# ToBI - Team of Bielefeld
# A Human-Robot Interaction System for RoboCup@Home 2023

Leroy Rügemer and Sven Wachsmuth

Research Center for Cognitive Interaction Technology (CITEC),
Bielefeld University, Inspiration 1, 33615 Bielefeld, Germany
https://clf.pages.ub.uni-bielefeld.de/Robocup/website/

**Abstract.** Team of Bielefeld (ToBI) was founded in 2009. The RoboCup team's activities are embedded in a long-term research agenda towards human-robot interaction with laypersons in regular and smart home environments. The RoboCup@Home competition is an important benchmark and milestone for this goal in terms of robot capabilities as well as the system engineering approach. In order to achieve a robust and stable system performance, we apply a methodical approach for reproducible robotic experimentation including automated tests. Another focus is reusability which means both, the reuse in different robot tasks as well as the reuse across different platforms. For RoboCup 2023, we introduce several innovations such as an improved world model coding appropriate poses for the robot interacting with its environment, generation of synthetic training data for object classification and a simulation pipeline for testing human-robot interactions.

## 1 Introduction

The RoboCup@Home competition aims at bringing robotic platforms to use in realistic domestic environments. Today's robotic systems obtain a big part of their abilities through the combination of different software components from different research areas. To be able to communicate with humans and interact with the environment, robots need to coordinate and dynamically configure their components to generate an appropriate overall robot behavior that fulfills parallel goals such as gathering scene information, achieving a task goal, communicating its internal status, and being always responsive to humans. This is especially relevant for complex scenarios in domestic settings.

ToBI was founded in 2009 and successfully participated in the RoboCup World Cup (2009-2018) as well as the RoboCup German Open from 2009 to 2019 on different robot platforms within RoboCup@Home. Each year a team of students has been newly formed and supervised by senior researchers and student members of the previous years.

In 2016, the team won the global @Home-competition [2] and finished first in several of the individual tests (Navigation, Person Recognition, GPSR, EE-GPSR, Restaurant). At RoboCup 2018, the team achieved first place in the

Social Standard Platform League (SSPL). Although the team newly acquired the TIAGo platform late in 2018, the Team placed 1st at the 2019 RoboCup@Home German Open.

Bielefeld University is involved in research on human-robot interaction for more than 20 years, especially gaining experience in experimental studies with integrated robotic systems [9,12]. Within this research, strategies are utilized for guiding the focus of attention [10], providing behaviors for object reference and object handovers [12], as well as work on Mixed Reality interfaces [3].

The introduction of a systematic approach towards reproducible robotic experiments [7] has been turned out as a key factor to maximally stabilize basic capabilities like, e.g., navigation or person following. Further aspects — regarding the RoboCup@Home — refer to the usage of systematic world models that introduce knowledge to the robotic system on how to interact with its environment. Based on these models, capabilities are needed to re-configure the navigation and grasping skills on-the-fly considering environmental factors. This is implemented within our system as a navigation plug-in and through the usage of the Task Constructor framework for `MoveIt!`.

## 2 Robot Platforms

In 2016, ToBI participated in RoboCup@Home with the two service robots Biron and Floka, in 2017 with Biron and Pepper. Although focusing on the TIAGo since 2019, we still aim at the development of platform-independent, as well as multi-platform robot capabilities.
**The Open Platform TIAGo** (Fig. 5) from PAL Robotics is a service robot for indoor environments. The technical specifications can be found online[1]. We equipped the robot with a Lenovo P52 notebook running Ubuntu Linux 18.04 on a 3D-printed mount. Additional computing power is provided by a Jetson TX1 module and an Intel Neural Compute Stick 2. For improved recognition, we added a Sennheiser MKE 400 shotgun microphone along with a RealSense L515 LiDAR Camera to the head. Our version of the robot is equipped with a Schunk WSG32 gripper with a maximum opening width of 68mm. As this is not enough to grasp the objects utilized in the competition, we augmented the robot with a custom expansion. This non-invasive add-on turns the parallel into a scissoring motion to increase the maximum opening width. The gripper is designed to mount up to four adaptive fingers.

## 3 System Architecture

Our service robots employ distributed systems with multiple clients sharing information over the network. On these clients, there are numerous software components written in different programming languages. Such heterogeneous systems require abstraction on several levels.

[1] http://pal-robotics.com/wp-content/uploads/2019/07/Datasheet_TIAGo_Complete.pdf

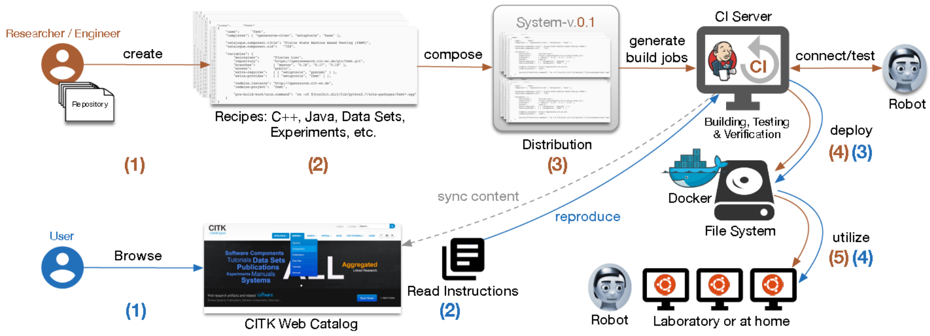



**Fig. 1.** Cognitive Interaction Toolkit: toolchain and workflow. The red numbers show the workflow of the system developer, while the blue numbers represent the workflow of a researcher reproducing the system.

### 3.1 Reusable Behavior Modeling

For modeling the robot behavior flexibly, ToBI uses the BonSAI framework. It is a domain-specific library that builds up on the concept of *sensors* and *actuators* that allow the linking of perception to action [11]. These are organized into robot *skills* that exploit certain *strategies* for informed decision-making.

To support the easy construction of more complex robot behavior, BonSAI supports modeling of the control flow, as e.g. proposed by [1]. The BonSAI `behavior-scxml` extension implements this modeling using State Chart XML (SCXML). It serves as a sequencer for the overall system by executing *skills* to construct the desired robot behavior. This allows separating execution of the skills from the data structures they facilitate, thus increasing the re-usability of the skills. Through the usage of BonSAI, the same skill could also be reused on different systems that use XCF, ROS, RSB or NaoQi as middleware. The BonSAI framework has been released under an Open Source License and is available online[2].

### 3.2 Development and Deployment Tool-Chain

The software dependencies — from operating system dependencies to inter-component relations — are completely modeled in the description of a *system distribution* which consists of a collection of so-called *recipes* [7]. In order to foster reproducibility, traceability, and potential software (component) re-use of the ToBI system, we provide a full specification of the 2023 system in our online catalog platform[3]. The catalog provides detailed information about the soft- and hardware system, including all utilized software components.

Our development and deployment process is illustrated in Fig. 1 (red numbers) [8].

[2] https://github.com/CentralLabFacilities/bonsai
[3] https://citkat-citec.bob.ci.cit-ec.net/distribution/tiago-melodic-nightly.xml

Since setting up a CI server and the required plugins takes time and requires expert knowledge, we provide prepackaged installations in the [Research & Robotics] Development Toolkit (RDTK)[4]. The RDTK further allows a to deploy RDTK-based systems using Linux containers, like Docker. System descriptions and their metadata, e.g. source code locations, wiki pages, issue tracker, current build status, experiment descriptions, and so forth are frequently synchronized to a web-based catalog Cognitive Interaction Toolkit Catalogue (CITKat) that also implements the RDTK data model providing a global, human-readable, and search-able platform which is a prerequisite for open research.

## 4 Entity Component World Model

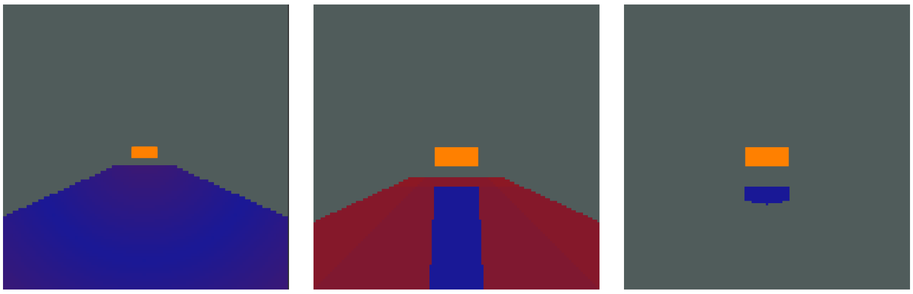
To facilitate sophisticated reasoning about the environment, a representation of the world is needed. The Entity component World Model (ECWM) framework provides an Entity component system (ECS) based World model for quick prototyping. Known information is encoded as different `components` in an entity pool of `models`. Most of these are geometry-based to describe rigid objects, such as furniture, with basic primitives or 3D meshes and more advanced semantic information, such as descriptions for named regions, e.g. the different shelves of a bookcase. Some `components`, such as point clouds and egocentric maps, providing suitable positions for arbitrary interactions, can be calculated offline or during startup by utilizing the `plugin` system. All `model components` are then static during the runtime of the robot task.

The world state comprises the instances of the `models` in the current `entity` pool, where each `entity` contains `components`, e.g. position, observations, themselves. ECWM provides the world as a list of entities and the ability to add, remove and update as well as query the associated (`model`)`components` of entities via ROS services. Additionally, it also provides interfaces for runtime plugins to collect sensor information and update `entity` properties such as the current position. As an example, to ensure the positions of the model reflect the real-world state, the `icp-matcher` plugin associates robot sensor data with the geometries enabling the position refinement with Iterative Closest Point (ICP).

**Interaction Maps** are model components that map robot positions to robot poses.

Each interaction map represents an arbitrarily sized 2D map centered at the object with action-dependent size and resolution. Each cell of the map has a value for suitability and the desired robot joint configuration. The `geometries` can have multiple interaction maps for different actions, like *recognize objects on the lower shelf* or *grasp from upper shelf* as shown in Fig. 2. As the capabilities of robots can be different, the maps are specifically generated based on the description of the robot.

[4] https://rdtk.github.io/documentation/



(a) View the bookcase (b) Recognize objects from the upper shelf (c) Grasp from the upper shelf

**Fig. 2.** Visualization of different interaction maps for the bookshelf (in orange). The blue to red shading encodes the suitability of the cell, other space is insufficient for the interaction

The `recognize objects` interaction, for a specific planar surface of the `geometry`, calculates suitability scores by computing how much of the surface is visible for different camera sensor positions using ray tracing. This reduces the occlusion of objects during robot operation. It also applies a factor calculated from measuring the deviation to an optimal view angle and distance to reproduce the settings applied during gathering the object recognition training data.

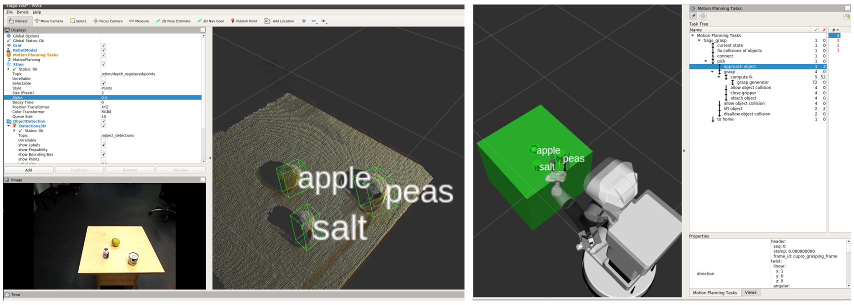
Even slight position changes may take a considerable amount of time, especially for robots with differential drives. Thus different maps can be combined to find a valid position for multiple interactions, possibly minimizing the required robot movement.

## 5 Further Software Contributions

**Navigation** To improve navigation capabilities for our different platforms, we maintained a fork of the ROS `navigation`[5] packages. To facilitate the reuse of our changes, we decided to repackage the changes into different plugins in separate packages `clf_navigation_plugins`[6]. As navigation tasks differ in their requirements for path planning and obstacle avoidance, our navigation stack is able to select different configurations for each navigation goal.

**3D Object Reconstruction** Software such as `Meshroom`[5], `Open3D`[7] enable 3d reconstruction of meshes, we refine these RGB-based techniques by using RGBD data for accurate depth measurements as well as visual markers for robust camera pose estimation during acquisition.

[5] https://github.com/CentralLabFacilities/navigation
[6] https://github.com/CentralLabFacilities/clf_navigation_plugins
[7] http://www.open3d.org/




(a) Object Recognition visualization (b) Grasp Task

**Fig. 3.** Object Recognition visualization and resulting object primitives for grasp generation

These meshes enable pose refinement e.g. ICP, and calculation of suitable grasp points with ECWM.

As the performance gap between models trained on real or synthetic data is shrinking[13], we have replaced manual recording and labeling with Physically based rendering (PBR) of reconstructed models utilizing `kubric`[8] and `NVISII`[9].

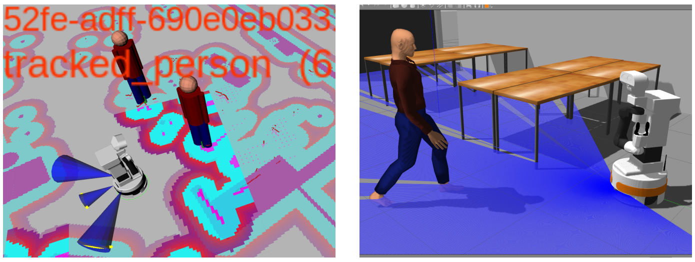
**Object Recognition** Our current 2d object recognition utilizes the ROS `vision_msgs` package. We augment the 2D recognition results with 3D segmentation and super quadratic fitting of object primitives. Figure 3(a) shows our visualization plugin for `vision_msgs`. The `clf_object_recognition` package is hosted on our GitHub[10].

We are currently in the process of evaluating state-of-the-art 6DOF pose estimation trained on datasets. While 6DOF pose estimation networks, trained on single models like DOPE[14], already show good results in our lab, the high memory footprint prevents the application on the robot.

**Manipulation** TIAGo utilizes the Task Constructor framework for MoveIt! [4] which provides a way to solve manipulation tasks by defining multiple interdependent subtasks organized in hierarchical containers, allowing for sequential as well as parallel compositions. The framework enables us to work on the improvements of subtasks. During manipulation, grasps are selected by specific quality metrics depending on the information of the known model or the inferred primitive shape of the object obtained from the perception pipeline. The hierarchy of our grasp task can be seen in Fig. 3(b).

---

[8] https://github.com/google-research/kubric
[9] https://github.com/owl-project/NVISII
[10] https://github.com/CentralLabFacilities/clf_object_recognition




(a) Visualization of person tracking (b) Simulation of Human Robot Interaction in Gazebo

**Fig. 4.** Person Perception

**Person Perception** Pose detection is done with `OpenPose`. We use a customized ROS wrapper[11] that can compute gesture (pointing, waving, raising arms) and posture (sitting, standing, lying) of detected people as well as their global position using the RGBD camera of TIAGo.

To increase the robustness of person tracking in follow-me-styled tasks, we utilize multiple trackers and merge them with bayesian filter tracking. We combine laser-based leg detection with video stream tracking such as tracking learning detection [6].

To facilitate automated testing of human-robot experiments, we currently develop a simulation pipeline based on gazebo actors. In the current state, we can manually control the simulated person, as well as trigger predefined movements. We plan to implement a fully automated benchmark for the follow-me tasks with the possibility to instruct the simulated guiding person via specific voice commands to change the speed or wait for the robot to catch up.

## 6 Conclusion

We have described the main features of the architecture and technical solutions of the ToBI system for the RoboCup@Home Open Platform League (OPL) 2023. Based on the already achieved development state and an analysis of the robot's performance at previous competitions, we improved the software architecture and development cycle in several aspects. The architecture allows to program and use robot skills across multiple ecosystems on both, internal and external computing resources of the robot. The incremental system development stages are completely reproducible by using the RDTK environment. We are confident to further improve the capabilities of the TIAGo robot to a significant degree utilizing the experiences from very successful RoboCup@Home competitions from 2009 to 2019.

[11] https://github.com/CentralLabFacilities/openpose_ros

## References


1. Boren, J., Cousins, S.: The smach high-level executive. Robotics & Automation Magazine, IEEE **17**(4), 18–20 (2010)
2. Meyer zu Borgsen, S., Korthals, T., Lier, F., Wachsmuth, S.: ToBI – Team of Bielefeld: Enhancing Robot Behaviors and the Role of Multi-Robotics in RoboCup@Home, vol. 9776. Springer (2016)
3. Meyer zu Borgsen, S., Renner, P., Lier, F., Pfeiffer, T., Wachsmuth, S.: Improving Human-Robot Handover Research by Mixed Reality Techniques. In: VAM-HRI 2018. The Inaugural International Workshop on Virtual, Augmented and Mixed Reality for Human-Robot Interaction. Proceedings (2018). https://doi.org/10.4119/unibi/2919957
4. Görner, M., Haschke, R., Ritter, H., Zhang, J.: Moveit! task constructor for task-level motion planning. 2019 International Conference on Robotics and Automation (ICRA) pp. 190–196 (2019)
5. Griwodz, C., Gasparini, S., Calvet, L., Gurdjos, P., Castan, F., Maujean, B., Lillo, G.D., Lanthony, Y.: Alicevision Meshroom: An open-source 3D reconstruction pipeline. In: Proceedings of the 12th ACM Multimedia Systems Conference - MMSys '21. ACM Press (2021). https://doi.org/10.1145/3458305.3478443
6. Kalal, Z., Mikolajczyk, K., Matas, J.: Tracking-learning-detection. Pattern Analysis and Machine Intelligence, IEEE Transactions on (2012)
7. Lier, F., Hanheide, M., Natale, L., Schulz, S., Weisz, J., Wachsmuth, S., Wrede, S.: Towards Automated System and Experiment Reproduction in Robotics. In: Burgard, W. (ed.) 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE (2016)
8. Lier, F., Wienke, J., Nordmann, A., Wachsmuth, S., Wrede, S.: The Cognitive Interaction Toolkit – Improving Reproducibility of Robotic Systems Experiments. In: Brugali, D. (ed.) SIMPAR 2014, LNAI. pp. 400–411. Springer International Publishing Switzerland (2014)
9. Lohse, M., Siepmann, F., Wachsmuth, S.: A Modeling Framework for User-Driven Iterative Design of Autonomous Systems. International Journal of Social Robotics **6**(1), 121–139 (2014). https://doi.org/10.1007/s12369-013-0209-8
10. Richter, V., Carlmeyer, B., Lier, F., Meyer zu Borgsen, S., Kummert, F., Wachsmuth, S., Wrede, B.: Are you talking to me? Improving the robustness of dialogue systems in a multi party HRI scenario by incorporating gaze direction and lip movement of attendees. In: Proceedings of the Fourth International Conference on Human-agent Interaction. ACM Digital Library (2016)
11. Siepmann, F., Wachsmuth, S.: A Modeling Framework for Reusable Social Behavior. In: De Silva, R., Reidsma, D. (eds.) Work in Progress Workshop Proceedings ICSR 2011. pp. 93–96. Springer, Amsterdam (2011)
12. Simmering, J., Meyer zu Borgsen, S., Wachsmuth, S., Al-Hamadi, A.: Combining static and dynamic predictions of transfer points for human initiated handovers. In: ICSR-2019 (2019)
13. Sundermeyer, M., Hodaň, T., Labbé, Y., Wang, G., Brachmann, E., Drost, B., Rother, C., Matas, J.: Bop challenge 2022 on 6d object localization (2022), http://cmp.felk.cvut.cz/sixd/workshop_2022/slides/bop_challenge_2022_results.pdf
14. Tremblay, J., To, T., Sundaralingam, B., Xiang, Y., Fox, D., Birchfield, S.: Deep object pose estimation for semantic robotic grasping of household objects. In: Conference on Robot Learning (CoRL) (2018), https://arxiv.org/abs/1809.10790

## Robot Description

Our TIAGo specifications are as follows:

- Base: Differential drive 1m/s max speed.
- Torso: 35cm lift.
- Arm: 7DOF, 87cm reach, Maximum load: 3kg.
- Head: Pan and tilt motion with an RGB-D camera.
- Dimensions: height: 110-145cm, footprint: ø54cm.
- Robot weight: 70kg.
- 25m Laser Range, 3×1m sonar.

*Additional Hardware:*

- Schunk WSG32 gripper with a custom extension.
- Lenovo P52 Notebook.
- Sennheiser MKE 400.
- Jetson TX1 module.
- Intel Neural Compute Stick 2.
- Intel RealSense LiDAR Camera L515.

**Fig. 5.** TIAGo

## Robot's Software Description

*A exhaustive list of our software and all its dependencies can be found online*[12].

- Operating System: Ubuntu 18.04 LTS with ROS Melodic
- Navigation: karto, movebase, CLF Planner[13]
- Object Recognition: YOLOX, DOPE
- Manipulation: Moveit Task Constructor [4]
- People Detection: Strands perception people[14], Openpose, CFTLD based Tracking, Bayes Tracking
- Speech Recognition: PocketSphinx with context-dependent ASR
- Speech Synthesis: Acapela TTS
- Behavior Control: BonSAI with SCXML [15]

## External Devices and Services

*TIAGo uses neither.*

[12] https://citkat-citec.bob.ci.cit-ec.net/distribution/tiago-melodic-nightly.xml
[13] https://github.com/CentralLabFacilities/clf_navigation_plugins
[14] https://github.com/strands-project/strands_perception_people
[15] https://github.com/CentralLabFacilities/bonsai

## Team Information

### Name of Team:

Team of Bielefeld (ToBI)

### Contact information:

Leroy Rügemer
Research Center for Cognitive Interaction Technology (CITEC)
Bielefeld University
Inspiration 1, 33619 Bielefeld, Germany
{tobi,lruegeme}@techfak.uni-bielefeld.de

### Website:

https://clf.pages.ub.uni-bielefeld.de/Robocup/website/

### Team video:

https://www.youtube.com/watch?v=tNkCyATHV74

### Team members:

Sven Wachsmuth, Leroy Rügemer, Bernd Heihoff, Carolin Klute, Dennis Holzmann, Hakan Yldirim, Jonas Vaquet, Jonas Zilke, Kathrin Lammers, Laurin Gräser, Leander von Seelstrang, Linus Hagen Schwarz, Lukas Walkenhorst, Michale Wagemann, Zoe Klinger