A Survey about the Usage of Semantic Technologies for the Description of Robotic Components and Capabilities

Stefan Zander
FZI Research Center for
Information Technology
Haid-und-Neu-Str. 10-14
76131 Karlsruhe – Germany
zander@fzi.de

Nadia Ahmed
FZI Research Center for
Information Technology
Haid-und-Neu-Str. 10-14
76131 Karlsruhe – Germany
ahmed@fzi.de

Matthias Frank
FZI Research Center for
Information Technology
Haid-und-Neu-Str. 10-14
76131 Karlsruhe – Germany
frank@fzi.de

ABSTRACT

This work surveys the state of the art in applying semantic technologies and ontology-based knowledge representation frameworks to the domains of robotics and cyber-physical systems. It specifically focuses on works that utilize semantic technologies for expressing the metadata models of hardware and software components together with their capabilities and characteristic features, as these are the main application areas of ontologies in model-driven engineering approaches in robotics. This work therefore analyzes sev-eral ontological description features such as the ontology language, scope and purpose of used or created ontologi-cal classification systems, application domains, ontology extensibility, reasoning problems together with the reasoning techniques and the technological approach being used. As this work shows, the usage of semantic technologies and ontologies in robotics and cyber-physical systems is constantly increasing regardless of specific domains.

Keywords

Ontology, Knowledge Representation, Semantic Web, Cyber-Physical Systems, Reasoning

1. INTRODUCTION

The development of robotic applications is usually a resource and time consuming task, involving the participation of different experts (hardware and software developers, system integrators etc.) and a wide variety of tools and component specifications. In order to cope with the high resource and knowledge demands and the inherent device and tool heterogeneity, model-driven engineering has recently introduced to the development of robotic applications and reveled promising results (cf. [1, 24, 29]). However, the impact of such initiatives can be improved if they are synthesized with semantic technologies and ontology-based knowledge representation frameworks. In this respect, a central role is given to the axiomatic description of components and their features in order to foster the utilization and interoperability between components and applications using ontological domain and task semantics. This work therefore surveys the current state of the art in using semantic technologies and ontology-based knowledge representation frameworks for the description of robotic components and their capabilities. A special emphasis is given to the practical application of such technologies in current robotic projects as well as on initiatives that aim at building industry-wide vocabularies.

SamI40 workshop at i-KNOW '16 October 18–19, 2016, Graz, Austria

2. ANALYZED APPROACHES

Table 1 provides an overview of the analyzed approaches along with their respective publications and their year of publication. For each approach, we chose the most relevant publications wrt. the scope of this work. We decided to subsume projects with several publications into one approach. Moreover, this method allows us to consider publications, which do not entirely match the selection criteria of our classification system, but describe important aspects of an approach. In the remainder of this section, we briefly and informally introduce each of the analyzed works.

Matching Sensors to Missions using a Knowledge-Based Approach— Preece et al [23] describe an approach for matching missions (consisting of tasks) to sensor systems with the use of capabilities. It aims at assigning unmanned aerial vehicles (UAVs), equipped with sensors for intelligence, surveillance and reconnaissance (ISR), to missions, that consists of several tasks. Apart from its military alignment, this approach was one of the first that describes the idea of matching robot components (sensors on platforms) with tasks (missions) via capabilities by utilizing semantic descriptions in a comprehensible manner. As evident from our analysis, the approach lacks the description of actuators (physically effecting body parts).

Ontology for Robotics: a Roadmap—In [21] the authors develop an ontology for designing a robot-system, illustrate it by a simple example and explain the tools they use, their prototyping flowchart, as well as some implementation details. Contrary to what its title might lead one to conclude, this work is not an overview of ontology approaches for robotics. We will discuss their approach and structure based on OWL-S[1], an OWL dialect for web services, and try to figure out why it seems to be difficult to reason automatically on top of it.

---

[1] http://www.w3.org/Submission/OWL-S/

Project Name	Publications	Year
Matching Sensors to Missions	Matching Sensors to Missions Using a Knowledge-Based Approach [23]	2008
Ontology for Robotics: a Roadmap	Ontology for Robotics: a Roadmap [21]	2009
A Workpiece-Centered Approach	Model-Based Configuration – A Workpiece-Centered Approach [5]	2009
KnowRob	KnowRob – Knowledge Processing for Autonomous Personal Robots [25]	2009
	The Semantic Robot Description Language [16]	2011
	Knowledge Processing for Autonomous Robot Control [26]	2012
	KnowRob – A Knowledge Processing Infrastructure for Cognition-enabled Robots [27]	2013
SRDL	The Semantic Robot Description Language [16]	2011
ROSETTA	Automatic Generation of Robot Applications Using a Knowledge Integration Framework [19]	2010
	A Knowledge Integration Framework for Robotics [22]	2010
	Knowledge and Skill Representations for Robotized Production [7]	2011
	Knowledge for Intelligent Industrial Robotics [6]	2012
Semantic Web for Robots	Using semantic technologies to describe robotic embodiments [13]	2011
	Semantic Web for Robots [12]	2012
IEEE ORA WG: Kitting Applications	Knowledge Driven Robotics for Kitting Applications [3]	2013
	Implementation of an Ontology for Industrial Robotics [4]	2014

Table 1: Overview of analyzed approaches including relevant publications and publication year

Model-Based Configuration: A Workpiece-Centered Approach— Within the context of the EU-Project SIARAS (Skill-based Inspection and Assembly of Reconfigurable Automation Systems) and the European Network of Excellence I*PROMS, Bengel et al. [5] published a workpiece-centered approach, demonstrated at Fraunhofers IPA's "Advanced Modular Microproduction System" (AMMS). The goal of their work is to provide engineers a tool (within their accustomed CAD program) in order to describe the evolutionary steps a workpiece has to pass through during the manufacturing process. The goal is to assist engineers in selecting an appropriate robot and tool together with the required configuration and operations using a reasoner. Therefore, the authors built an ontology in order describe each working step (e.g. required skills and devices for a given product).

KnowRob: Knowledge processing for robots— KnowRob is an actively maintained open-source knowledge processing system for service robots. It provides a central knowledge representation framework in form of an ontology, along with tools for acquisition, reasoning, probabilistic inference, classification and clustering, grounding and capability matching among others. The ontologies, the source-code and its documentation and tutorials are available on KnowRob website[2]. The project was originally developed in the IAS group at the Technical University of Munich, Germany, and has received funding from and as part of several research projects (CoTeSys[3], the EU FP7 projects RoboEarth 2, RoboHow[4] and SHERPA[5], amongst others). Since 2009, various papers have been published as part of this project, some relevant of which have been considered for this survey (see Table 1); A full list of publications can be retrieved on the publication's section[6]. Further information can be found in [28].

SRDL: Semantic Description Language— There is one specific paper [16] to emphasize, introducing the Semantic Robot Description Language (SRDL). SRDL is a part of the KnowRob project, and focuses our topic, namely semantic descriptions of robot components and capabilities. The main goal of SRDL is linking descriptions of robot components via capabilities to actions in an ontology. SRDL is one part of the KnowRob processing system, and focuses on reasoning about the robots components and capabilities.

ROSETTA (EU FP7 project)— ROSETTA[7] is a research project[8] conducted within the context of the 7th Framework Programme (EU-FP7). The goal of the project is to provide a flexible manufacturing system where the desired product evolves and changes frequently. Their approach covers *task-level programming* (use of high-level instructions instead of low-level motion programming), *robot program reuse* through centralized knowledge repositories with reusable control programs and a *safety approach* to assure the safety of humans within a working cell (see the website[9] for the list of publications).

RoboEarth (EU FP7 project)— RoboEarth[10] [30] is an open-source knowledge base, which aims to store and share information for robots within a cloud infrastructure. It contains knowledge about maps for the robot navigation, object recognition models, robot tasks and its software components. A robot systems can access the cloud platform and collaborate with other robots to learn and achieve differ-

---

[2] http://www.knowrob.org

[3] http://www.cotesys.org/

[4] http://www.robohow.eu

[5] http://www.sherpa-project.eu

[6] http://www.knowrob.org/publications

[7] ROSETTA is the acronym for "RObot control for Skilled ExecuTion of Tasks in natural interaction with humans; based on Autonomy, cumulative knowledge and learning"

[8] http://fp7rosetta.org

[9] http://fp7rosetta.org/?q=node/6

[10] http://www.roboearth.org

ent tasks. RoboEarth uses the KnowRob system as a local knowledge base for a given robot, especially for the mapping between components and tasks. Therefore, the classification of this approach is covered by the classification of KnowRob (see the website[11] for a list of RoboEarth publications).

RoboDB— In 2011, Juarez et al. proposed RoboDB [13], an approach that uses semantic Web technologies to describe robotic embodiments, along with their capabilities. RoboDB is intended to be a usable by humans as well as robots or software agents. For that reason, it provides Web-browser access and a Web-service endpoint. In order to encourage knowledge creation and collaboration, the system and the Web interface in particular is designed to be easy to use. Reusability, availability, facilitating knowledge creation and web-friendliness are the main goals of this project. The RoboDB is used by [12] in a more comprehensive system that is introduced in the subsequently following paragraph and also included in the classification framework.

Semantic Web for Robots— Alex Juarez published an approach [12] that aims at "applying semantic web technologies for interoperability between virtual worlds and real robots" in 2012. The approach uses the semantic descriptions gathered in RoboDB [13] to build an ontology for robotic components. RoboDB is especially used as a collaborative knowledge acquisition system for encoding information about robotic devices. The ontology covers component capabilities and is extended by a rule language to model more complex capabilities and uses also a reasoning engine. Additionally, the work focuses also on interoperability aspects between real devices and virtual worlds via a prototype for Assisted Communication (PAC4), a service registration system that is responsible for the connection between virtual world and real robots. In his approach, the author uses various semantic Web technologies and mixes elements from the robotic domain with virtual worlds. The use case demonstrates the applicability on a LEGO Mindstorm NXT[12] robot, which is controlled by a virtual person within a virtual world.

IEEE ORA WG: Kitting Applications— The ORA (Ontologies for Robotics and Automation) working group consist of 157 members from 23 different countries and aims at building an IEEE standard ontology for robots. Within the scope of this standardization effort, a work (see [4]) was published to give an overview about an ontology within the domain of manufacturing kitting. The authors describe an ontology, which is used to assist the generation of task plans for robots. The ontology consists of knowledge about robot environment, e.g., location of the tray or the parts contained in the tray, that must be placed in a kit. Additionally, the ontology is extended by concepts and relations that aim at validating and verifying whether a given task is successful and correct (where the preconditions and effects must satisfied for the achievement of a task).

3. CLASSIFICATION FRAMEWORK AND EVALUATION

This section introduces the classification framework being used for the analysis of the works introduced in Section 2. We analyzed them according to the following six dimensions that are relevant for describing robotic components and capabilities: (i) domain and scope of an approach, (ii) system design and architecture, (iii) ontology scope and extensibility, (iv) reasoning features, (v) technological foundation of the reasoning techniques, and (vi) additional relevant technological features. Each dimension is discussed and analyzed in a separate sub section, which starts with a general description of the aspect and its criteria followed by an evaluation both in textual and tabular form. Table 2 provides an overview of the symbols being used in the evaluation tables.

Legend to the Symbols	
✓	The examined papers of the approach mention the criterion as supported or the context suggests it as supported. The papers do not necessarily describe their solution in detail.
(✓)	The examined papers of the approach mention the criterion, but the provided solution is incomplete or different from the description of the criterion.
?	The examined papers of the approach do not mention the criterion and we do not know if it is supported.
–	Either the examined papers of the approach mention the criterion as not supported or the criteria is not in the focus of the examined papers and thus it is probably not supported. *It is important to notice that the approach might support the criteria nevertheless.*

Table 2: Explanation of used symbols for the tabular categorization of analyzed works. These symbols express the degree to which a certain aspect has been fulfilled by a specific work.

3.1 Domain and Scope

3.1.1 Description

An obvious and relevant criteria is the domain of the inspected approach. The most common one is the manufacturing and the service robotics domain, but there are also papers focusing on the military domain. The domain influences the weights of problems to solve and the goals an approach tries to accomplish. While in a work cell, it is more relevant how several robots act together; for a service robot, it is more important to communicate with humans or to ground symbols in the knowledge base.

A further dimension, which is important for the classification of an approach is whether it addresses the design-time or the run time aspects of a robot system. For instance the matching between components and task is relevant at design time. And the real time performance is required. The criteria for the domain dimension are listed in Table 3 and the evaluation results are shown in Table 4.

3.1.2 Evaluation Results

The approach "Matching Sensors to Missions" addresses the military domain especially the unmanned aerial vehicles in the context of intelligence, surveillance and reconnaissance resource allocation. The authors focus on matching tasks (missions) to autonomous sensor-systems with use of a semantic description approach. They describe their reasoning approach, a proof of concept implementation and feedback from domain experts. The paper "Ontology for Robotics: a Roadmap" describes an approach for matching devices (robot components) to tasks via skills using a semantic description of the involved concepts in the context

---

[11] http://www.roboearth.org/publications/

[12] http://www.lego.com/en-us/mindstorms

Domain	
Industrial	The domain of manufacturing robotics focuses on automatic assembly lines. Often several robots have to work together and fast reprogramming is important.
Service	The domain of personal, domestic and household robotics comprises mostly mobile autonomous robots with emphasis on human interaction and working out solutions by the robot itself.
Military	The military domain includes unmanned aerial or ground vehicles (UAVs / UGVs), developed to collect intelligence or fighting wars.
Other	Other domains not covered by the predefined list.
Design-Time	The approach is suitable for and focuses on the design phase of a robot system
Run-Time	The approach can be utilized at and addresses run-time applications

Table 3: Overview of the extracted criteria within the domain dimension with brief explanations

of the manufacturing domain and covers the idea, workflowchart and some implementation details, but no working example. The objective of "A Workpiece-Centered Approach" is clearly the manufacturing domain, where workpieces get modified and combined to derive a product. The paper covers the idea, architectural details, related and future work as well as a working experiment. KnowRob addresses the domain of service robotics. An autonomous robot has to orient itself in an environment, where unforeseen events may occur, and must be able to infer the detailed course of action and the action parametrization at run-time. The grounding problem, managing uncertainty and fast inference are explicitly discussed. The domain of SRDL is the field of service robotics. The paper covers more details of the ontology and the possible inference algorithms with sample problems and their solution. The EU FP7 project "ROSETTA" clearly addresses the manufacturing domain, with focus on automatic generation of executable code from high-level process information. One speciality of this approach being the integration of already existing data (plant data in Automation ML) in the Knowledge Base. The authors assume that in the domain of robotics in manufacturing data is normally available in the AutomationML[13] exchange format [8], and develop a procedure to automatically convert XML documents into RDF triples. The "Semantic Web for Robots" approach describes the knowledge engineering, as well as a case study, connecting a LEGO Mindstorm NXT robot to the virtual world "Second Life". The developed subsystems RoboDB and PAC4 are discussed and design decisions explained. Since most of the work is rather abstract, particularly addresses rather general problems, it is difficult to assign a specific domain. The case study may show the closest resemblance to service robotics, therefore the approach is classified to address this domain partly. As part of the Industrial Subgroup of the IEEE Working Group, the "IEEE ORA WG: Kitting Applications" papers address the manufacturing domain. Within this area, their case for application is industrial kitting, where a collection of separate items is grouped, packaged and supplied together as one unit.

	Domain					
	Industrial	Service	Military	Other	Design-Time	Run-Time
Matching Sensors to Missions	–	–	✓	–	✓	–
Ontology for Robotics	✓	–	–	–	✓	–
Workpiece-Centered Appr.	✓	–	–	–	–	✓
KnowRob	–	✓	–	–	–	✓
SRDL	?	✓	?	?	–	✓
ROSETTA	✓	–	–	–	–	✓
Semantic Web for Robots	?	(✓)	?	✓	–	✓
IEEE ORA: Kitting Apps	✓	–	–	–	–	✓

Table 4: Summary of the evaluation results of the domain related criteria as described in Table 3

3.2 Idea and Architecture

3.2.1 Description

This section analyses the system architecture and the main idea of each approach which is a crucial dimension since it gives an overview about the work and the adopted concept. We also discuss the architecture, whereby if a whole system is presented, the architecture of its parts is also discussed. However, the idea gains more attention for more theoretical work. Since this dimension relies on exposing the uniqueness of each approach and because of the comparability of ideas is not possible, it is not covered within the tabular classification.

3.2.2 Evaluation Results

The idea of "A Workpiece-Centered Approach" is to describe manufacturing workpieces from the users perspective. It aims to deduce what must be achieved rather than what operations must the devices execute for a given task. The workpiece is within the focus of this approach and is described during all its evolutionary stages without knowledge neither about the necessary devices neither how to program them. Since these devices and algorithms together with their parametrization are automatically inferred. Additionally, combining several workpieces together consist of pairwise aligning them on respectively selected supporting points. Therefore, a tree structured dependency graph is built in order to derive the ordering of all required steps. Afterwards, devices are selected and a control software translates the problem statement into executable program steps.

The idea of KnowRob is to provide a knowledge base which consists of encyclopedic knowledge as well as knowledge about instances of physical objects, action models and common-sense knowledge. Additionally Knowrob provides an interfaces between the robot and the knowledge found in the web through sensors. The knowledge base interacts with an observation system and the robot itself. The observation system consists of sensors mounted on the robot and mounted within a kitchen environment. It includes a vision system, RFID tag readers, magnetic switches, a full body pose tracking system and the robot action log. While the

---

[13] http://www.automationml.org/

Unified Robot Description Format (URDF)[14] can be used to describe the geometry and the kinematics of robot hardware components, it lacks semantic descriptions of those components. Semantic Robot Description Language (SRDL) includes not only the semantic descriptions of components but also describes actions and capabilities.

The idea of the ROSETTA approach is to use a *Knowledge Interchange Framework* (KIF) as the center of the architecture. An engineering station is used to interact with the system which enables users to specify the work cell and define the tasks to accomplish. The KIF captures various types of knowledge in one big ontology. The center of the KIF is the Knowledge Storage, which contains the mostly static data like device and skill descriptions, planning algorithms, CAD data, process and domain knowledge, as well as sensor data interfaces, recovery strategies and information about injury potential. The KIF is modelled in three layers, namely Knowledge Storage, Code Generation and Execution. Starting with an informal process description, entered by a user, the *Knowledge Storage* generates a formal application description, beginning with natural language processing and applying its domain, process and task knowledge. Any missing information is queried by the user. The formal description contains all necessary information, the tasks to be executed, their order and parameters, but it stills device independent. The server processes the abstract description in two phases. Firstly, a high-level controller decomposes the task in steps, which are realizable by the available devices. Secondly, in the *Code Generation*-Layer, devices are selected and assigned to operations. The task specifications are instantiated with parameters, followed by a generation of executable code fragments for each task finally this code is integrated within an application. The *Execution Layer* is responsible for surveillance and error recovery for events that interrupt the normal execution. E.g., a human enters the robot cell.

The main purpose of the "Semantic Web for Robots" approach is to allow the reusability of semantic robot component and capability descriptions. RoboDB addresses this issue by using semantic web technologies. An user-friendly web interface facilitates knowledge creation in a collaborative way. The gathered knowledge is usable by robots or software agents, which can connect to a web service endpoint. A further major goal is to allow the interoperability between robots and virtual worlds. PAC4 is responsible for that part and provides a communication system, which implements the observer design pattern. In a nutshell this pattern can be described as a notification system between objects, whereby when an object changes its state, the system automatically notifies the object observers (other objects) and updates them.

The design methodology of the article "IEEE ORA WG: Kitting Applications" is to provide a system with more agility and fast reconfiguration. This is accomplished by providing the appropriate knowledge in the correct format to all modules of the system.The authors mention, that it "is not intended to replace a sound engineering of an intelligent system" [3]. The systems overall knowledge model is organized in four layers. In the first layer, where the design process starts, Domain specific information (DSI) is gathered. Form one ore more use cases the typical operations of the system are derived and high-level descriptions of actions and their preconditions are derived. With this knowledge an ontology (second layer) is build, containing all objects relevant for system operation, the State of the world, Ordering constructs, Actions and Predicates (the SOAP ontology), as well as the initial and goal states for the system. The Planning Domain Definition Language (PDDL) [18] is utilized in the third layer. A PDDL Domain File and a PDDL Problem File are generated automatically from the knowledge base. The planning language contains still high-level actions. In the last layer, an interpreter combines knowledge from the plan file with actual values of object locations or similar properties, in order to form robot language commands. Those values for properties of concrete instances are stored in a MySQL database[15]. The tables for the database are automatically constructed from the ontology and maintained by the sensor processing system.

3.3 Ontology Scope and Extensibility

3.3.1 Description

This subsection describes the covered *scope* of the ontology as well as its *extensibility*:

- Scope refers to the covered parts of a robot. It considers robot components that can be software or hardware components (sometimes also referred to as device or device components). Moreover, components can be classified into more fine-grained types of taxonomical classifications systems such sensor and actuator or more complex such as composition of components or more abstract such as tasks or capabilities. The studied approaches are categorized along the feature of knowledge about an object that the robot manipulates. The scope of the ontologies is shown in Table 5.
- Extensibility considers the ability to learn new concepts on class level and integrate them in the ontology which is called *learning capability*. Some of the examined ontologies are embedded within an *upper ontology*, like the Suggested Upper Merged Ontology (SUMO) [20] or the Unified Foundational Ontology (UFO) [9]. The two criteria among *external connectivity* state if the ontology provides interfaces for or integrates with other ontologies or further sources. The extensibility of ontologies is shown in Table 6.

3.3.2 Evaluation Results

Table 7 contains an overview of the covered scope of the ontology within each approach. In all analyzed works except "IEEE ORA WG: Kitting Applications" (abbreviated as I3E7 in Table 7), the sensors of a robot are semantically described. Additionally, in the approach proposed by "Matching Sensors to Missions" (denoted by S2M in Table 7), the ontology covers *sensors* mounted on *platforms*. However, *actuators* are not considered. In their approach, a task is composed of sub-tasks where a top-level task is called a mission. A Sensor-system consists of several sensors mounted on a platform. Both, sensors and platforms provide capabilities, which are required by tasks or sub-tasks. The sensors deliver typical observation data (e.g., optical, acoustic, thermal) and are mounted on a platform, which provides some properties such as the range, speed and endurance.

---

[14] http://www.ros.org/wiki/urdf

[15] https://www.mysql.com/

Ontology Scope	
Sensor	Hardware component for measuring and observing the environment or the robot state
Actuator	Hardware component for effecting the physical world, e.g. a robot arm or gripper
Software	Any kind of control program or algorithm for processing or planning
Task	A formal description of actions and often sub actions, mostly organized in a hierarchical structure
Capability	A feature or functionality that a hardware or a software component provides or requires to achieve a specific task.
Information Object	Semantic representation of environmental objects, like workpieces or topological maps
Composition	A component consisting of other components is called a component composition
Robot Group	A set of robots collaborating together to achieve a specific task

Table 5: Overview of the extracted criteria within the ontology scope dimension with brief explanations

Ontology Extensibility		
Learning Capability		
Upper Ontology		A core or domain ontology might be embedded in an upper ontology
External Connectivity	Ontology	The approach is designed to connect with some other ontology
	Other Sources	External data is or can be included in or accessed by the ontology by default

Table 6: Overview of the extracted criteria within the ontology extensibility dimension with brief explanations

"Ontology for Robotics: a Roadmap" (abbreviated as O4R in Table 7) specifies tasks, devices (hardware components), and skills (capabilities), whereby devices can perform a task if they are associated via a skill. Particularly they define atomic processes that correspond to single actions that a service can perform, and composite processes to represent multi-step protocols. A composite process is composed of atomic or composite processes where control constructs (e.g. sequence, if-then-else) are also modelled.

The Ontology of "A Workpiece-Centered Approach" (abbreviated as WPC in Table 7) is used to model the workpieces to be manufactured (e.g., material, weight, grasping and fitting points). A workpiece model must be available in several evolutionary stages during the manufacturing process and the current status of the workpiece is tracked. The work-cell is is also modelled in a scene tree, where all initial positions of all objects are defined. Now a manufacturing process can be modelled by regarding only the workpiece and its alterations, especially without the need of knowledge about the devices.

The overall structure of the ontology in"KnowRob" Approach (abbreviated as KRob in Table 7) is inspired by the Cyc ontology [17]. Some knowledge is imported from the OMICS database. The encyclopedic knowledge models classes of Things in a hierarchical structure. Things are SpatialThings, Actions, Events and Computables. Instances model physical objects, performed actions and observed events. They can be generated at run-time using computable classes. In contrast, action models generate new classes from observed data, leading to the learning capability of the system.

In "SRDL" the most relevant concepts are *Robot*, *Component*, *Action* and *Capability.* A *Robot* consists of a set of components, and a component can be a hardware component, a software control program or an object. An *Action* definition follows OpenCyc's definition of *PurposefulAction* and *requires* one or more *Capabilities.* Where a *Capability* *needs* one or more *Components* and *depends* on other *Capabilities.* Additionally, a hardware component may have an successor in the kinematic chain, modelled by a transitive property. A high-level component is defined as a component composition. An *Action* consists of sub-actions, which are *Actions* themselves and consists of sub-actions.

The ontology created in "ROSETTA (EU FP7 project)" (abbreviated as ROS in Table 7) aims to be extensible. The main idea is to decouple devices, skills and processes. A *Process* is mainly a sequence of *Tasks* which is an assignment of work to be accomplished under certain constraints. It comprises the knowledge how to assemble a desired product. Multiple *Tasks* may depend on each other. *Objects* are modelled as a physical entity. A *Workpiece* is modelled as an *Object* which is influenced or modified by a process. Furthermore the ontology classifies a *Device* representing a physical *Object* providing abilities described as *Skills. Device* concepts are structured and classified in a taxonomy.

The ontology of "Semantic Web for Robots" (abbreviated as S4R in Table 7) contains the concept of *Robot*, *Component* and *Sensor.* Their approach separates *Sensor* from *Component* by defining both concepts on the same level, rather than *Sensor* being a subconcept of *Component.* Hence *Component* describes any other physical part of the robot, except sensors. The robot structure is modelled with a set of properties: *has component*, *has sensor* and *is connected to* are relevant for this purpose. Each time a user enters a new description in the knowledge base, a subconcepts of *Sensor* or *Component* is generated appropriately. Moreover, the modelling of capabilities is inspired by the web services ontology OWL-S. Thus OWL concepts for *inputs*, *outputs*, *requirements* and *effects* are included in the ontology. A *Capability* can be defined to be fulfilled, if a robot meets the specified requirements, which means it has specific properties or relationships. For example, a robot *belongs* to the concept *DrivingCapability*, if it is related to some *Wheel* and *Motor* (by the property *has component*) and some *MotionEffect.*

The "IEEE ORA WG: Kitting Applications" (abbreviated as I3E in Table 7) ontology defines two top-level concepts. While *SolidObject* represents physical objects, *DataThing* represents data for *SolidObject.* For example, subconcepts of *SolidObject* are parts, kits and trays, thereby their *Location* is a subconcept of *DataThing.* The ontology also contains models of "executable information", including actions, preconditions, effects and failures. Additionally, this approach allow to generate automatically PDDL[16] files responsible for the description of an action plan for a robot

---

[16] http://cse3521.artifice.cc/pddl.html

	Ontology Scope								
	Sensor	Actuator	Software	Task	Capability	Information Object	Product / Workpiece	Composition	Robot Group
S2M[1]	✓	–	–	✓	✓	–	–	✓	✓
O4R[2]	✓	✓	–	✓	✓	–	–	✓	–
WPC[3]	✓	✓	(✓)	✓	–	(✓)	✓	✓	✓
KRob[4]	✓	✓	✓	✓	✓	✓	–	✓	(✓)
SRDL	✓	✓	✓	✓	✓	✓	–	✓	?
ROS[5]	✓	✓	✓	✓	✓	✓	✓	✓	✓
S4R[6]	✓	✓	–	(✓)	✓	–	–	✓	?
I3E[7]	?	?	(✓)	✓	(✓)	✓	✓	?	?

[1] Matching Sensors to Missions
[2] Ontology for Robotics: a Roadmap
[3] A Workpiece-Centered Approach
[4] KnowRob
[5] ROSETTA
[6] Semantic Web for Robots
[7] IEEE ORA WG: Kitting Applications

Table 7: Summary of the evaluation results of ontology scope related criteria as described in Table 5

	Ontology Extensibility			
	Capability Learning	Ontology Upper	External Connectivity	
			Ontology	Other Sources
S2M[1]	–	–	–	–
O4R[2]	–	?	?	–
WPC[3]	–	?	Device Library	CAD data
KR[4]	✓	Custom based on OpenCyc	RoboEarth DB	Web instructions, Online Shops, URDF
SRDL	–		?	
ROS[5]	(✓)	?	?	CAEX, AutoML, CAD, MathML
S4R[6]	–	–	RoboDB	PAC4/MPEG-V
I3E[7]	–	SUMO CORA	?	PDDL

[1] Matching Sensors to Missions
[2] Ontology for Robotics: a Roadmap
[3] A Workpiece-Centered Approach
[4] KnowRob
[5] ROSETTA
[6] Semantic Web for Robots
[7] IEEE ORA WG: Kitting Applications

Table 8: Summary of the evaluation results of the ontology extensibility related criteria as described in Table 6

task. Table 7 shows the evaluation of the scope dimension of the analyzed ontologies whereas Table 8 summarizes the evaluation results of the extensibility dimension.

3.4 Reasoning Features

3.4.1 Description

One of the main features that distinguishes logic-based knowledge representation frameworks from other modeling approaches such as UML is the ability to draw inferences on the logical entailments of constituting axioms [15]. This feature is called *reasoning* or synonymously *inference* and provides answers wether a model is consistent or whether an assertion is satisfiable given a specific knowledge base. Reasoning is often used for the deduction of new facts derived from explicit assertions via logical calculus. For this survey, we differentiate between *standard reasoning tasks* (e.g., classification, class membership or subsumption computations) (cf. [10]) and reasoning specifically applied for *component to task matching.*

- For the dimension of standard reasoning, we analyzed several criteria. Firstly, the *automated classification* of components along a given set of criteria, e.g., the capabilities a component offers or characteristic features. The axiomatic representation of such features using terms from domain ontologies allows to compute class memberships (a form of classification) in which an individual component is classified as a specific type when necessary and sufficient conditions that determine class membership are fulfilled. Secondly, we considered the feature of *multidimensional classification.* Given a component that is already classified according to a specific type but has, e.g., an output dimension of 2D and an output format of image, the reasoner is able to compute additional classifications, i.e., it can classify this component e.g. as 2DCamera. Thirdly, we analyzed the *capability aggregation* feature where the reasoner infers aggregated capabilities of a component compositions, i.e., systems that consists of several individual components and which provide superior features based on the cooperation and interplay of involved components. Furthermore, we also analyzed the *robot system integration* feature. This feature refers to the reasoning about the assembly of hardware components (e.g., kinematic chain of a robot or of several machines in a production line). Additionally, we considered also the incompleteness problem and extracted the following two criteria: (1) completeness of an instruction assigned by a human to a robot (*instruction completion*) and (2) completeness of a task description, i.e., whether reasoning techniques are used to infer whether a given task description is complete or misses information. Finally, we analyzed the criteria of (*realtime performance*) as reasoning is inherently a computationally expensive task. This feature is of particular importance for scenarios or settings that require ad-hoc decision-making processes and hence fast inference calculations. A tabular overview of the considered feature is listed in Table 9.
- As illustrated in Table 10, the component-task matching dimension refers to the deployment of reasoning

techniques for inferring the set of required components wrt. a given task as well as the set of tasks that a given component is able to accomplish. Furthermore, the considered reasoning approaches also include the computations of rankings for valid matching component candidates by analyzing several quality criteria such as resource efficiency, price, and availability.

Reasoning: General Features	
Automated Classification	Classify new concepts by their properties and relations to existing concepts
Multidimensional Classification	Automated classification within several branches of one or more concept structures.
Capability Aggregation	Aggregate required capabilities for a task over subtasks and aggregate provided capabilities of component compositions vice versa
Robot System Integration	Assemble physical components in a working way, and construct a robot or assembly line in doing so.
Action Completion	Infer the detailed course of action, insert (sub-)actions if applicable and deduce the action parametrization
Instruction Completion	Using (e.g. common sense) knowledge to complete underspecified instructions given by humans
Realtime Performance	Especially for run-time applications fast inference algorithms are important. This criteria reflects if an approach emphasizes realtime performance.

Table 9: Overview and brief explanations of the aspects that constitute the general reasoning feature

Reasoning: Matching	
Component-Task Matching	Ability to deduce required components for a given task, mostly by expressing its capabilities
Ranking Matchings	Ability to rank more than one matching solution of components and tasks by some property or properties (cheapest, most available, resource efficient, best quality-of-information)
Filtered Matching	Prefilter the components available for the matching
Filter by Run-Time Information	Consider Run-Time Information, like the component status, for the filtering

Table 10: Overview and brief explanations of the reasoning matching dimensions

3.4.2 Evaluation Results

The reasoner approach used in “Matching Sensors to Missions” calculates matches between tasks, platforms and sensors via capabilities.

In “Ontology for Robotics: a Roadmap”, the authors assume that a complex task does not inherit the skills of subtasks; thus, a complex task is described by a new skill “ and must be explicitly recorded because only elementary tasks, and so elementary skills, are directly connected with physical devices”. The same approach is applied to devices. Each new combination of tasks or devices needs to be manually related by a new skill. Consequently, the automated reasoning depends on a lot of manually set skills.

The reasoner methodology used in “A Workpiece-Centered Approach” discovers an appropriate device for each skill with assigned properties in three steps. Firstly, the available devices are filtered by their general ability to accomplish a given skill. Then, these devices are evaluated whether they satisfy the specified requirements and properties of the given skill. Finally, quality criteria are evaluated to find the best suitable device.

The knowledge base of “KnowRob” enables different types of reasoning with near real-time performance. It is possible to query objects by their functionality and to find matching actions based on their specifications, which is evaluated against the observed robot motions. Additionally, it is possible to complete underspecified instructions given by humans.

In “SRDL”, the reasoner is involved in checking whether a robot can perform an action based on several conditions that must be satisfied. Firstly, the set of required capabilities for a specific action must be provided by a robot. Secondly, the robot must be able to perform all sub-actions that are related to an action. Checking whether a robot provides a capability is performed by checking whether all required components for a specific capability as well as all its sub-capabilities are available to the robot. In case a robot is not able to perform an action, a list of missing capabilities and components to perform a desired action is suggested. Additionally, the number of trials and successes of a specific actions are collected in order to calculate the success probability of an action. If the action is performed the first time, the success probability is calculated based on the success probabilities for all sub-actions.

The hierarchical classification structure of the ontology developed within “ROSETTA (EU FP7 project)” allows to classify concepts by specifying relations to existing objects. However, in [7], a specific reasoner is not explicitly described, since the rationale is that a knowledge-based system can be extended by external reasoners. In [6] the authors discuss the use of Jena2[17] for a rule-based reasoning using the Knowledge Interchange Format (KIF)[18].

The authors of “Semantic Web for Robots” use SWRL (Semantic Web Rule Language) (see [11]) which follows the structure of implications and provides built-in expressions for comparison, mathematical operations, string manipulation, date, time and duration specification. Therefore, SWRL allows for example to define that DrivingCapability requires at least two wheels. However, the usage of these rules may lead to the undecidability problem.

To the best of the authors knowledge, the existence of matching algorithms between robot components and tasks is not explicitly addressed in “IEEE ORA WG: Kitting Applications”.

3.5 Technological Foundation

3.5.1 Description

This feature evaluates the technological and formal foundation of the notation frameworks and reasoning engines used for the representation and processing of ontological information. Three aspects were evaluated in particular

---

[17] http://jena.apache.org/

[18] http:/logic.stanford.edu/kif/specification.html

	Reasoning: Matching			
	Component-Task Matching	Ranking Matchings	Filter-Based Matching (Resource Allocation)	Filter by Run-Time Component Status
Matching Sensors to Missions	✓	–	(✓)	–
Ontology for Robotics: a Roadmap	✓	–	✓	–
A Workpiece-Centered Approach	✓	✓	✓	–
KnowRob	✓	?	✓	?
SRDL	✓	?	✓	–
ROSETTA	✓	?	?	–
Semantic Web for Robots	✓	–	✓	?
IEEE ORA WG: Kitting Applic.	?	?	?	?

Table 11: Summary of the evaluation results of the reasoning features for matching computations as described in Table 10

	Reasoning: General Features						
	Automated Classification	Multidimensional Classification	Capability Aggregation	Robot System Integration	Action Completion	Instruction Completion	Realtime Performance
Matching Sensors to M.	✓	✓	✓	?	–	–	–
Ontology for Robotics	?	–	–	?	–	–	–
Workpiece-Centered	?	?	?	–	✓	–	–
KnowRob	✓	?	✓	–	✓	✓	✓
SRDL	✓	?	✓	–	?	–	✓
ROSETTA	✓	?	✓	–	✓	✓	?
Semantic Web for Robots	✓	✓	✓	–	–	–	?
Kitting Applications	?	?	?	–	✓	?	✓

Table 12: Summary of the evaluation results of the general reasoning features as described in Table 9.

Technological Foundation	
Ontology	The language used for the formal representation of the ontology
Reasoning	The reasoning engine or logic programming language used to infer from terminological knowledge and constraints
Other	Other mentionable technologies, which are special for a particular approach

Table 13: A summarization of the evaluation criteria of the Technology dimension

(see Table 13): The ontology languages used for the representation, the reasoning engine used for the inference and materialization, and additional frameworks and APIs for processing and querying. Most of the considered works make use of the Web Ontology Language (OWL) or some of its language profiles. For the reasoning aspect, different state of the art reasoning engines where considered such as Pellet, Jena, or even Prolog. Almost all approaches use additional frameworks or APIs in order to achieve specific objectives or to provide additional functionalities such as querying or the interoperability with external standards such as Collada. The evaluation of the technological foundations of the considered works is summarized in Table 14.

3.5.2 Evaluation Results

The work "Matching Sensors to Missions" uses the DL profile of OWL, OWL DL [2] for representing the ontology. As reasoner they utilize Pellet[19], an OWL 2 reasoner for Java. The assignment of specific platform configurations to several tasks (missions) is achieved while solving the set-cover problem with a separate algorithm.

Nilsson et al. use the "Semantic Markup for Web Services" OWL-S ontology language[20] for "Ontology for Robotics: a Roadmap", which is built on top of OWL. Since things become complicated later on they use the JastAdd[21] meta-compilation system to generate new compiler specifications to support dynamic states for objects in the ontology without rewriting the compiler (reasoner).

The focus of "A Workpiece-Centered Approach" is on the workpiece-centered view, and so the information about technological details are scarce. Though the ontology is modelled using OWL, and the interface to describe the evolution steps is embedded within a CAD application.

The ontology of "KnowRob" is represented with OWL, whereas the reasoning is mainly done with SWI Prolog[22]. The integration into the robot control program is facilitated by the YARP middleware[23]. The robot action log uses the Robot Learning Language (RoLL [14]).

The SRDL ontology is implemented in the Web Ontology Language OWL. For reasoning algorithms it uses the logical programming Language PROLOG.

The implementation of the "ROSETTA" KIF uses the RDF database Sesame[24] for all stored knowledge. Some kinds of data are not represented in the RDF triple store. Instead the node URIs get associated to HTTP accessible data with the Linked Data method [7]. The information and relations can be queried with SPARQL[25]. For CAD data, the Collada file format[26] is used. The conversion procedure of AutomationML[27] into RDF triples is implemented in the XSLT language[28].

---

[19] http://clarkparsia.com/pellet/

[20] http://www.w3.org/Submission/OWL-S/

[21] http://jastadd.org/web/

[22] http://www.swi-prolog.org/

[23] http://wiki.icub.org/yarp/

[24] http://www.openrdf.org

[25] http://www.w3.org/TR/rdf-sparql-query/

[26] http://www.khronos.org/collada/

[27] https://www.automationml.org/o.red.c/home.html

[28] http://www.w3.org/TR/xslt20/

	Technological Foundation		
	Ontology	Reasoning	Other
S2M[1]	OWL DL	Pellet	Set-Covering Algorithm
O4R[2]	OWL-S	JastAdd	
WPC[3]	OWL	?	CAD Application
KRob[4]	OWL	SWI Prolog	YARP, RoLL URDF, ROS
SRDL	OWL	SWI Prolog	URDF
ROS[5]	OWL/RDF	Jena2/RIF	SPARQL, Collada, ABB RobotStudio
S4R[6]	OWL(-S)	SWRL/Jena	TDB, Joseki
I3E[7]	OWL,XML	?	PDDL, ROS, CRCL

[1] Matching Sensors to Missions
[2] Ontology for Robotics: a Roadmap
[3] A Workpiece-Centered Approach
[4] KnowRob
[5] ROSETTA
[6] Semantic Web for Robots
[7] IEEE ORA WG: Kitting Applications

Table 14: Summarization of the evaluation results for the technological foundation aspect

The "Semantic Web for Robots" ontology is implemented in OWL. The example implementation utilizes the Jena Semantic Web framework4[29], which provides an OWL/RDF API and a SPARQL endpoint. As local repository for the data TDB[30] is used. Jena can be accessed through the Joseki HTTP engine[31], which provides a web-service endpoint with SPARQL support. Additional reasoning rules are formulated in SWRL, as mentioned above.

The authors of "IEEE ORA WG: Kitting Applications" implemented their ontology in both, OWL and XML. They utilize a MySQL database to store instance properties in oder to achieve higher performance. Using the Planning Domain Definition Language enables the use of many open-source planning systems. To be independent of the language used by the robot controller, and the planning language, the canonical robot command language (CRCL) is deployed in between. The authors themselves use the ROS platform[32] to control the robot system [3].

3.6 Additional Features

3.6.1 Description and Evaluation

The final aspect evaluates additional relevant features an approach offers. Those features are not directly involved with the component-task matching process but give an impression about other relevant areas for the deployment of semantic Web technologies in robotic systems. One relevant criteria, especially for run-time applications, is the (symbol) grounding problem, where recognized objects in the environment are related and represented in a knowledge base. For instance, a robot may observe the visual features of a cup. Then the grounding algorithm creates an instance of the concept `cup` in the knowledge base and the robot is able to connect all known information about cups to the recognized object (e.g., cups can be filled with liquids). *Managing uncertainty* is another real-world problem that most robotic systems have to address. For instance, most observed sensor data are not accurate enough to process out-of-the box; a robotic system therefore has deploy several data processing steps in order to use acquired data for achieving a specific task. The interaction with humans is another crucial aspect—not only in the domain of service robotics but also for industrial robots. Therefore, we also considered the feature of *Natural Language Processing* (NLP) as part of the human-robot-interaction. Also important for robotics is *injury risk management*, which is related to *failure handling and recovery.* For example, when a human worker enters a working cell of an assembly line, an unexpected event that disturbs the usual workflow might be happening and requires technical precautions in order to reduce the likelihood of such an event and resulting consequences.

Additional Features	
Grounding Problem	Automated generation of (sub-)concepts by environmental observations through sensor data
Managing Uncertainty	Ability to handle real-world (sensor-) data with tolerance, e.g. probability calculations
Failure Handling / Recovery	Identify execution failures or unforeseen events and manage them accordingly
Natural Language Processing	A procedure to translate human language in a machine comprehensible one
Injury Risk Management	Some system that prevents humans from being injured by the robot system

Table 15: Summarization of the different additional features the works contain and provide

	Additional Features				
	Grounding Problem	Managing Uncertainty	Natural Language Processing	Failure Handling / Recovery	Injury Risk Management
Matching Sensors to Missions	(✓)	–	–	–	–
Ontology for Robotics	–	–	–	–	–
A Workpiece-Centered Approach	?	?	–	?	–
KnowRob	✓	✓	✓	?	–
SRDL	–	–	–	–	–
ROSETTA	?	✓	✓	✓	✓
Semantic Web for Robots	–	–	–	–	–
IEEE ORA WG	✓	✓	–	✓	?

Table 16: Summarization of the evaluation results for the additional features aspect

---

[29] http://jena.sourceforge.net/
[30] http://openjena.org/TDB/
[31] http://www.joseki.org/
[32] http://www.willowgarage.com/pages/software/ros-platform

The grounding problem has been addressed by KnowRob and the IEEE ORA Working Group. Although it is not explicitly mentioned in “Matching Sensors to Missions”, their proposed classification framework can be used as basis for the deduction of sub concepts from real-world observations. The management and handling of uncertain data is a topic that could be found in almost four out of eight works—mainly in those that require the incorporation and processing of sensorial data. Natural language processing is relevant for both the KnowRob and ROSETTA projects although in different manifestations. The failure handling and recovery feature has been addressed by the ROSETTA project and the IEEE ORA working group. For the KnowRob project and a “A Workpiece-Centered Approach”, no reliable information that would allow an unambiguous assignment to the failure handling and recovery category could be found. The injury risk management feature has been explicitly treated only by the ROSETTA project and indirectly by the IEEE ORA working group.

4. CONCLUSIONS

Ontologies are a well-suited and promising technique in the field of robotics. Scientists separately research ontologies and robotics for quite some time. Researching the combination of both is relatively new in turn. There is not one formal, explicit specification of a shared conceptualization of robot components, their capabilities and tasks, that stands out or seems to be superior. In this work, we surveyed recent approaches (projects and scientific works) from the domains of robotics specifically and cyber-physical systems in general that utilize semantic Web technologies and ontology-based knowledge representation frameworks for the description of hardware and software components together with their capabilities. We created a classification framework that analyzes domain and purpose of each approach, the ontology features and ontology language used together with the aspects scope and extensibility. A special emphasis was given to reasoning, a major aspect that distinguishes ontology-based description models from common component models used in computer science such as the UML. In the reasoning aspect, we analyzed the reasoning problems (matching, classification, aggregation, integration, completion) that were addressed in each work together with the deployed reasoning technique and underlying ontology languages. The approaches reviewed in this survey each focus on a subdomain of robotics. Some present concepts or core ontologies that could be utilized in other subdomains; however, their adaptability has to be shown in the future. Regarding the application areas of ontology-based semantic descriptions, most approaches apply them for managing run-time aspects. One analyzed feature that deserves special attention is the grounding symbol problem, which describes the instantiation of observed objects in the knowledge base. From an abstract point of view, it connects a dynamic and continuous world with a more or less static and discrete knowledge base. Some approaches already reported promising progress, but there is much room for further research.

5. ACKNOWLEDGMENTS

We want to thank Mr. Hannes Wächter for conducting the initial analysis of the considered works. This work has been partially funded by the German Federal Ministry for Economic Affairs and Energy through the project ReApp (no. 01MA13001). Parts of the presented results were also created within the Digital Innovation Center (DIZ – Digitales Innovationszentrum) funded by the Ministerium für Wirtschaft, Arbeit und Wohnungsbau Baden-Württemberg.

6. REFERENCES

[1] D. Alonso, C. Vicente-chicote, F. Ortiz, J. Pastor, and B. Alvarez. V3CMM: a 3-View Component Meta-Model for Model-Driven Robotic Software Development. *Journal of Software Engineering for Robotics (JOSER)*, 1(January):3–17, 2010.

[2] F. Baader, D. Calvanese, D. L. McGuinness, D. Nardi, and P. F. Patel-Schneider. *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press, 2003.

[3] S. Balakirsky, Z. Kootbally, T. Kramer, A. Pietromartire, C. Schlenoff, and S. Gupta. Knowledge driven robotics for kitting applications. In *Robotics and Autonomous Systems*, volume 61, pages 1205–1214, 2013.

[4] S. Balakirsky and A. Price. Implementation of an Ontology for Industrial Robotics. pages 10–15, 2014.

[5] M. Bengel. Model-based configuration - A workpiece-centred approach. *2009 ASME/IFToMM International Conference on Reconfigurable Mechanisms and Robots*, 2009.

[6] a. Björkelund, J. Malec, and K. Nilsson. Knowledge for intelligent industrial robots. *AAAI Technical Report: Designing Intelligent Robots: Reintegrating AI*, (Schmolze 1986), 2012.

[7] A. Björkelund, J. Malec, K. Nilsson, and P. Nugues. Knowledge and skill representations for robotized production. In *IFAC Proceedings Volumes (IFAC-PapersOnline)*, volume 18, pages 8999–9004, 2011.

[8] R. Drath. *Datenaustausch in der Anlagenplanung mit AutomationML*. Springer-Verlag, 2010.

[9] G. Guizzardi and G. Wagner. Towards Ontological Foundations for Agent Modelling Concepts Using the Unified Fundational Ontology (UFO). In P. Bresciani, P. Giorgini, B. Henderson-Sellers, G. Low, and M. Winikoff, editors, *Agent-Oriented Information Systems II*, volume 3508 of *Lecture Notes in Computer Science*, pages 110–124. Springer Berlin Heidelberg, 2005.

[10] P. Hitzler, M. Krötzsch, and S. Rudolph. *Foundations of Semantic Web Technologies*. CRC Press, 2010.

[11] I. Horrocks, P. F. Patel-schneider, H. Boley, S. Tabet, B. Grosof, and M. Dean. SWRL : A Semantic Web Rule Language Combining OWL and RuleML, 2004.

[12] A. Juarez. *Semantic Web for Robots: Applying Semantic Web Technologies for interoperability between virtual worlds and real robots*. PhD thesis, Eindhoven University of Technology, 2012.

[13] A. Juarez, C. Bartneck, and L. Feijs. Using semantic technologies to describe robotic embodiments. In *Human-Robot Interaction (HRI), 2011 6th ACM/IEEE International Conference on*, pages 425–432, 2011.

[14] A. Kirsch. Integration of programming and learning in a control language for autonomous robots performing

everyday activities. *Learning*, 2008.

[15] M. Krötzsch, F. Simančík, and I. Horrocks. Description logics. *IEEE Intelligent Systems*, 29:12–19, 2014.

[16] L. Kunze, T. Roehm, and M. Beetz. Towards semantic robot description languages. In *Proceedings - IEEE International Conference on Robotics and Automation*, pages 5589–5595, 2011.

[17] C. Matuszek, J. Cabral, M. Witbrock, and J. Deoliveira. An introduction to the syntax and content of Cyc. *Proceedings of the 2006 AAAI Spring Symposium on Formalizing and Compiling Background Knowledge and Its Applications to Knowledge Representation and Question Answering*, 3864(1447):44–49, 2006.

[18] D. McDermott, M. Ghallab, A. Howe, C. Knoblock, A. Ram, M. Veloso, D. Weld, and D. Wilkins. PDDL - The Planning Domain Definition Language, 1998.

[19] M. Naumann, M. Bengel, and A. Verl. Automatic Generation of Robot Applications Using a Knowledge Integration Framework. In *Robotics (ISR), 2010 41st International Symposium on and 2010 6th German Conference on Robotics (ROBOTIK)*, pages 1–8, 2010.

[20] I. Niles and A. Pease. Towards a Standard Upper Ontology. *The 2nd International Conference on Formal Ontology in Information Systems (FOIS-2001)*, pages 2–9, 2001.

[21] A. Nilsson, R. Muradore, K. Nilsson, and P. Fiorini. Ontology for robotics: A roadmap. *2009 International Conference on Advanced Robotics*, 2009.

[22] J. Persson, A. Gallois, A. Björkelund, L. Hafdell, M. Haage, J. Malec, K. Nilsson, and P. Nugues. A knowledge integration framework for robotics. *International Symposium on Robotics ISR*, pages 1–8, 2010.

[23] A. Preece and M. Gomez. Matching sensors to missions using a knowledge-based approach. *Proceedings of SPIE - The International Society for Optical Engineering*, 44(0), 2008.

[24] C. Schlegel and T. Haß ler. Robotic software systems: From code-driven to model-driven designs. In *International Conference on Advanced Robotics*, pages 1–8, 2009.

[25] M. Tenorth and M. Beetz. KNOWROB - Knowledge processing for autonomous personal robots. In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2009*, pages 4261–4266, 2009.

[26] M. Tenorth and M. Beetz. Knowledge Processing for Autonomous Robot Control. *Proceedings of the AAAI Spring Symposium on Designing Intelligent Robots: Reintegrating AI . Stanford, CA: AAAI Press.*, 2012.

[27] M. Tenorth and M. Beetz. KnowRob: A knowledge processing infrastructure for cognition-enabled robots. *The International Journal of Robotics Research*, 32(5):566–590, 2013.

[28] M. M. Tenorth. Knowledge Processing for Autonomous Robots. page 225, 2011.

[29] S. Zander, G. Heppner, G. Neugschwandtner, R. Awad, M. Essinger, and N. Ahmed. A model-driven engineering approach for ROS using ontological semantics. *CoRR*, abs/1601.03998, 2016.

[30] O. Zweigle, R. van de Molengraft, R. D'Andrea, and K. Häussermann. RoboEarth: connecting robots worldwide. In *Proceedings of the 2nd International Conference on Interaction Sciences: Information Technology, Culture and Human*, pages 184–191. ACM, 2009.