) and usage of standard laboratory equipment such as jars of different size and shape, filter cartridges, and a centrifuge. In order to be successful, the process has to be executed under the required constraints (temperature, time schedule, etc.) stated in the respective lab protocol. This scenario addresses basic research. Thus, research oriented benchmarks will be defined for it.

Success can be validated easily: the result of the successful process is a visible DNA pellet. All sub-processes involved either have an intermediate result which can be defined precisely or it can be observed directly if the sequence of actions is executed appropriately (e.g., with the required amounts of substances and according to the time constraints).

Our research focus, however, is not on the success of physical process itself. Our focus is rather on the planning, reasoning, and knowledge representation problems that have to be solved in order to enable the robot to master this particular task as well as other related ones. Therefore, it is not sufficient to define the benchmark in terms of the question if the robot finally happens to extract some DNA, or not. The benchmark should explicitly reflect if the robot succeeds because of its ability to *understand the content* of the instructions given in the lab protocol, to combine these typically underspecified and vague information with appropriate *background knowledge* (both domain-specific and commonsense), and to *reason* on the basis of this integrated knowledge.

The CHEMLAB scenario is implemented at University of Bremen on a PR2 robot, see Figure 4. The implementation on the robot platform demonstrates if the plans derived from the vaguely formulated, under-specified instructions are sufficient for successful execution in a real-world environment, and how strong the impact of the level of abstractness in the plans on the robot's performance is.

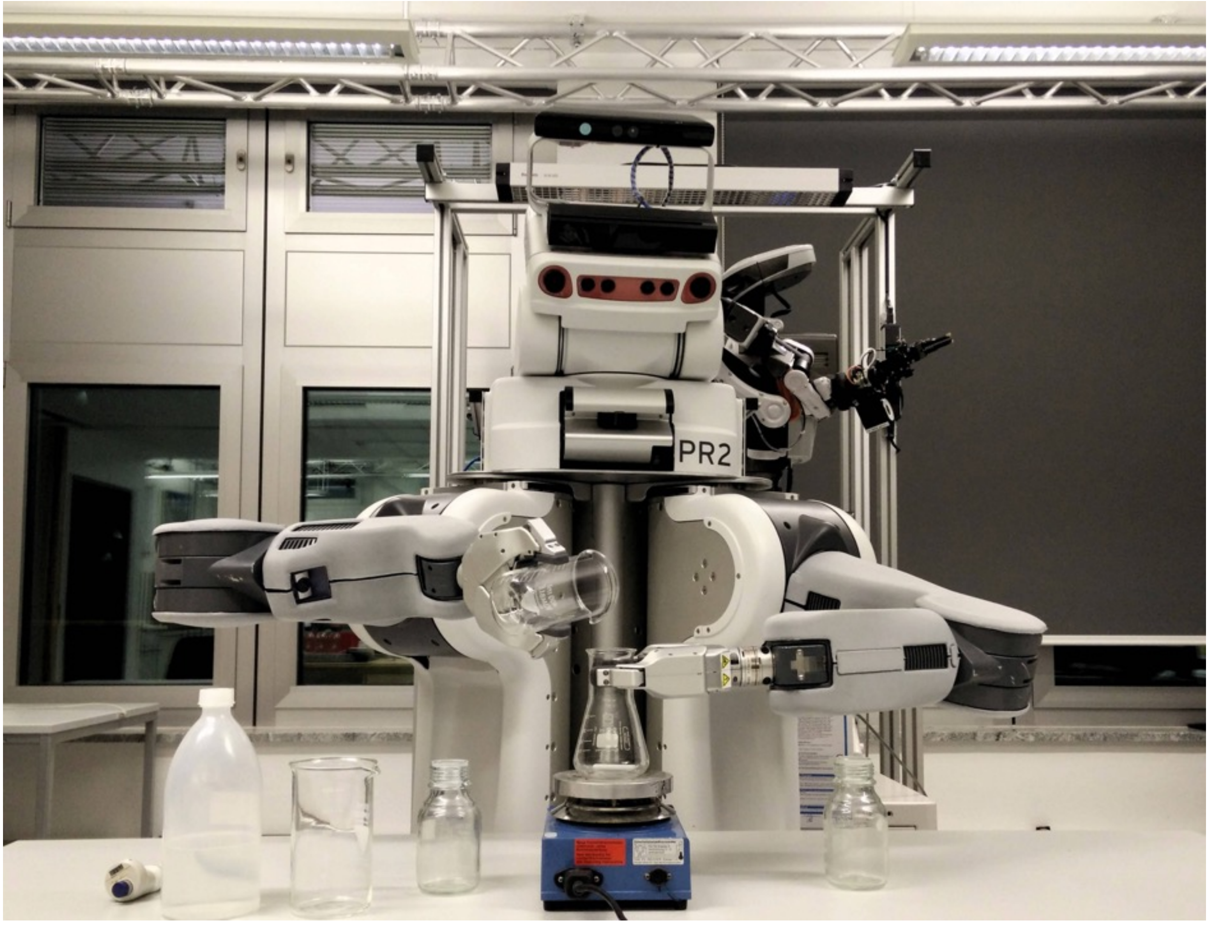


Figure 4: The PR2 robot at University of Bremen that is used for the CHEMLAB scenario.

4 Key Performance Indicators

Benchmark indicators have been divided into (1) End-User-oriented, (2) Language-oriented and (3) Research-oriented benchmarks. Number 1 addresses aspects that could be of interest to potential end-users of this system. This addresses also the reviewers' comments from the M24 review. Number 2 is concerned with the ontology and 3 with other more general aspects.

5 Overall System – End-User-oriented Benchmarks

As we discussed in previous deliverables, in IASSES scenario, we executed variations of the "Rotor cap Collection" benchmark. These variances enable us to extract useful information and specific insights in order to evaluate the overall performance of the system in terms of domain complexity and human intervention. In the following section we will describe the details of each variation and evaluate its performance.

5.1 Evaluation of End-User-oriented KPIs

All three versions of the "Rotor cap Collection" benchmark have things in common such as the robot platform (LH4), the mockup conveyor or robot platform and a fixture available to place the industrial part as a last part of the benchmark. However, they have key differences as well. The first version incorporated a motion and grasp planner and the first prototype of an ADT translator. It addressed mainly the Key Performance Indicators #1.1(a-d) and #1.2.

Key Performance Indicator 1.1(a) - Setup time for execution of a known task

Description Total setup time for an instruction with an existing ADT. This is the variation with the least complexity during the execution.

Measurement Feed the ACAT system a new instruction and measure the setup time until task is ready for execution (measured in seconds).

Evaluation This KPI was evaluated extensively during the M24 demo using the instruction "Take rotor cap from conveyor and put it on the fixture". This instruction is similar to the "Pick up rotor cap from conveyor and place it on a fixture" since the main, primary and secondary objects are the same. Therefore, the system uses a stored ADT where all the details of the task were already taught including action chunks and visual and grasp poses of the rotor cap, the conveyor and the fixture. Thus, the setup time is minimum and the user is able to setup the system and start the execution in only **5 seconds**.

The execution part takes only 20 seconds as this video indicates https://goo.gl/OhZnPc.

Specifically, for the evaluation of this KPI an experienced robot operator performed the task for 100 times. The deviation in the setup time was ±2 seconds and in the execution ±3 seconds.

Key Performance Indicator 1.1(b) - Setup time for execution of a semi-known task with unreliable pose inputs

Description Total setup time for an instruction with an existing ADT but with re-calculation of the poses for some of the objects.

Measurement Feed the ACAT system a new instruction and measure the setup time until task is ready for execution (measured in seconds).

Evaluation This KPI was evaluated using the instruction "Take rotor cap from conveyor and put it on the fixture". In this case the system is able to re-use the main, primary and secondary objects of the instruction and the related action chunks. However, the operator can choose to re-calculate the pose and grasp poses in order to verify their applicability. This re-calculation incorporates the use of a motion and grasp planner thus the movement of the robot is slower. Besides, the vision system requires extra time to perform another pose estimation of the object and the operator needs to teach a new place location.

Therefore, the setup time is increased to **111 seconds** as the video indicates https://goo.gl/imWW0C.

The task was performed 100 times. The deviation in the setup time was ±10 seconds due to the delays for the estimation of new object visual and grasp poses and specification of a new place location. The deviation of the execution was also 25±5 seconds.

Key Performance Indicator 1.1(c) - Setup time for execution of a semi-known task with unreliable grasp inputs

Description	Total setup time for an instruction with an existing ADT but with re-calculation of the grasp poses for some of the objects.
Measurement	Feed the ACAT system a new instruction and measure the setup time until task is ready for execution (measured in seconds).
Evaluation	This KPI was evaluated using the instruction "Take rotor cap from *table* and put it on the fixture". In this case the system is able to identify that the main and secondary objects of the instruction are the same and pair an existing ADT from the instruction "pick up rotor cap from conveyor and place it on the fixture". However, because the primary object is different, the existing visual and grasp poses were unreliable. As a result, the vision system requires extra time to perform another pose and grasp estimation of the object located in the new primary object and the operator needs to teach a new place location after that. Therefore, the setup time needed **126 seconds** as this video indicates https://goo.gl/vVFq83. The task performed another 100 times. A small number of setup efforts required 10 seconds less depending on the outcome of the motion and grasp planner and the availability of a new feasible trajectory. However, there were cases where the setup of the task required 160 – 180 seconds due to the delays of the extra calculations. Regardless the delays in the setup time of the task the deviation of the execution of the task was 40±5 seconds.

Key Performance Indicator 1.1(d) - Setup time for execution of a semi-known task with similar model information

Description	Total setup time for an instruction where only the model similarity is used from the ADTs.
Measurement	Feed the ACAT system a new instruction and measure the setup time until task is ready for execution (measured in seconds)
Evaluation	This KPI was evaluated using the instruction "Take *metal bottle* from conveyor and put it on the *table*". In this case the system is able to identify that both the main and secondary objects of the instruction are different from the ones in existing ADTs. As a result, the existing visual and grasp poses were unreliable. As in the 1.1(c), the vision system requires extra time to perform another pose estimation of the object and the operator needs to teach a new place location. Therefore, the setup time needed **141 seconds** as the video shows https://goo.gl/OYYWL4. The task performed another 100 times. Similarly, a small number of setup efforts required significantly less time (-15 seconds) depending on the outcome of the motion and grasp planner. However, there were cases where the setup of the task required up to 2 minutes more in total due to the delays of the extra calculations and the extra restrictions introduced by the motion planner. The deviation of the execution of the task was 50±10 seconds.

Key Performance Indicator 1.2 - Robustness during setup

Description	Robustness of the system when processing a new instruction sheet.
Measurement	Feed the ACAT system multiple new (unknown) instruction sheets and measure the percentage of successful established task sequences. A task sequence is successful if it achieves the specified goal of the task (end state).
Evaluation	This KPI was evaluated during the setup of the various instantiations of the instruction sheet presented before. As a new instruction we count the cases where either the main, primary or secondary objects were other than the ones in the stored ADTs. Therefore, the user needed to respond to many pop-up messages in the monitor and take decisions on how to proceed with the setup of the task sequence. In the cases where all the parameters of an instruction were unknown then the operator could feel confused and make mistakes. Hence, from a total of 300 trials the operator set up the task sequence successfully **95%** of the time.

The second variation of the Rotor Cap Collection benchmark, utilised more advanced vision modules along with force-adaptation systems and a second prototype of the IASSES ADT translator. These additions improved the general performance of the system. Besides, the participation in an exhibition (where the robot needed to execute tasks for hours without falling into errors) provided to us the opportunity to gather data related to the robustness of the system during execution and specifically address KPIs #1.3 and #1.4.

Key Performance Indicator 1.3 - Robustness during execution

Description	Robustness of the execution phase
Measurement	During multiple executions of a task instantiated from an instruction sheet, measure the successful completions of the task.
Evaluation	During the participation in IROS exhibition we had the opportunity to set up the system and execute the same task for a prolonged period of time and major part of the duration of the exhibition. In total of 5 days we executed the same task continuously for almost 4 hours per day. Naturally, due to overload of the systems in the fair premises, there were hardware and systematic issues that caused breakdowns of the workstation (e.g., loss of power, loss of internet connection). However, these problems are not related to the robustness of the ACAT system and the average execution time for one repetition of the task was almost 60 seconds as we can notice in the video (https://goo.gl/e1mg5I). As a result, we can state that the robustness of the ACAT system is close to 99% since it executed the same task successfully for about 1190/1200 times.

Key Performance Indicator 1.4 - Cycle time during execution

Description	The cycle time of the task during execution
Measurement	During multiple executions of a task instantiated from an instruction sheet, measure the mean cycle time. This could be compared to other task instantiation methods.
Evaluation	As mean cycle time in ACAT we define the combined time needed for specification, teaching and execution of a task. Since we use an ADT translator we are able to re-use data from stored ADTs and produce the specifications for the list of required skills. In case the task is new then the operator has to teach some parameters of the skills. Considering the new vision modules and the adaptation force – based methods we have integrated in the ACAT platform, large portion of the cycle time is spent in communication between systems. As a result, the mean cycle time in ACAT is **240 seconds** when the main, primary and secondary

objects are known and 4 skills are generated for an advanced pick and place task. Compared to the task instantiation method used in the project TAPAS (were the main integrator partner, AAU, was using the previous generation of Little Helper) with manual selection and specification of skills, the mean cycle time for a simple pick and place task with two skills was 191 seconds and for an advanced pick and place task with 5 skills was 400 seconds. So, we can safely conclude that we benefit significantly from the application of the ACAT system.

The final version of this benchmark was used to assist the Human Robot Interaction (HRI) trials and to provide results in terms of human intervention with the ACAT system. The performance of the system is analyzed further below and via the KPIs #1.5 – 1.7.

We had the opportunity to have 12 participants in the HRI test of the ACAT system by using the Little Helper robot and the Skill-based System as a core communication server between the ACAT sub-systems.

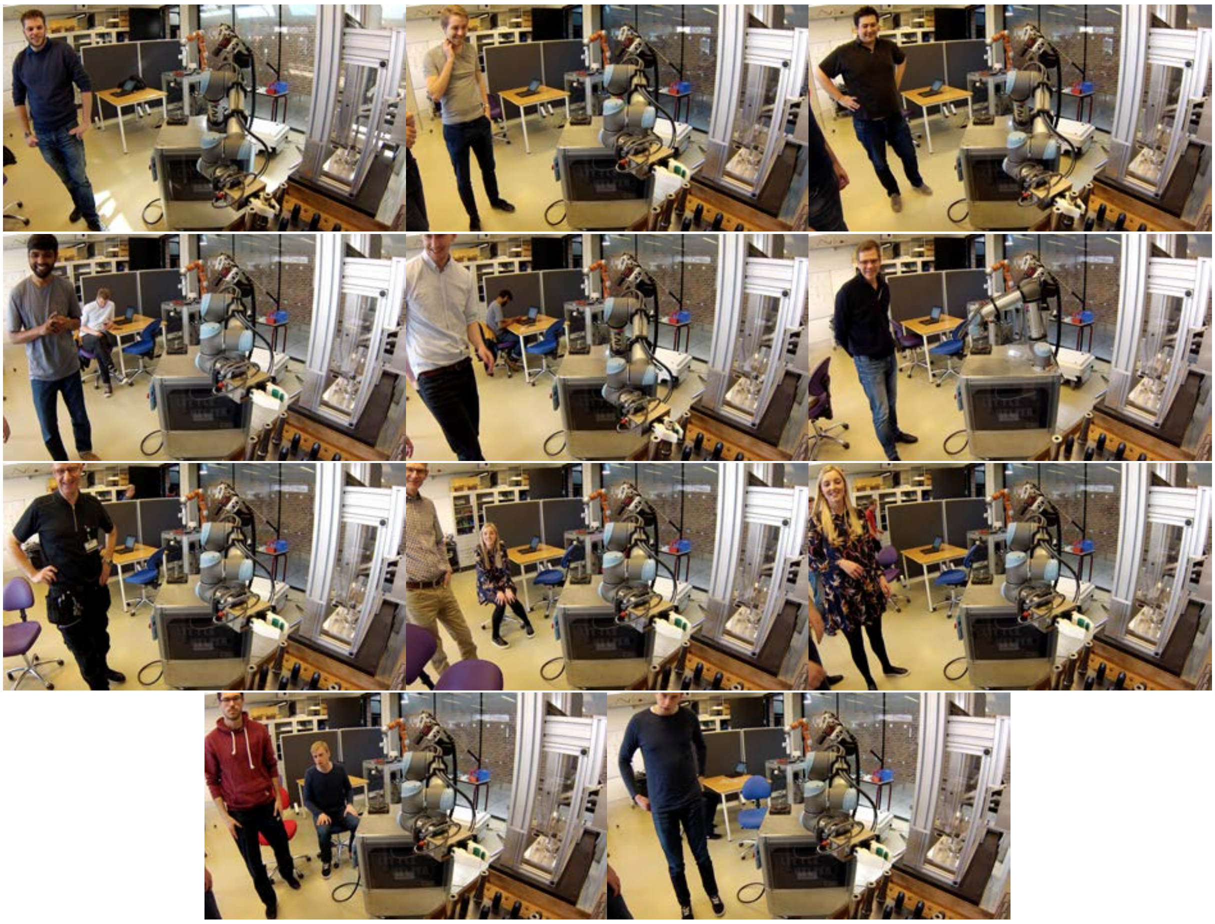


Figure 5. Eleven of the twelve participants in the ACAT HRI test

Description of the HRI experiment

In this experiment, users were asked to program an advanced pick and place task by writing an instruction and putting together a sequence of skills that encapsulates expert robotics knowledge. A graphical user interface was used to setup a task as a sequence of skills instantiated previously from the translation of an

ADT. Kinaesthetic teaching was used for further parameterization of coordinate frames necessary for the execution of the skills. The skills employed for assembly tasks are different variations of pick and place skills. The main purpose was to obtain feedback from various users, according to the KPIs, and to assess how well they comprehended and operated the system.

Evaluation scenario

The scenario consisted of the task of Rotor Cap Collection using the instruction “Pick up rotorcap from platform and place it on the fixture”. The users performed the test individually and were each given a 4:30 minutes introduction, including the basic concept of the project, the concept behind skills, the kinesthetic teaching, the graphical user interface and the concept of written instructions and ADTs. In the setup phase, the users worked independently, but an expert was available nearby to provide assistance when requested by the operator. All 12 persons were instructed to setup the task and sequence each skill using the steps in the GUI as shown in Figure 6. After providing the instruction and sequencing the skills, they enter the teaching phase, where the users kinesthetically move the robot arm and tool to positions instructed by the GUI. Three of the ACAT sub-systems, the background ADT translator service, the visual pose estimation and the adaptation with DMPs, play a crucial role for the successful execution of the taught task.

Timeline

The experiment was conducted over the course of two days. A general 4:30-minute introduction was given to all participants, providing an overview of the experiment. After this, each participant had the time to perform the test in his/her own pace. During this time, the following took place:

1. A 4-5-minute introduction to the system and task was given. The GUI and the whole teaching sequence were explained.
2. The participants provided the instruction that would initiate the task.
3. The participants taught the robot kinesthetically and, by the assistance of the sub-systems and the GUI messages, they completed the task.
4. The participants filled out a short questionnaire about the experiments (See Appendix).

Evaluation parameters

For each participant we logged the usability measures according to the ISO 9241-11 (1998) defined as efficiency, effectiveness and satisfaction. In ACAT we use similar measures such as training time needed, demonstration efficiency and a questionnaire for measure the satisfaction (see Appendix).

1. **Efficiency-** *”The spent resources in relation to the accuracy and completeness with which users achieve specified goals”.*
   Measured objectively as the time spent to complete the task and is addressed in ACAT with KPI #1.5.
2. **Effectiveness** - *"The accuracy and completeness with which users achieve specified goals”.*
   Measured objectively as the number of times where the participant required help to complete a task. In ACAT we monitor it by KPI #1.6.
3. **Satisfaction -** *”The freedom from discomfort and positive attitudes towards the use of the product”.*

Measured subjectively through the Lewis' ASQ questionnaire and an option to provide additional comments/suggestions. The Lewis' ASQ evaluates satisfaction with three Likert-scale question (evaluated from 1-7 and N/A):

a. Overall, I am satisfied with the ease of completing the tasks in this scenario.
b. Overall, I am satisfied with the amount of time it requires to complete the tasks in this scenario
c. Overall, I am satisfied with the support information when completing the tasks

In the HRI test of ACAT we asked all the participants to provide answers to a questionnaire (see Appendix) so we can form a conclusion regarding satisfaction and user-friendliness (KPI #1.7).

Results

Key Performance Indicator 1.5 – Training time required

Description	Operators with different levels of robot expertise will try to setup the robot for the *Rotor Cap Collection* benchmark. Non-robot experts, robot experts unfamiliar with ACAT and robot experts within the ACAT consortium will be asked to setup the robot after a brief training session.
Measurement	The training time elapsed for the setup of a known task will be measured in seconds.
Evaluation	As we can see in the analysis of the questionnaire (see Appendix) the participants had different levels of expertise ranging from fundamental awareness to recognized authority. All of them received the same introduction to the system in the beginning that lasted 04:30 minutes. Naturally, the required time to complete the same task was varying as well in relation to the robot expertise. The shortest time to complete the task was 223 seconds (03:43 mins) from a user with Expert knowledge and the longest was 448 seconds (07:28 mins) from someone with basic knowledge of robotics. The average time for all participants was 313 seconds (05:13 mins).

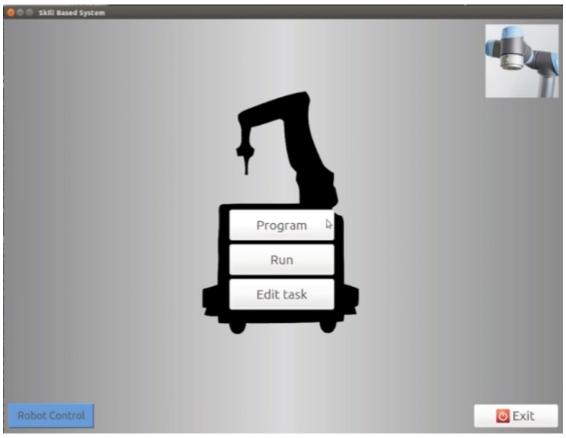
Key Performance Indicator 1.6 – Demonstration efficiency

Description	How many times the operators asked for help or guidelines during the setup of the robot after the training
Measurement	Ratio of successful setup efforts and number of extra guidelines required.
Evaluation	All the test users were allowed to ask for help while programming the task. As an average of all participants they asked 1.8 times for extra guidance during the programming of the task. As a ratio we can state that the demonstration was 67% efficient since all participants managed to set up the task successfully but still asked some extra guidance.

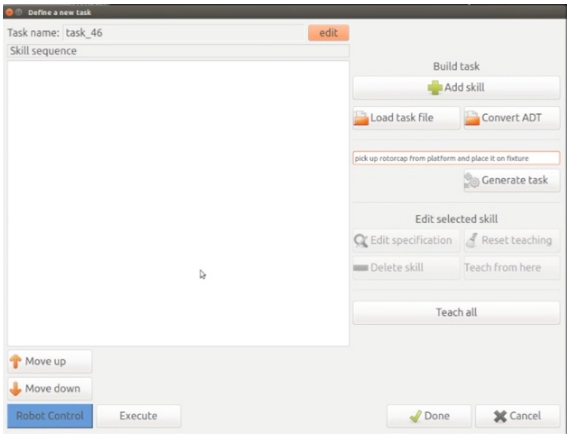
Key Performance Indicator 1.7 – Robot expertise (user friendliness)

Description	The combination of training time, the setup time and the demonstration efficiency define the degree of robot expertise required from the operator in order to use the demonstrator successfully.
Measurement	Measured as a total indicator of user-friendliness of the robot
Evaluation	Results from the questionnaire reveal that the participants generally found the approach quite intuitive. Especially the ability to write an instruction and accelerate the setup of the task along with the easy-to-use GUI. The vast majority of users were satisfied with the ease

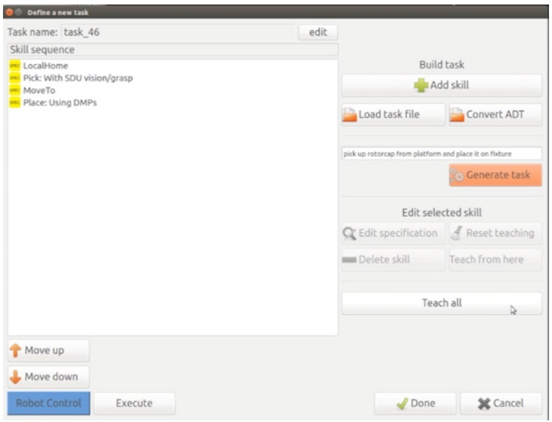
of completing the task of the test giving a grade of 6.5 in a scale from 1 to 7. Moreover, they found the amount of time needed to complete the task also satisfactory since they rate 75% of them rated with 7. Overall, they were also very satisfied with the support information shown on the screen during the execution since 92% of them graded with 6.5 out of 7. According to the general comments we can verify that there were some minor details that caused confusion during the setup but this fact did not discourage the participants to conclude that the system is quite intuitive and that they clearly prefer to program it through text instructions instead of manually selecting the necessary skills.



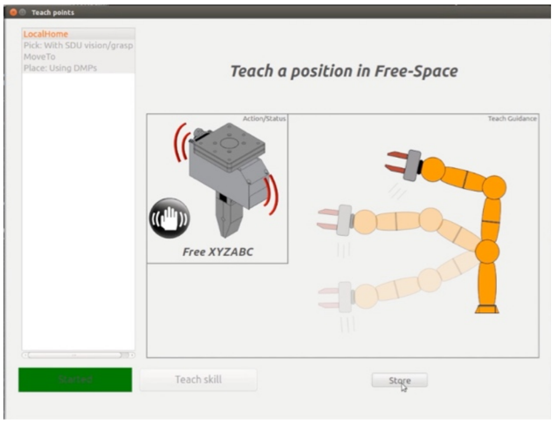
(a) Launch screen of the minimal user interface



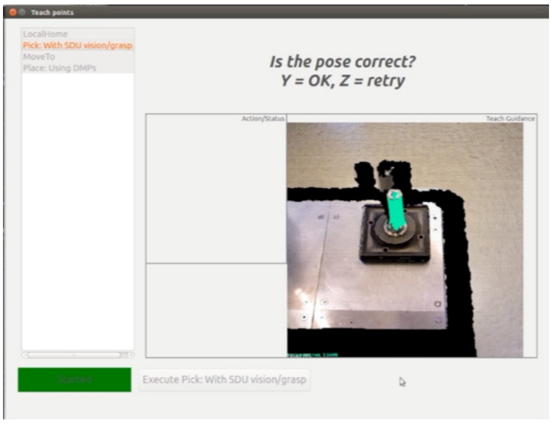
(b) user provides written instruction (in this case this instruction is new to the system)



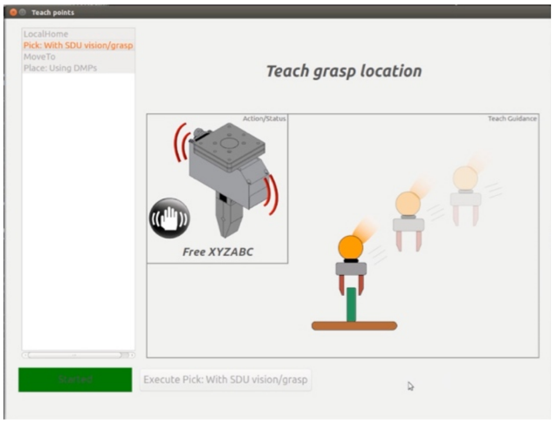
(c) By "generating the task", a list of skills is provided which has a yellow symbol meaning that their parameters are not yet known



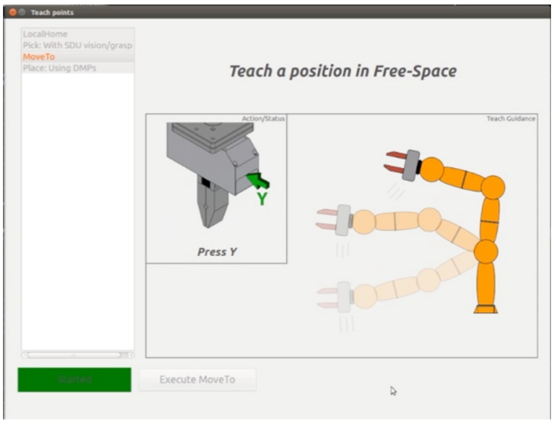
(d) Teaching the "LocalHome" skill



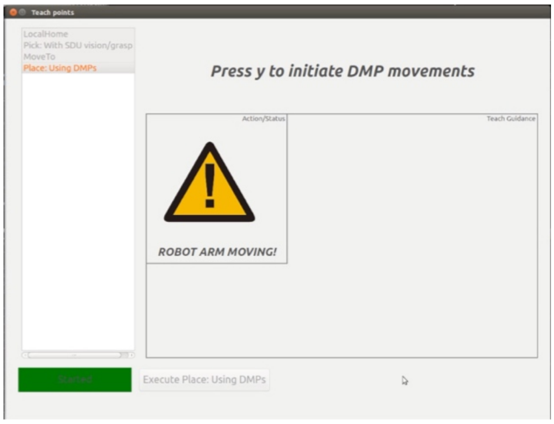
(e) The vision system finds the main object of the instruction



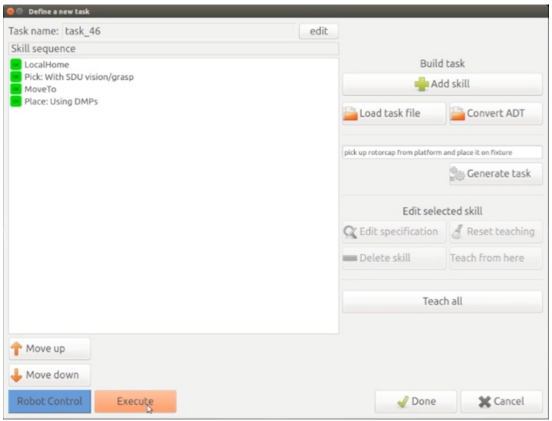
(f) Teaching the "Pick" skill



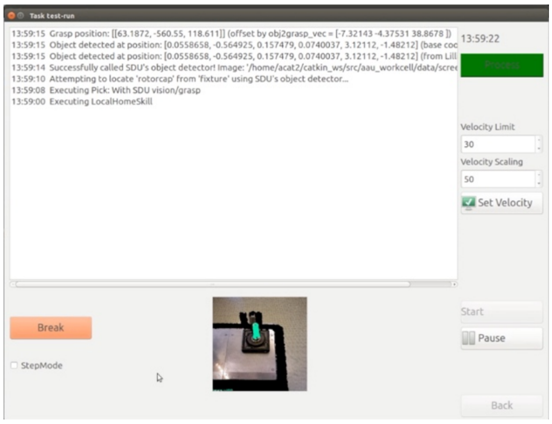
(g) Teaching the “MoveTo” skill



(h) Teaching the last, “place” skill using DMPs for the last part



(i) After teaching the necessary set of skills, the user can now execute the taught task by using the same instruction which will provide the same set of skills with a green symbol next to them



(j) During execution the pose of the object is estimated again to verify that the saved poses are still usable.

Figure 6: Interface screen shots

6 Process Memory Formation – Language-oriented benchmarks.

6.1 Evaluation of Language-oriented KPIs

KPI 2.1	*Name*	**Linguistic action ontology**
	Description	Number of action verbs in the ontology and number of synsets.
	Measurement	Determined by the number of action verbs and synsets available in the process memory by the end of the project.
	Evaluation	Ontology contains 322 action verbs, organized into 189 synsets.

KPI 2.2	*Name*	**Object categories**
	Description	Number of object categories saved in the process memory

	Measurement	Determined by the number of object categories in the linguistic object ontology and the number of associated object images/models available to aid recognition and robotic manipulation of those objects.
	Evaluation	Ontology contains 304 object categories (synsets). 66 object categories has associated images/models.

KPI 2.3	***Name***	**Number of action grounding instances**
	Description	Number of robot execution/control instances stored in the process memory
	Measurement	Determined by the number of robot execution/control instances stored in the process memory by the end of the project.
	Evaluation	53 ADTs

KPI 2.4	***Name***	**Action categories**
	Description	Number of action categories saved in the process memory
	Measurement	Determined by the number of action categories available in the process memory by the end of the project.
	Evaluation	24 action categories are defined in the ontology.

7 Knowledge and Information Content – Basic Research-Oriented Benchmarks

As already stated in Section 4.4 of the Deliverable D5.2 (Benchmark and performance index definition), we measure the performance of cognition-enabled robot control systems in terms of their knowledge and information content.

Illustration

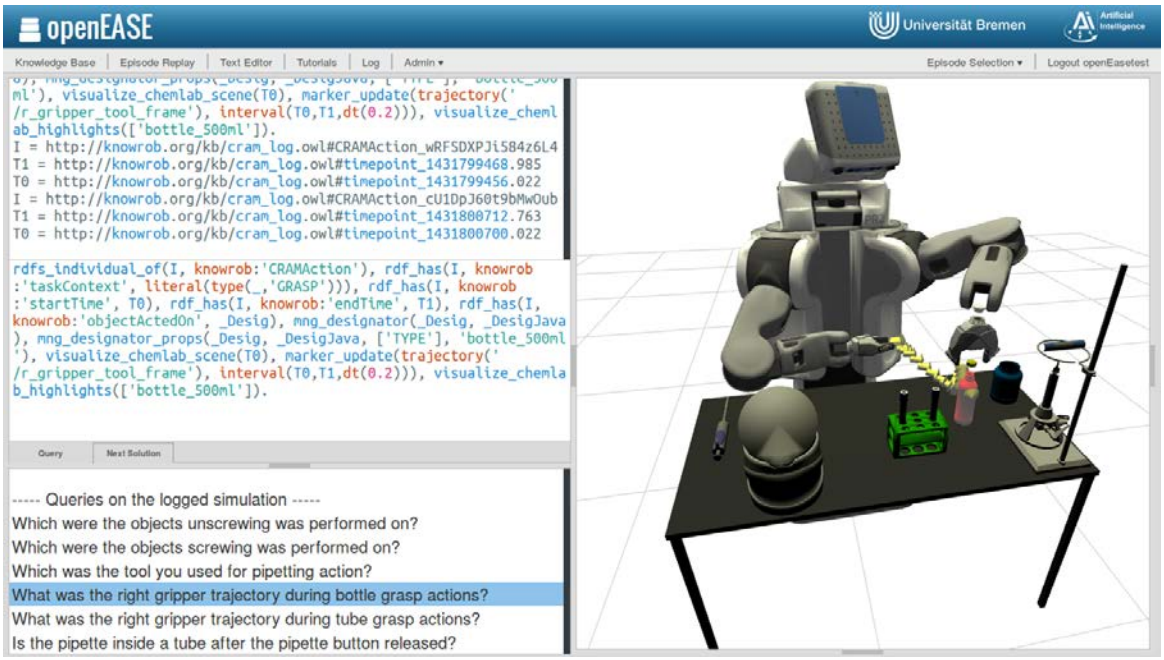


Figure 7: Screenshot of the openEASE framework with an exemplary trajectory query

We developed a publicly available web-based service[1] allowing the user to query the system to test certain capabilities such as (1) the competent interpretation of vaguely, ambiguously, and incompletely formulated tasks, (2) the successful answering of queries regarding what the robot has done, how, and why, and what it is capable of accomplishing, (3) the robot's ability to answer queries about the lab environment, it is to operate in, and the equipment it uses, (4) the ability to understand scenes and the ability to form memories and process models of the environment and (5) the ability to answer queries about the expected consequences of actions depending on the action parameterizations and the contexts they are executed in.

In particular, the system can be used to generate ADTs from simulations, experience and natural language, as well as to query them. Figure 7 shows an example query in the openEASE system in which the queried trajectory of the gripper is highlighted yellow in the visualization on the right.

Listing 1 shows the example query executed in Illustration. The code snippet will cause the system to calculate the trajectory of the right gripper (r_gripper_tool_frame) in an interval from startTime to endTime during a GRASP action operated on a bottle_500ml.

The system is capable of dealing with vaguely formulated and underspecified natural-language instructions such as *'Add some water to the purine.'* or *'Add some arsenic_acid to the imidazole.'*

```
rdfs_individual_of(I, knowrob:'CRAMAction'),
rdf_has(I, knowrob:'taskContext',
literal(type(_,'GRASP'))),
rdf_has(I, knowrob:'startTime', T0),
rdf_has(I, knowrob:'endTime', T1),
rdf_has(I, knowrob:'objectActedOn', _Desig),
mng_designator(_Desig, _DesigJava),
mng_designator_props(_Desig, _DesigJava, ['TYPE'], 'bottle_500ml'),
visualize_chemlab_scene(T0),
marker_update(trajectory('/r_gripper_tool_frame'),
interval(T0,T1,dt(0.2))),
visualize_chemlab_highlights(['bottle_500ml']).
```

Listing 1: Prolog query for computing and visualizing a trajectory of the grippers

Given an instruction missing certain objects or roles which are necessary to execute the task, the system is furthermore able to infer those information pieces and complete the instruction accordingly. For example, the information about the amount in the instruction *'Add 5 drops of the lysergic_acid to the pyrimidine.'* will cause the system to assume, that a pipette is most likely an appropriate object to serve as the *instrument*, which is not explicitly provided in the instruction.

7.1 Benchmarking Probabilistic Knowledge Bases for Instruction Interpretation

Additionally, the system is capable of inferring the correct meanings of ambiguous words by using a similarity measure based on the taxonomic relations between word concepts. It is therefore possible to generate executable plans from instructions containing words the system has not faced before.

1 http://open-ease.org

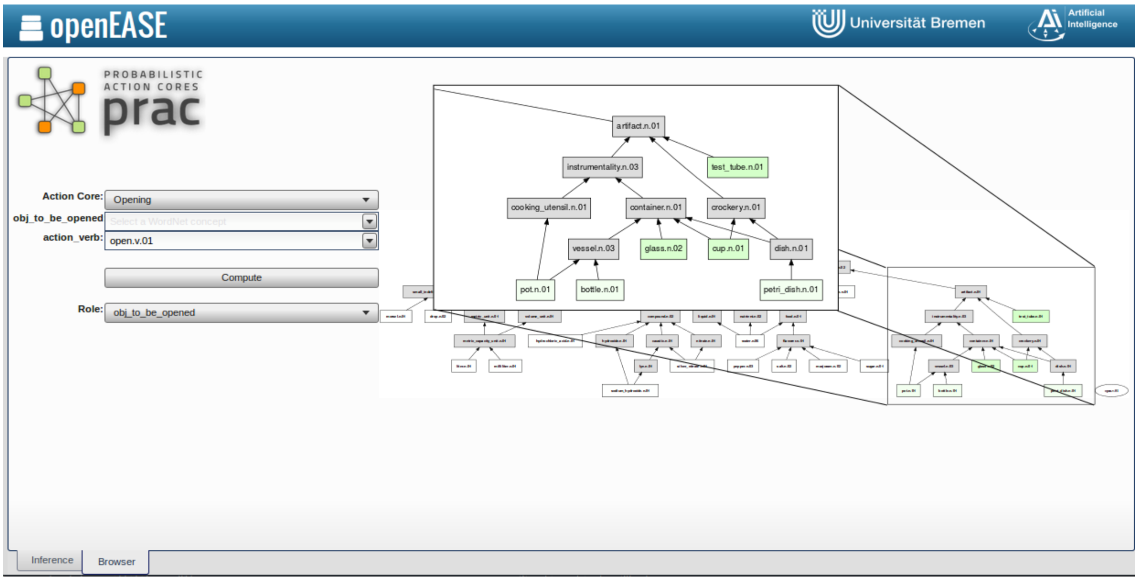


Figure 8: PRAC Browser result for querying openable objects in the knowledge base

Another benchmarking tool we developed within the scope of the ACAT project is the PRAC Browser[2]. For each action core, the user can select *WordNet*[3] concepts for each of the roles that need to be specified to be able to perform the respective action. The system will compute the distribution for each role that has not been specified and visualize it in a graph.

The illustration above (Fig. 8) shows such a distribution highlighting the relevant sub-graph for the missing role *obj_to_be_opened* in an *Opening* action providing only the role *action_verb*.

Table 1: (Optimistic) estimate of the number of instructions that PRAC is able to interpret.

Action Core	Action Roles	#training examples	#sister terms	#total combinations
Filling/Pouring	Stuff	11	242	31,460
	Goal		130	
Adding	Theme	22	444	31,968
	Goal		72	
Flavoring	Goal	9	246	12,054
	Spice		49	
Preheat	ObjToBeHeated	3	62	124

2 http://prac.open-ease.org
3 https://wordnet.princeton.edu/

	Temperature		2	
Neutralizing	Neutralizer	8	85	17,643,450
	Neutralizee		85	
	Amount		66	
	Unit		37	
Adding	Theme	16	99	23,934,042
	Goal		99	
	Amount		66	
	Unit		37	
Opening	ObjToBeOpened	3	17	17
Starting	ObjToBeStarted	1	28	28

It is difficult to estimate the number of possible situation that can be represented using this formal notation. However, one can compute a combinatorial extrapolation of this number for the currently existing action cores. The table above (Table 1) shows an estimate of how many instructions PRAC is able to interpret. It is optimistic in the way that the number of total combinations is a product over the number of sister terms of the respective action core, of which not necessarily all are useful. Still, the number of total possible combinations is disproportionately larger than the number of used training examples, which gives an indication of the generality of the system.

8 Conclusions

In this document we evaluate the performance of the ACAT system and its demonstrators according to previously specified key performance indicators. It is apparent from the results in the end-user oriented benchmark that the system is intuitive and fast to use even for non-robot experts with minimal previous robotic experience. The robustness during numerous executions is outstanding and the cycle time of the task is significantly shorter that previous generations of a similar robot system.

9 Appendix: Questionaire and Results

ACAT Questionnaire

We will not collect data that makes it possible for outsiders to identify the participant. All data is kept anonymous to any third party.

* Required

Questions related to GENERAL SATISFACTION

1. **Overall, I am satisfied with the EASE OF COMPLETING the tasks in this scenario** *
Mark only one oval.

	1	2	3	4	5	6	7	
Strongly disagree	⬭	⬭	⬭	⬭	⬭	⬭	⬭	Strongly agree

2. **Overall, I am satisfied with the AMOUNT OF TIME it took to complete the tasks in this scenario** *
Mark only one oval.

	1	2	3	4	5	6	7	
Strongly disagree	⬭	⬭	⬭	⬭	⬭	⬭	⬭	Strongly agree

3. **Overall, I am satisfied with the SUPPORT INFORMATION when completing the tasks** *
Mark only one oval.

	1	2	3	4	5	6	7	
Strongly disagree	⬭	⬭	⬭	⬭	⬭	⬭	⬭	Strongly agree

4. **Comments for the general satisfaction**

....

....

....

....

....

Questions related to SPECIFIC PARTS of the learning process

5. **The robot can be programmed through TEXT INSTRUCTIONS and by MANUAL SELECTION OF SKILLS. Which method would you prefer when programming the robot? ***
Mark only one oval.

	1	2	3	4	5	6	7	
Clearly prefer MANUAL SELECTION OF SKILLS	⬭	⬭	⬭	⬭	⬭	⬭	⬭	Clearly prefer TEXT INSTRUCTIONS

6. **It was very intuitive to program the PICK operation**
Mark only one oval.

	1	2	3	4	5	6	7	
Strongly disagree	⬭	⬭	⬭	⬭	⬭	⬭	⬭	Strongly agree

7. **It was very intuitive to program the PLACE operation**
Mark only one oval.

	1	2	3	4	5	6	7	
Strongly disagree	⬭	⬭	⬭	⬭	⬭	⬭	⬭	Strongly agree

8. **Comments for specific parts of the learning process**

....
....
....
....
....

General Information

9. **Gender ***
Mark only one oval.

⬭ Male
⬭ Female

10. **Age ***

....

11. **COMPUTERS / IT** *

Please rate your own proficiency / level of expertise
Mark only one oval.

- 1 - Fundamental Awareness (basic knowledge)
- 2 - Novice (limited experience)
- 3 - Intermediate (practical application)
- 4 - Advanced (applied theory)
- 5 - Expert (recognized authority)
- N/A (not applicable)

12. **ROBOTICS** *

Please rate your own proficiency / level of expertise
Mark only one oval.

- 1 - Fundamental Awareness (basic knowledge)
- 2 - Novice (limited experience)
- 3 - Intermediate (practical application)
- 4 - Advanced (applied theory)
- 5 - Expert (recognized authority)
- N/A (not applicable)

13. **Additional comments/notes/suggestions**

dimchrisos@gmail.com
Edit this form

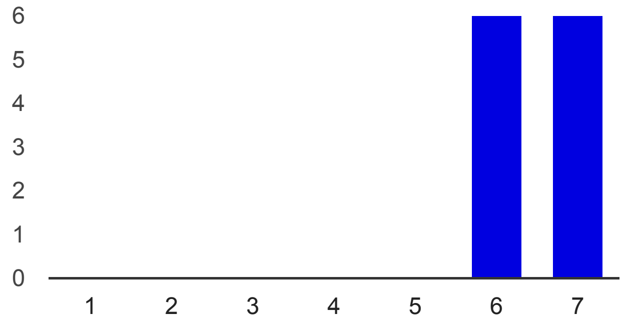
12 responses

View all responses Publish analytics

Summary

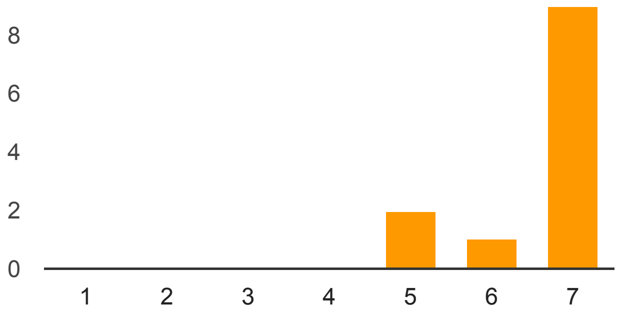
Questions related to GENERAL SATISFACTION

Overall, I am satisfied with the EASE OF COMPLETING the tasks in this scenario

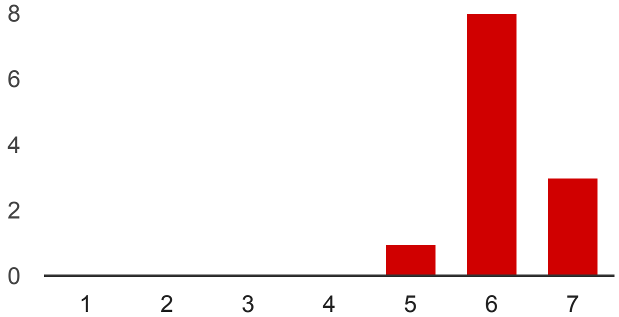


Strongly disagree: 1	**0**	0%
2	**0**	0%
3	**0**	0%
4	**0**	0%
5	**0**	0%
6	**6**	50%
Strongly agree: 7	**6**	50%

Overall, I am satisfied with the AMOUNT OF TIME it took to complete the tasks in this scenario



Overall, I am satisfied with the SUPPORT INFORMATION when completing the tasks



Strongly disagree: 1	**0**	0%
2	**0**	0%
3	**0**	0%
4	**0**	0%
5	**1**	8.3%
6	**8**	66.7%
Strongly agree: 7	**3**	25%

Comments for the general satisfaction

Instructed to press in the z-direction - before an illustration of the z-direction is presented. May give confusion.

The text telling me to put the robot in "camera position" was a little unintuitive, I would have preferred an instruction like: "remove the robot from the camera view" (it would also be nice to see the camera view live while doing that). Also is there any reason why the "local home" and the "camera position" isn't the same thing? Some times pictures showing what was x,y and z was missing in the GUI. I believe it would be very intuitive with audio feedback when e.g. you confirm an teaching (play a sound after your press). Lastly i thought it was a little strange that the confirm action changed from press "Y" to "Z" at some point. Overall it was pretty intuitive to teach the skill although and autocomplete on the keyword you write in the beginning could be a help (i imagine it could be hard to remember all keyword that are possible to put in).

Pictures for all instructions would be good

I think the process is well defined. One suggestion about GUI. The task, shown in GUI, that is to be performed can be made more prominent by highlighted text, highlighter or by adding some voice commands as well. Just to make it much more easier for a person to understand.

The process would be easier if the GUI was closer to you. Like using an iPad or similar.

Easier to use than when you have to select skills. Would be even more usefull for larger tasks The screen and mouse should have been closer to the task.

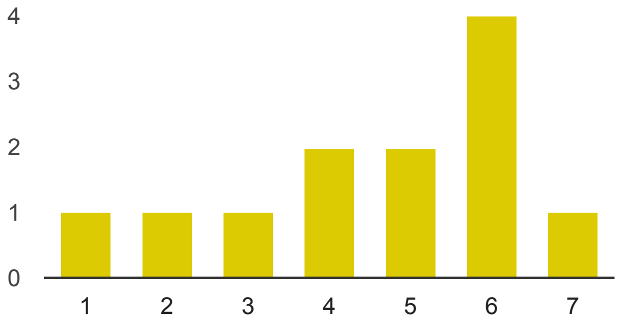
very easy to yuse

The last task where the part were to be placed above the final position was a bit confusing. I did not know whether to press y or z, or both in a sequence. Otherwise, it was really easy to follow.

a few times i was not 100% sure about what to do, so i think the instructions some times could be more clear

Questions related to SPECIFIC PARTS of the learning process

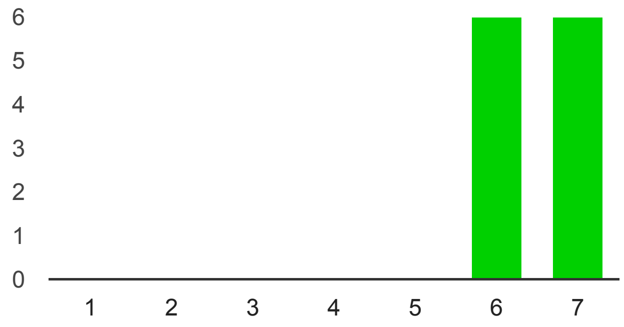
The robot can be programmed through TEXT INSTRUCTIONS and by MANUAL SELECTION OF SKILLS. Which method would you prefer when programming the robot?



Clearly prefer MANUAL SELECTION OF SKILLS:	1	**1**	8.3%
	2	**1**	8.3%
	3	**1**	8.3%
	4	**2**	16.7%
	5	**2**	16.7%
	6	**4**	33.3%

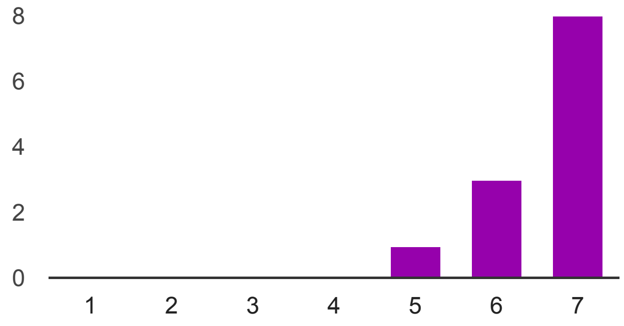
Clearly prefer TEXT INSTRUCTIONS: 7	**1**	8.3%

It was very intuitive to program the PICK operation



Strongly disagree: 1	**0**	0%
2	**0**	0%
3	**0**	0%
4	**0**	0%
5	**0**	0%
6	**6**	50%
Strongly agree: 7	**6**	50%

It was very intuitive to program the PLACE operation



Strongly disagree: 1	**0**	0%
2	**0**	0%
3	**0**	0%
4	**0**	0%
5	**1**	8.3%
6	**3**	25%
Strongly agree: 7	**8**	66.7%

Comments for specific parts of the learning process

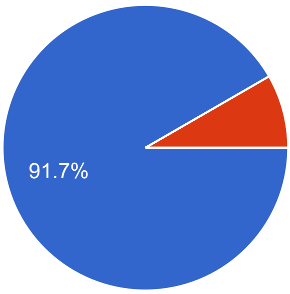
Could be more intuitive if the axis was color coded together with stickers on the robot showing the axis.

As mentioned in the earlier form it would be nice with an overview of what to write or an autocomplete/recommend "function" to help suggest the words.

Please do not take the first question into account. My answer would not be so precise as I did not completely understand what you mean by TEXT INSTRUCTIONS and MANUAL SELECTION OF SKILLS.

General Information

Gender



Male	**11**	91.7%
Female	**1**	8.3%

Age

26

29

28

27

51

53

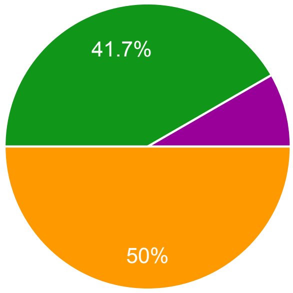
52

23

24

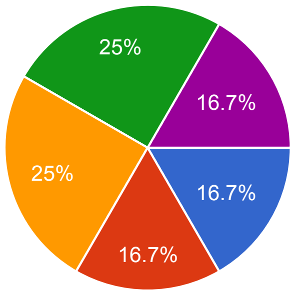
31

COMPUTERS / IT



1 - Fundamental Awareness (basic knowledge)	[illegible]	[illegible]
2 - Novice (limited experience)	**0**	0%
3 - Intermediate (practical application)	**6**	50%
4 - Advanced (applied theory)	**5**	41.7%
5 - Expert (recognized authority)	**1**	8.3%
N/A (not applicable)	**0**	0%

ROBOTICS

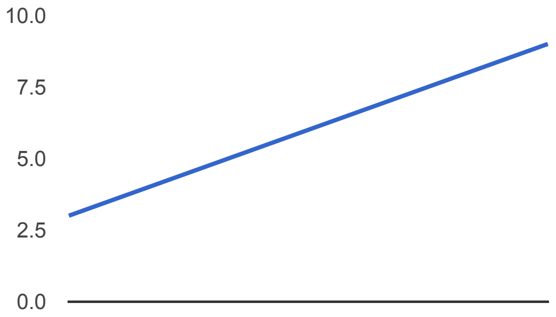


1 - Fundamental Awareness (basic knowledge)	**2**	16.7%
2 - Novice (limited experience)	**2**	16.7%
3 - Intermediate (practical application)	**3**	25%
4 - Advanced (applied theory)	**3**	25%
5 - Expert (recognized authority)	**2**	16.7%
N/A (not applicable)	**0**	0%

Additional comments/notes/suggestions

New to the skill-base system.

Number of daily responses


10.0
7.5
5.0
2.5
0.0