CHAPTER 3

Standard Ontologies and HRI

Sandro Rama Fiorini, Abdelghani Chibani,
Tamas Haidegger, Joel Luis Carbonera, Craig Schlenoff,
Jacek Malec, Edson Prestes, Paulo Gonçalves,
S. Veera Ragavan, Howard Li, Hirenkumar Nakawala,
Stephen Balakirsky, Sofiane Bouznad, Noauel Ayari,
and Yacine Amirat

CONTENTS

3.1 INTRODUCTION

While robots and humans might interact in a multitude of ways, communication seems to be one of the most important types. As in human–human communication, successful communication between humans and robots usually involves the sharing of symbols with a shared meaning. This is the case for traditional verbal communication, but also for the case in situations where a human communicates his or her intentions through clicks in a user interface or through commands issued to an application programming interface (API). In any case, robots have to be able to *represent* the meaning of those symbols in a computable, formal way. More importantly, humans have to be able to access and understand that representation in order to know how to communicate with the robot.

In the last 15 years, ontologies have emerged as one of the best techniques to represent information and meaning in computer systems. Since the early ages of artificial intelligence, it became clear that in order to understand humans, intelligent systems must be able to share with them a common representation of the world. There has been a multitude of approaches for doing that, where logic-based techniques came up on top. However, their high computational complexity and unfit modelling techniques limited the use of symbolic approaches in robots. New research during the 1980s and 1990s brought up the idea that intelligent systems, in order to be able to work in tandem with humans, must represent the world with logical theories modelled taking in to account the knowledge *shared* by the community in which the system is situated. These theories are what we call *ontologies*. In this context, ontologies are formal information artefacts that define the concepts of a domain as well as the relationships between these concepts. Ontologies can be used in knowledge sharing between communities of intelligent agents and humans, usually employed as a tool to provide meaning to information sharing protocols. Also, being logical theories, ontologies can be used as a knowledge source for implementing symbolic reasoning techniques.

In the case of robotics, an ontology can serve as a communication middle-layer between humans and robots. However, as groups of humans might communicate in different languages, there is nothing preventing distinct human/robot groups of employing distinct ontologies in communications. This creates communication islands akin to the information islands that prevent the integration of heterogeneous information systems. In order to help prevent such cases, the IEEE RAS Ontologies for Robotics and Automation Working Group (ORA WG) was formed with the notable objective of providing a consensus-based set of ontologies for the domain. Their aim was to link existing ISO, IEC, etc. standards, current research efforts and new regulatory frameworks to a generic Robotics and Automation Ontology [LAP+12]. The ORA WG is comprised of over 170 members, representing over 20 countries — a cross-section of industry, academia and government. The group has spent over four years to develop and publish the IEEE Standard 1872-2015 [IEE15], which specifies a set of high-level ontologies about robotics and related notions. Notably, the main component is the Core Ontology for Robotics and Automation (CORA), which formally defines the main notions of robotics and automation (R&A) as a whole. CORA and other ontologies are now being used by experts globally, both for supporting human–robot communication technologies and also for other areas of robotics. Currently, sub-groups of ORA WG are developing specific ontologies to complement IEEE 1872-2015, focusing on robot task representation and autonomous robots. We expect that the complete family of ORA ontologies will help ensure a common understanding among members of the community and facilitate more efficient integration and data transfer.

Our aim with this chapter is to give the reader a brief overview of IEEE 1872-2015 and efforts related to it. We start by discussing what ontologies are and why we need them. We then present the general structure the IEEE community groups that developed CORA and are developing new ontologies. CORA and related ontologies in IEEE 1872-2015 standard are presented next, followed by a brief overview of different works employing CORA and the current groups working on extensions to CORA. We finish with a discussion about the requirements for, and human–robot interaction (HRI) ontology based on, IEEE 1872-2015.

3.2 WHAT IS AN ONTOLOGY AND WHY IS IT USEFUL

In computer science and related technology domains, an ontology is considered a formal and explicit specification of a shared conceptualization [SBF98]. The conceptualization specified by an ontology, according to this point of view, encompasses the set of concepts related to the kinds of entities that are supposed to exist in a given domain, according to a community of practitioners. Thus, the main purpose of an ontology is to capture a common conceptual understanding about a given domain. Due to this, ontologies can be used for promoting the semantic interoperability among stakeholders, since sharing a common ontology is equivalent to sharing a common view of the world. Moreover, because ontologies specify the domain conceptualization in a formal and explicit way, this ensures that the meaning of every concept is rigorously specified and can be analyzed by humans and machines. Therefore, an ontology could be used as a common basis for communication between humans and machines. Finally, ontologies can also be viewed as reusable components of knowledge, since they capture the knowledge about a domain in a task-independent way. Thus, considering that the development of knowledge models is a notoriously difficult, time-consuming and expensive process, the adoption of ontologies promotes a more rational use of the resources in a project.

An ontology includes at least a set of terms and their definitions as shared by a given community, formally specified in a machine-readable language, such as first-order logic. These terms and definitions are structured in terms of

- **classes**, which stand for concepts at all granularities
- **relations**, which establish associations between concepts
- **formal axioms**, which constrain and add consistency rules to the concepts and relations

There are different kinds of ontologies and different ways of classifying them. In [Gua98], the author proposes four main classes of ontologies

- **Top-level ontologies**, which describe very general concepts like space, time, matter, object, event, action, etc., which are independent of a particular problem or domain
- **Domain ontologies**, which describe concepts of a specific domain
- **Task ontologies**, which describe generic tasks or activities
- **Application ontologies**, which are strictly related to a specific application and used to describe concepts of a particular domain and task

In [Obr10, PCF+13] the authors also mention a fifth kind of ontology called *core ontology*. Core ontologies can be viewed as mid-level ontologies, standing between top-level ontologies and domain ontologies. Core ontologies reuse concepts specified by top-level ontologies and specify new concepts that can be used in specific domains and tasks. They specify concepts that are general in a large domain but that are too specific for being included in a top-level ontology. In general, this kind of ontology specifies the most important concepts of a given broad domain [PCF+13].

3.2.1 Process of Ontology Development

The range of activities concerning the ontology development process, the ontology life cycle, the methods and methodologies for building ontologies and the tools and languages that support them is called *Ontology engineering*. Nowadays, there are several different methodologies that can be adopted for developing an ontology engineering process, including METHONTOLOGY [FLGPJ97], KACTUS [SWJ95], On-To-Knowledge [SSS04], DILIGENT [DSV+05], NeOn [SFGPFL12] and so on. Most of these methodologies specify sequences (or cycles) of activities that should be carried out for developing an ontology, including:

- Feasibility study, which is an assessment of the practicality of a proposed project.
- Knowledge acquisition, which is responsible for capturing the relevant domain knowledge from different sources.
- Conceptual modelling, whose goal is to structure the captured knowledge in a semi-formal ontology conceptual model;
- Axiomatization, which imposes a formal structure on the modelled domain knowledge (usually adopting a representation based on First Order Logics);
- Implementation, whose purpose is to implement the ontology in a computer-processable representation format, such as OWL.[*]
- Evaluation, which evaluates the developed ontology for ensuring its quality.

AU: Please expand acronym OWL.

---

[*] www.w3.orgTRowl-features

- Maintenance, whose purpose is to fix errors, and keep the quality of the ontology when it is modified, by inclusion of novel knowledge or by updating some definitions.

For further details, in [SSS09] the authors provide a deep discussion about ontology engineering in general.

The fundamental objective of domain-specific ontology development is to identify, develop and document the common terms and definitions within a sub-field, so that they can serve as a common reference for the R&D community. It needs to be completed at a very sophisticated way to fulfil its goals, since only high-quality ontologies can be hoped to become cornerstones of the community effort. High quality, high profile ontologies are called Exemplary Ontologies (ontologydesignpatterns.org/wiki/Ontology:Main). The general methodology for building ontologies specifies certain modelling principles that need to be followed in order to assure that the finished tool commits to the shared knowledge. It needs to ensure the mutual agreement among stakeholders and increase the potential of reuse of knowledge, allowing smooth data integration upwards and downwards as well. When it is targeted to develop exemplary ontologies, the following attributes need to be considered [pro14]:

- the ontology must be well designed for its purpose
- shall include explicitly stated requirements
- must meet all and for the most part, only the intended requirements
- should not make unnecessary commitments or assumptions
- should be easy to extend to meet additional requirements
- it reuses prior knowledge bases as much as possible
- there is a core set of primitives that are used to build up more complex parts
- should be easy to understand and maintain
- must be well documented

3.3 OVERVIEW OF THE IEEE EFFORT ON ROBOT ONTOLOGIES

Recognizing the advantages that a standard ontology for R&A would bring to the field, the IEEE Standard Association's Robotics and Automation Society (RAS) created the Ontologies for Robotics and Automation (ORA) Working Group in 2011. The goal of the group was to develop a standard to provide an overall ontology and associated methodology for knowledge representation and reasoning in R&A, together with the representation of concepts in an initial set of application domains. It achieved this goal in 2015 with the publication of the IEEE 1872-2015 Standard Ontologies for Robotics and Automation.* The group was composed of 175 members representing 23 countries and was made up of approximately 50% educational institutions, 25% private companies and 25% government

---

* http://standards.ieee.org/findstds/standard/1872-2015.html

entities. With the release of the standard, the ORA working group completed its task and was required to disband. However, many of the working group members remain involved in this work by focusing on sub-groups, as described later in this chapter.

The IEEE 1872-2015 standard provides a unified way of representing knowledge and provides a common set of terms and definitions, allowing for unambiguous knowledge transfer among any group of humans, robots and other artificial systems. It was awarded the prestigious Emerging Technology Award by the IEEE Standards Association in December 2015. The standard was also mentioned in the "The National AI Research and Development Strategic Plan" released by President Obama in October 2016.* This strategic plan focuses on the role of AI, machine learning, automation and robotics in addressing complex national problems.

One of the main parts of IEEE 1872-2015 is the Core Ontologies for Robotics and Automation (CORA). CORA will be described in greater detail later in this chapter, but as an introduction, it aims to describe what a robot is and how it relates to other concepts. It defines four big broad entities: robot part, robot, complex robot and robotic system. The term robot may have as many definitions as authors writing about the subject. The inherent ambiguity in this term might be an issue when one needs to specify an ontology for a broad community like ours. We acknowledge this ambiguity as an intrinsic feature of the domain and, therefore, we decided to elaborate a definition based purely on necessary conditions, without specifying sufficient conditions. Thus, it is ensured that CORA covers all entities that the community actually considers as a robot, at the cost of classifying some entities as robots that may be counterintuitive to some roboticists. Also, the concepts in our ontology could be specialized according to the needs of specific sub-domains or applications of R&A.

CORA was developed to be a high-level standard in which domain-specific efforts could build from. The approach was to define concepts in CORA that were generic to all robot domains and then these domains could specialize these concepts to address their specific information requirements. When the working group that developed CORA was created, based on the interests of the working group members, we developed Figure 3.1 to show the sub-groups that we expected to emerge to specialize the concepts represented in CORA.

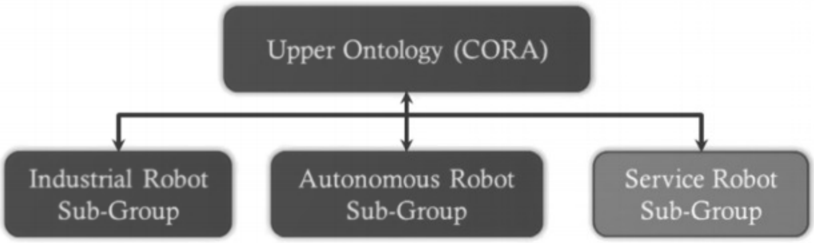


FIGURE 3.1 Initial organization of ORA groups and sub-groups. The boxes represented in dark grey show on-going efforts, the one in light grey is currently inactive. We expect additional domain ontology sub-groups to emerge.

---

* www.whitehouse.gov/sites/default/files/whitehouse_files/microsites/ostp/NSTC/national_ai_rd_strategic_plan.pdf

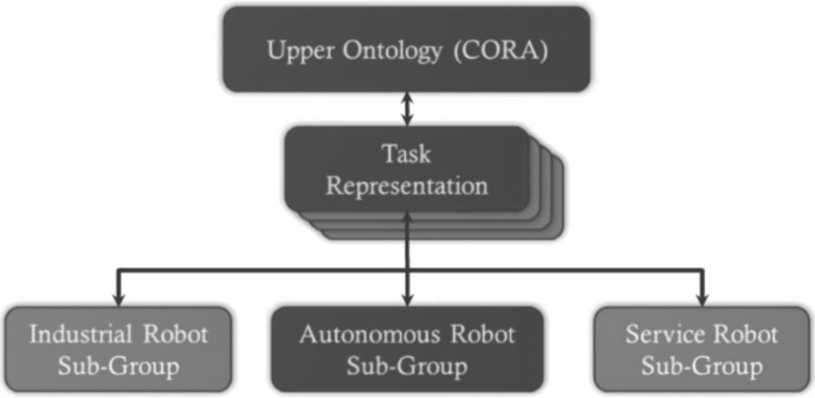


FIGURE 3.2 Present organization of ORA groups and sub-groups.

Over time, we realized that there was a level missing in the above structure. The Upper Ontology (CORA) provided very high-level concepts and the domain ontologies provided concepts that were very specific to individual domains. What was missing was a middle level that contained cross-cutting concepts that were applicable to many, if not all, robot domains. Examples of these middle-level, cross-cutting concepts could include representation of sensors and perception data, tasks and environmental objects.

The industrial robot ontology sub-group, after meeting for a few months, realized that there was a greater need for a cross-cutting representation of task information (a middle-level ontology) than for a detailed industrial robot ontology. Because of this, the group changed its name, and its scope, to focus on Robot Task Representation. More specifically, this group will develop a broad standard that provides a comprehensive ontology for robot task structure and reasoning. While initially focusing on industrial tasks, the resulting standard is expected to be applicable across most, if not all, robot domains. Hence, Figure 3.1 morphed into Figure 3.2. More details about the Robot Task Representation group are discussed later in the chapter.

3.4 ONTOLOGIES IN IEEE 1872-2015

The IEEE 1872-2015 standard is composed of a collection of ontologies covering different general aspects of R&A. These ontologies provide definitions for notions such as robot, robotic system, positioning, interaction and so on. Each ontology is specified in the language SUI/KIF, a first-order language, extending concepts and relation of Suggested Upper Merged Ontology (SUMO) top-level ontology. The ontologies are also meant to be specialized in specific domain and applications ontologies in different areas of robotics. In this section, we describe the general aspects of each ontology in IEEE 1872-2015.

3.4.1 Preliminaries: SUMO

The majority of the entities defined in the standard are specializations of the SUMO concepts and relations. SUMO is a vast top-level ontology developed as part of an IEEE-sponsored effort to create a standard top-level ontology [NP01]. While SUMO never became an actual standard, its flexibility and extensive vocabulary of terms made it one of the main top-ontologies available in the literature. It includes formal theories about processes, spatial relations, temporal relations, information objects and so on. Also, SUMO has been extended along the years to address specific domains, such as engineering and the military.

Before presenting the specifics of IEEE 1872-2015, let us consider some basic aspects of SUMO. SUMO divides all entities that exist into two big groups: physical and abstract (Figure 3.3). Physical entities exist in space-time. Abstracts do not exist in space and/or time, and include mathematical and epistemological constructs. Physical entities are separated into *objects* and *processes*. An object is an entity that has spatiotemporal parts. This concept corresponds to the notion of ordinary objects, but also include physical regions. Processes, on the other hand, are physicals that occurs in time and that are not objects. Instances of processes are physicals such as events, manufacturing processes, movements, cognitive processes and so on.

On the abstract side of the taxonomy, SUMO has concepts such as quantity, attribute, class and proposition. Quantities are akin to numeric properties one can use to characterize other entities. Attributes are lexical properties. Class and proposition concepts give SUMO the ability to represent facts about fact (e.g., metamodeling). For instance, a proposition in SUMO is an abstract entity that represents a thought. For example, the sentence "the book is on the table" expresses the proposition that there is a book situated on top of a particular table. The sentence in Portuguese "o livro está sobre a mesa" is a different sentence that expresses the same proposition. SUMO allows one to capture the materialization of a proposition as instances of Content-bearing Object, a sub-class of Object that represents one or more propositions, such as the two sentences above.

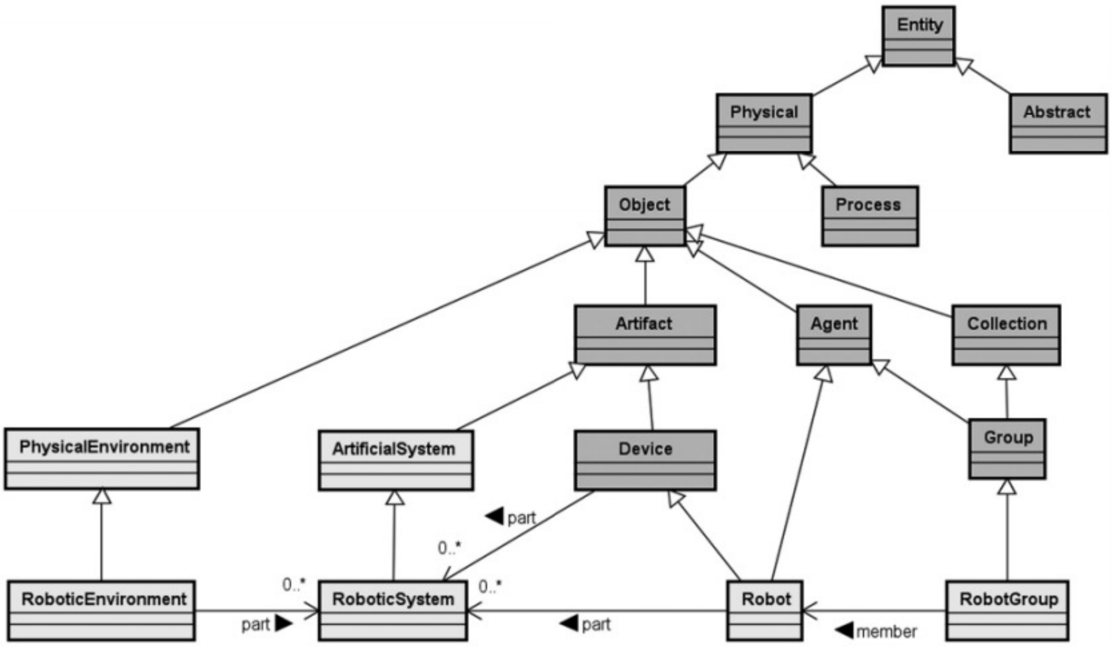


FIGURE 3.3 Taxonomic structure (arrows) of the main concepts in SUMO (light grey boxes) and CORA (dark grey boxes).

3.4.2 CORA: The Core Ontology for R&A

As discussed in Section 2, a *core ontology* is a domain ontology that aims at defining the most general concepts in a broad domain, such as medicine or geology. The core ontology for robotics and automation is the main ontology in IEEE 1872-2015. As its name implies, it aims at providing formal definitions for central notions in R&A. The focus on *notions* rather than simply on *terminology* is not an accident: CORA does not try to accommodate any possible terminology in R&A, but rather it aims at making explicit concepts and relations behind different terminologies.

CORA gravitates around three main concepts: *robot*, *robot group* and *robotic system* (Figure 3.3). As we mentioned earlier in this chapter, the term *robot* may have as many definitions as people using it. CORA acknowledges this fact by defining robot in a very general way, allowing for specific subontologies to provide the sufficient conditions for specific kinds of robots. CORA only states that *robots are agentive devices*, designed to perform purposeful actions in order to accomplish a task. According to SUMO, an instance of Device is an artefact (i.e. a *physical object product of making*), which has the intended purpose of being a tool in a class of processes. Being a device, robot inherits from SUMO the notion that devices have parts. Therefore, CORA allows one to represent complex robots with robot parts. Also, the concept Robot inherits from the Device the necessity for having a described purpose. A robot is also an *Agent*. SUMO states that agent is "*something or someone that can act on its own and produce changes in the world*". Robots perform tasks by acting on the environment or themselves. Action is strongly related to agency, in the sense that the acting defines the agent. In some cases, the actions of a robot might be subordinated to actions of other agents, such as software agents (bots) or humans.

Robots can form robot groups. A *robot group* is also an agent in the sense that its own agency emerges from its participants. The concept has been left underspecified in the standard, and can be used to describe robot teams or even complex robots formed by many independent robotic agents acting in unison.

Robotic systems are systems composed of robots (or robot groups) and other devices that facilitate the operations of robots. A good example of a robotic system is a car assembly cell at a manufacturing site. The environment is equipped with actuated structures that manipulate the car body in a way that the industrial robots within the system can act on it. Finally, an environment equipped with a robotic system is a *robotic environment*.

Apart from these three main concepts, CORA also defines the notion of *robot interface*. A robot interface is a kind of device that represents the interface of the robot with the external world. A robot has one and only one interface that aggregate all parts that the robot uses to sense, act and communicate. The robot interface can be seen as a virtual device, in the sense that it might not coincide with a particular component. Any robot interface must have at least a sensor, an actuator or a communication device (such as a networks card).

Autonomy is a core notion in R&A, yet one of the hardest to define in precise terms. CORA incorporates the idea that the degree of autonomy of a robot can only be defined in relation to the task at hand [HMA03]. Thus, CORA does not define specific *classes* of autonomous robots. It rather defines specific types of *agent* roles that a robot can have when participating in a process where it is the agent. The *agent* role is a binary predicate (which is related to, but different from the concept Agent) relating processes with the object that acts as the agent of the process. CORA defines five sub-relations of *agent*, namely, *fullyAutonomousRobot*, *semiAutonomousRobot*, *teleoperatedRobot*, *remoteControlledRobot* and *automatedRobot* (i.e. automaton). Each sub-relation qualifies the role of a particular robot in the process. CORA does not provide sufficient conditions for these relations but clarifies that the same robot can assume different roles in different processes.

3.4.3 CORAX: CORA Extended

Even SUMO and CORA together do not cover all aspects of R&A. CORAX is another ontology part of IEEE 1872-2015 [FCG+15] that specifies some concepts that are not present or clear in SUMO and that are too general for CORA. CORAX provides concepts such as design, environment and interaction.

Design is an important concept in engineering, especially in manufacturing. In R&A, the concept is frequently related to industrial robotics, where robots perform the job of building artefacts. Those robots have to represent the design of the artefacts they are building in order to coordinate their actions. In CORAX, a design is a kind of proposition. That is, it is an abstract concept that can be materialized in content-bearing objects, such as manuals and blueprints. Furthermore, artefacts are associated to design, so that one should expect that an artefact realizes the design. Such notions have been further extended in the architecture ontology developed by the AuR group (Section 6.1).

Furthermore, the properties of the object must be expressed in its design. For instance, the design of a phone is about an (*idealized*) phone that is materialized in the individual phones built on that design. CORAX specifies the ideal object as a separate entity called a *Design Object*, which specifies the idealized object that is the *content* of a design. The ideal phone has ideal properties, such as ideal weight and shape. These are related to real properties but have different pragmatics in modelling and reasoning. While SUMO provides two main relationships to represent properties, namely *attribute* and *measure*, CORAX specifies two analogue relationships, namely *designAttribute* and *designMeasure*. Both sets of properties, the physical and the abstract, can be used with any quantity type of attribute type already present in SUMO. In this way, we can specify that, for instance, an idealized phone (an instance of *Object Design*) has a *design shape* and a *design weight*. The properties of the design object and those of the artefact may correlate, but CORAX does not provide a theory of how that correlation occurs.

CORAX also includes the notion of *physical environment* in order to support specification of *robotic environments*. An *environment* is intuitively composed of a physical region, plus eventual objects that characterize the environment. In addition, the definition of physical environment depends on the presence of a landmark from which it is possible to define the main region of an environment. Landmarks may or may not be located within the region of interest of the environment. For instance, an office room environment depends on the physical configuration of its walls, which are located in the environment. But we can also define an arbitrary environment consisting of a cube in outer space that depends on Earth as a landmark, even if the planet itself is not part of the environment.

As CORA defines the concept Robotic System, it becomes necessary to define what is a system. CORAX specify the concept Artificial System as an artefact formed from various devices and other objects that interact with each other and with the environment in order to fulfil a purpose. This requires a basic definition of *interaction*. CORAX introduces the notion of interaction as a process in which two agents participate, where an *action* generated by one agent causes a *reaction* by the other. More specifically, an interaction process is composed of two sub-processes corresponding to action and reaction. The action sub-process initiated by *x* on *y* causes a reaction sub-process, where *y* acts upon *x* (Figure 3.4).

AU: We have inserted citation for Figures 3.4 and 3.10 according to the placement. Please check and confirm.

Finally, CORAX defines certain general classes of robot–robot and human–robot communication. Both cases are specific types of content bearing processes (i.e. a process that carries a proposition).

3.4.4 RPARTS: The Parts Ontology

RPARTS is a sub-ontology of CORA that specifies general notions related to some kinds of robot parts. According to CORA, robots are (agentive) devices *composed of* other devices. A myriad of devices can be robot parts, and CORA cannot determine in advance what *kinds* of devices can or cannot be robot parts. Notice that this is an issue that arises at the *conceptual level*. This is a consequence of the "open-ended" nature of robots, whose designs are only constrained by human needs, human creativity and available technological resources. Therefore, a type of device that has never been considered as a potential robot part can be used as a robot part by some future designer. An ontology for R&A, as CORA is, must take this issue into account.

Furthermore, there is another issue regarding the notion of robot parts that arises at the *instance level*. None of the objects that can be classified as robot parts are *essentially* robot parts, since they can exist by themselves when they are not connected to a robot (or when they are connected to other complex devices). For instance, a power source is essentially a device, and we cannot consider a power source as a sub-class of the class of robot parts, because this would imply that all instances of power sources are always robot parts. This is not true, as a specific instance of power source can be dynamically considered as a part of different complex devices during different specific time intervals. Due to this, CORA

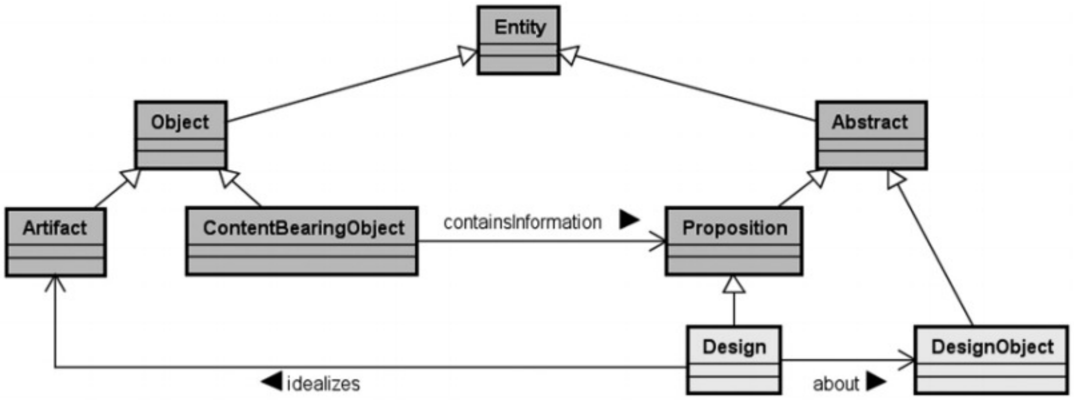


FIGURE 3.4 Entities associated with design in CORAX.

assumes that the notion of "robot part" is a *role* (in the sense previously discussed) that can be played by other devices.

In the earlier proposals of CORA [PCF+13], the notion of robot parts was considered as a *class*, whose instances are not *essentially* instances of it. Thus, instances of robot parts could cease to be robot parts, without ceasing to exist. In this sense, for example, an instance of a power source that is considered as a robot part at a given moment (when it is connected to a robot) could cease to be a robot part in another moment without ceasing to exist (as an instance of power source). Thus, *Robot part* was considered as an *anti-rigid* universal, in the sense of [Gui05]. The ontology pattern proposed in [PCF+13] was developed accordingly, inspired by [Gui05]. It represents how a specific instance of a specific kind of device (e.g. power source) could be classified as a robot part.

This pattern becomes complex when we take into account the principles advocated in [Gui05]. According to these frameworks, an anti-rigid class (e.g. robot part) cannot subsume a rigid one (e.g. power source). Considering this principle, for each rigid class *c* that can play the role of a robot part, we must create another specific anti-rigid class (a specific role) that will be subsumed by both *c* and *Robot Part*. For example, an instance of the rigid class *Wheel* only becomes a robot part when it is attached to a particular robot. Given this condition, it becomes a member of the more specific class (e.g. "*Wheel as Robot Part*"), which is subsumed by the rigid class *Wheel* and the anti-rigid class *Robot Part* (see [PCF+13] for further details.

The representation of robot parts in the final proposal of CORA was changed, mainly because the modelling pattern proposed for representing robot parts results in domain models that are overwhelmingly complex. Some classes that must be created in order to maintain the consistency of the model do not fit well into the domain conceptualization and the resulting complexity is hard to manage. Therefore, this modelling pattern could hinder the broad adoption of the ontology in the domain. Another factor leading to the revision was that it is not clear how to fit the dynamical behaviour that is expected from roles in the framework of SUMO. The modelling of roles adopted in [Gui05] relies on the notion of *possibility* (a *modal* notion). However, as pointed out in [OAH+07], the treatment of possibilities in SUMO is not clear.

Robot part is then a relationship between a given device *d* and a robot *r*, indicating that *d* is playing the role of robot part when it is connected to *r*. RPARTS defines four specific roles for robot parts.

Robot Sensing Part: responsible for sensing the surrounding environment. Formally, robot sensing parts must be measuring devices connected to the robot. A measuring device, according to SUMO, is *any device whose purpose is to measure a physical quantity.* For example, a *laser sensor* can play the role of robot sensing part when connected to a robot.

Robot Actuating Part: responsible for allowing the robot to move and act in the surrounding environment. Formally, robot actuating parts must be devices that are instruments in a process of robot motion, which is any process of movement where the robot is the agent and one of its parts is acted upon.

Robot Communicating Part: responsible for providing communication among robots and humans by allowing the robot to send (or receive) information to (or from) a robot or a human.

Robot Processing Part: responsible for processing data and information. Formally, robot processing parts must be processing devices connected to the robot. A processing device is any electric device whose purpose is to serve as an instrument in a sub-class of a computer process.

It is important to emphasize that although these different types of robot parts are modelled as relations between specific devices and robots, they are intended to behave as roles.

This modelling choice also provides interesting modularity characteristics. It keeps CORA as a minimal core of high-level concepts that provides the structure to the domain without going deep into details regarding the myriad of different devices that could play the roles specified here. In this sense, this structure of roles can be viewed as an interface (in the sense of *object-oriented programming paradigm*) that can be implemented in different ways. Naturally, this schema poses the need for sub-ontologies to define the taxonomies of devices that can play the roles specified in CORA, such as an *ontology of sensors*, *ontology of grippers* and so on.

3.4.5 The POS Ontology

Also included in the IEEE 1872-2015, the position (POS) ontology [CFP+13] specifies the main concepts and relations underlying the notions of *position*, *orientation* and *pose*. These are essential for dealing with information about the relationship between a robot and its surrounding space. In this section, we summarize the main concepts relating to positional information.

POS defines two kinds of positioning information: *quantitative* and *qualitative*. In the quantitative case, a position is represented by a *point* in a given coordinate system. In the qualitative case, a position is represented as a *region* defined as a function of a reference object. For instance, one can describe a robot as being positioned at the coordinates (x, y) in the global coordinate system, or that the robot is positioned *at the front of the box*, where *front* comprises a conical region centered on the box and pointed forwards.

POS states that a *position* can be attributed to a (physical) *object*. In this sense, when we say that "a robot x is positioned at y", this means that there is a *measure* (SUMO binary relation) that relates a given "robot x" to a *position measurement y*.

Position measurements are *physical quantities* that can be *position points* or *position regions*. A position point denotes the *quantitative* position of an object in a coordinate system. More specifically, position points are always defined in a single coordinate system. A position region is an *abstract region* in a *coordinate system* defined with reference to a series of position points.

A *coordinate system* is an *abstract* entity that is defined in relation to *a single reference* object, i.e. there is an object that is the reference for each coordinate system. For instance, the local coordinate system of a robot is referenced by the robot itself. Additionally, the reference object does not need to be at the origin of the coordinate system. This ontology does not commit to a particular kind of coordinate system. It can be stated, however, that a coordinate system defines at least one dimension in which points get their coordinate values. A fundamental aspect of coordinate systems is the notion of *transformation*, which maps position points in one coordinate system to position points in another coordinate system. Transformations can be composed, generating new transformations. In POS, an object can display multiple positions in different coordinate systems only if there is a transformation that can map between the two. In addition to that, coordinate systems are related through *hierarchies* (i.e. trees). We say that a given coordinate system c_1 is the parent of a coordinate system c_2 if there is a transformation t_1 that maps the points of c_1 to points in c_2, and there is a transformation t_2 that maps the points of c_2 to points in c_1. According to this, if two coordinate systems share a parent node in the hierarchy tree, there is a transformation between them. Usually, an agent chooses a coordinate system as the global reference frame that constitutes the *global coordinate system* (GCS) for that agent. This GCS can be *arbitrarily* chosen and does not have reference to a particular coordinate frame. *Local coordinate systems* (LCS) are defined in relation to GCS by hierarchical links. This hierarchy is arbitrary, in the sense that it can be defined by the designer or agent.

Besides the quantitative position, POS also provides concepts about qualitative positions that are defined in terms of position regions. Example of qualitative positions are "left of", "in front of", "on top of" and so on. These expressions define regions in relation to a reference object o_r in which other objects are placed. More specifically, a *position region* is composed of poses in the coordinate system generated by a *spatial operator* on the reference object. The spatial operator is a *mathematical function* that maps reference objects to regions in a coordinate system in arbitrary ways.

POS also allows for the specification of *relative positions* between objects and a given reference object. In general, this kind of information is represented through *spatial relations* that hold between objects. An example is the relation leftOf(*o*, *or*), which represents that the object *o* is positioned to the left of the object o_r. This kind of relation can be defined in POS using the notions of *relative position* and *spatial operator*. For example, the relation leftOf(*o*, *or*) holds when there is a qualitative position *s* (a position region) that was generated by the spatial operator leftOfOp over the reference object o_r, and the object *o* has the relative position *s* regarding o_r. Through this mechanism, POS provides the semantics for spatial relations like "to the left of".

AU: Should "or" in "leftOf(o, or)" be "o_r", to match object "o_r" at conclusion of sentence?

The usual notion of orientation is similar to position as far as its conceptual structure is concerned. An object can have a quantitative orientation defined as a value in an orientation coordinate system, as well as a qualitative orientation defined as a region in relation to a reference object. For example, orientation is used in the phrase "the robot is oriented at 54 degrees"; the orientation value in this case is 54 in the circular, one-dimensional coordinate system of a compass. On the other hand, orientation regions capture a less intuitive notion. The expression "the robot is oriented to the north of the Earth" allows for interpretations where the robot has a range of possible orientation points around 0 degrees. Thus, we can represent "north" as a region (or interval) in the one-dimensional, circular compass coordinate system that overlaps with the general orientational extension of the object.

In POS, a position and an orientation constitute a pose. The pose of an object is the description of any position and orientation simultaneously applied to the same object. Often, a pose is defined with a position and an orientation referenced to different coordinate systems/reference objects. In addition, since objects can have many different positions and orientations, they can also have many different poses.

3.5 APPLICATIONS OF CORA TO DATE

3.5.1 CORA as Part of a Robotic Skill Ontology

Modularity of robotic ontologies and the possibility of building a specific application-related ontology from existing, well-defined building blocks is a long-looked-for property, enabling faster development of cognitive robotic systems and easier debugging. The case described below is an illustration of this approach, exploiting IEEE CORA as one of its building blocks.

The Rosetta suite of ontologies describing industrial robotic skills has been developed in a series of EU projects for almost a decade. The individual ontologies serve different purposes. The core ontology, rosetta.owl, is a continuous development aimed at creating a generic ontology for industrial robotics. It is described in [SM15] and is available on the public ontology server http://kif.cs.lth.se/ontologies/rosetta.owl.

The ontology hierarchy is depicted in Figure 3.5, where arrows denote ontology import operation.

We use either the QUDT (Quantities, Units, Dimensions and Types) or the OM ontologies and vocabularies in order to express physical units and dimensions. The core Rosetta ontology focuses mostly on robotic devices and skills, as described in [HMN+11]. According to it, every device can offer one or more skills, and every skill is offered by one or more devices. Production processes are divided into tasks (which may be considered specifications), each realized by some skill (implementation). Skills are compositional items: there are primitive skills (non-divisible) and compound ones. Skills may be executed in parallel if the hardware resources and constraints allow it. The rosetta.owl ontology is

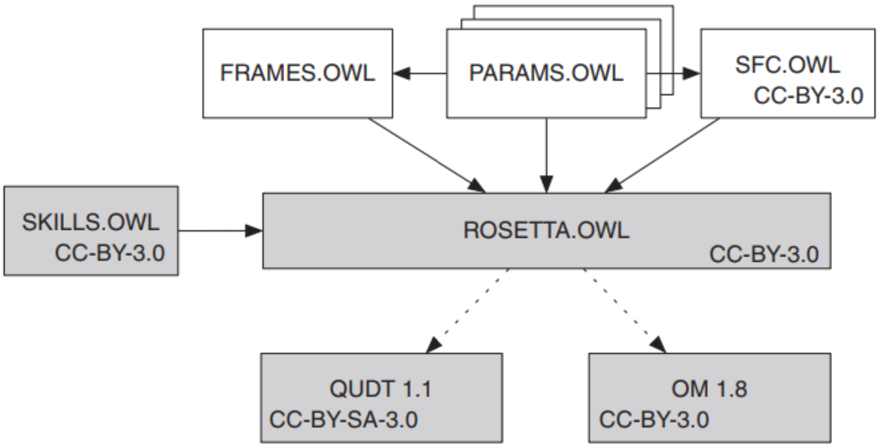


FIGURE 3.5 The Rosetta ontologies.

accompanied by topical ontologies describing behaviours (sfc.owl) as finite state machines, robotic skill parameterizations (params.owl) and feature and object frames (frames.owl). All these ontologies are available from the ontology server above.

The definition of skills has been based on the so-called production triangle: product, process, resources (PPR) [CDYM+07]. The workpieces being manufactured are maintained in the *product*-centered view. The manufacturing itself (i.e. the *process*) is described using concepts corresponding to different levels of abstraction, namely tasks, steps and actions. Finally, the *resources* are materialized in devices (capable of sensing or manufacturing). The central notion of *skill* links all three views and is one of the founding elements of the representation.

Due to the growing complexity of those ontologies and the availability of external upper ontologies like IEEE standard CORA [PCF+13], we have decided to refactor this structure into one enabling modular addition of new skills (see Figure 3.6). In this new structure, support ontologies (frames, params, sfc) have been moved to configuration.owl and coordination.owl, for separation of concerns (according to the 4C suggestion: Computation, Communication, Configuration, Coordination) and much easier maintenance. The details of this solution are found in [JMN16]. The refactored ontologies are also available from our knowledge server.

The particular advantage of using CORA in this case consists of anchoring the concept of the (industrial) robot pose in terms already defined in the position-related part of CORA. This way, any other robotic knowledge-based system also based on the common standard ontology IEEE 1872-2015 will be automatically aligned with Rosetta, at least with

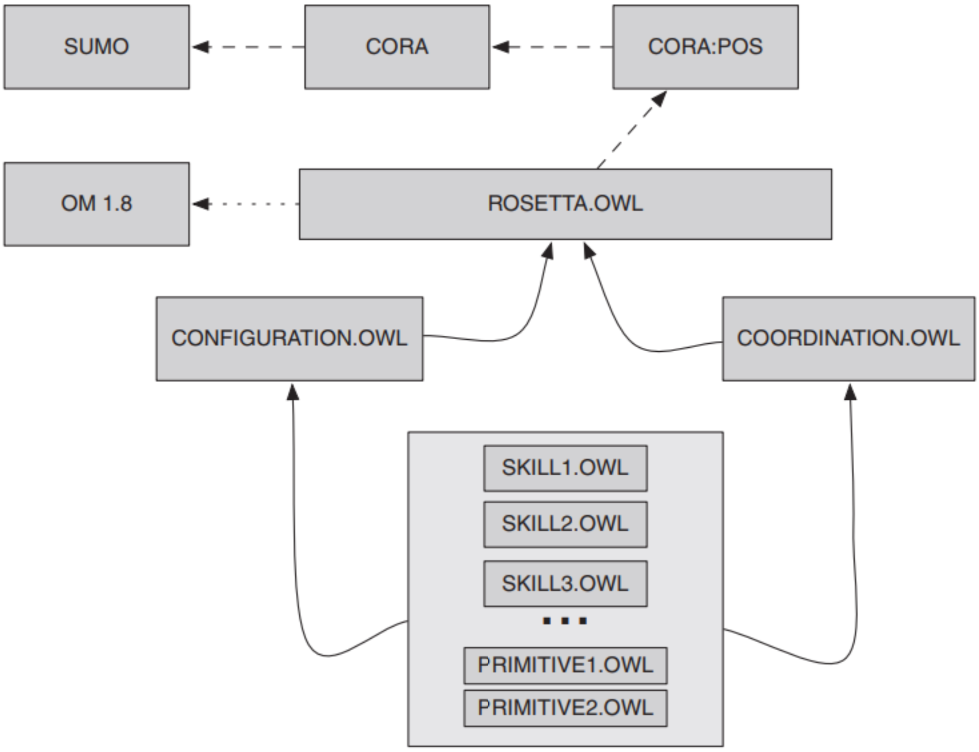


FIGURE 3.6 The refactored robotic skill ontologies.

respect to its idea of position. This should enable easier transfer of knowledge about robotic skills between different systems, possibly in an automatized manner.

3.5.2 The Cargo Delivery Scenario

CORA has been used and tested in very specific scenarios. One of them was a toy scenario where two robots with different physical and sensing capabilities communicated and coordinated to deliver cargo (a simple pen) to a human, i.e. the human solicits the cargo through a user interface and the robots coordinate themselves to pick and deliver the cargo (see Figure 3.7). To perform this task, all players should have the same communication language which must be formal and unambiguous. Otherwise, they will not be able to understand each other and, consequently, they will not attain their goal.

For this scenario, we used two robots from different manufacturers: an Aldebaran NAO H25 (aka manipulator) and a Pioneer 3DX (aka transporter). The manipulator had the cargo to be delivered to the human by request through a custom user interface. As the manipulator had its mobility limited to short distances due to its battery autonomy and speed, it could not deliver the cargo directly to the human, who could have been situated anywhere in the environment. However, it could manipulate the cargo by grabbing, lifting and/or dropping it. On the other hand, the transporter could move over long distances at higher speed than the manipulator and also could carry considerable payloads. However, it was not able to manipulate objects and had limited sensing capabilities (only range finding sensors).

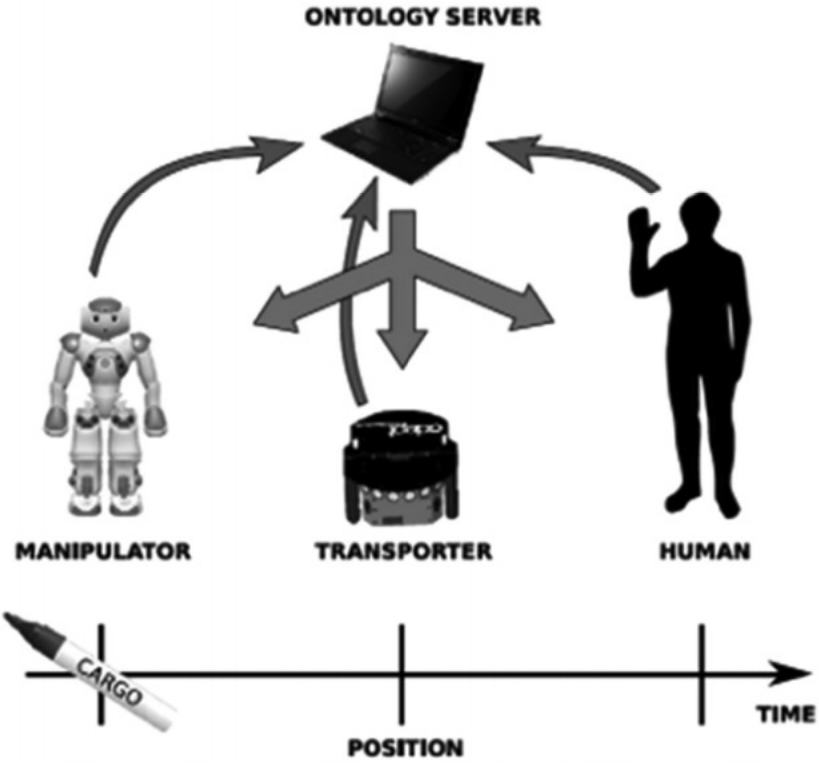


FIGURE 3.7 General organization of the system.

CORA concepts were represented using OWL and loaded into an ontology server written in Java. This server runs on a specific machine in a centralized architecture and it is responsible for managing all messages exchanged by human and robots. Thus, all server clients receive/send ontology information about themselves, which are used, for instance, to determine the relative positioning among the robots to allow the transporter to align itself and receive the cargo from the manipulator. Hence, we used the POS ontology, included in IEEE 1872-2015, to define the position and orientation of a player in a coordinate system to allow the spatial operations to determine the visibility, proximity, placement and relative orientation of a player in relation to another.

The experiment started with humans soliciting the cargo through a mobile phone application. This information was received by the manipulator and transporter, which first determined who had the cargo or had the ability to get the cargo. In this case, the manipulator had the cargo in its hands and sent this information to the system. The transporter started searching the manipulator in order to receive the cargo. When the manipulator was found, the transporter aligned itself to allow the manipulator to put the cargo on its shipment compartment. When this procedure was done, the transporter moved towards the human to complete the task. More information about this work can be found in [JRM+15].

Although this scenario is simple, it is complex enough to test our ontological framework. Ontology-based communication played a fundamental role. It endowed all players with a shared vocabulary with explicit semantics that allowed an unambiguous knowledge transfer even though using robots from different manufacturers.

3.5.3 Cora as Part of a Surgical Robot Ontology

Ontologies are widely used by the medical community to model knowledge. By using them, clinical terminology is clear and explicit to the community [GT14]. SNOMED-CT* © is nowadays the major reference in health terminology, where clinical terms are defined.

In surgeries, ontologies do exist with special focus on computer-assisted surgeries, Surgical Workflow (SWOnt ontology) [NJS+09], for assessment studies in human–computer interaction [MJD+15], and laparoscopic surgery (LapOntoSPM) [KJW+15], with application to situation interpretation, in other words, the recognition of surgical phases based on surgical activities. The latter case is of special use for HRI, when robots, surgeons, nurses and other medical staff and engineers co-work in the operating room. In fact, complete awareness of surgical procedures is needed for all agents.

Robotic ontologies with application to surgeries were recently proposed in the literature, applied to neurosurgery [PNDM+14] and orthopaedics (OROSU) [Gon13, GT15]. The latter applied CORA within the recently available IEEE 1872-2015 standard [Mad15, FCG+15]. In the following, a brief presentation of OROSU is presented focusing on a surgical procedure workflow for a proper interaction between the robot and medical staff in the operating room.

OROSU was developed with the integration of CORA [Mad15, FCG+15] and biomedical/human anatomical ontologies, sourced from *NCBO BioPortal* [NSW+09] and

---

* www.ihtsdo.org/snomed-ct

Open Biological and Biomedical Ontologies [SAR+07]. For tasks (surgical procedures) definition, and also as an engine to process the ontology, i.e. for reasoning, *KnowRob* [TB13] was used with success.

In the OROSU application presented in [GT15], robotic hip surgery was used to test the ontology for surgical procedure representation and reasoning. Figure 3.9 depicts an example for a complete surgical procedure definition (robotic bone tracking using an ultrasound probe) and the knowledge represented therein. This representation is suitable for a proper HRI between the medical staff and the robot, both working with the operating room ICT infrastructure. Figure 3.8 depicts an excerpt of the knowledge related to medical devices used in OROSU. For example, the sensor data is obtained using *CTimaging* or *USimaging* to gather 3D point clouds and then obtaining the 3D model of the bone.

In conclusion, the presented application is important to show the application of CORA to a complex HRI scenario. Using CORA and OROSU, robotic surgical procedures can be

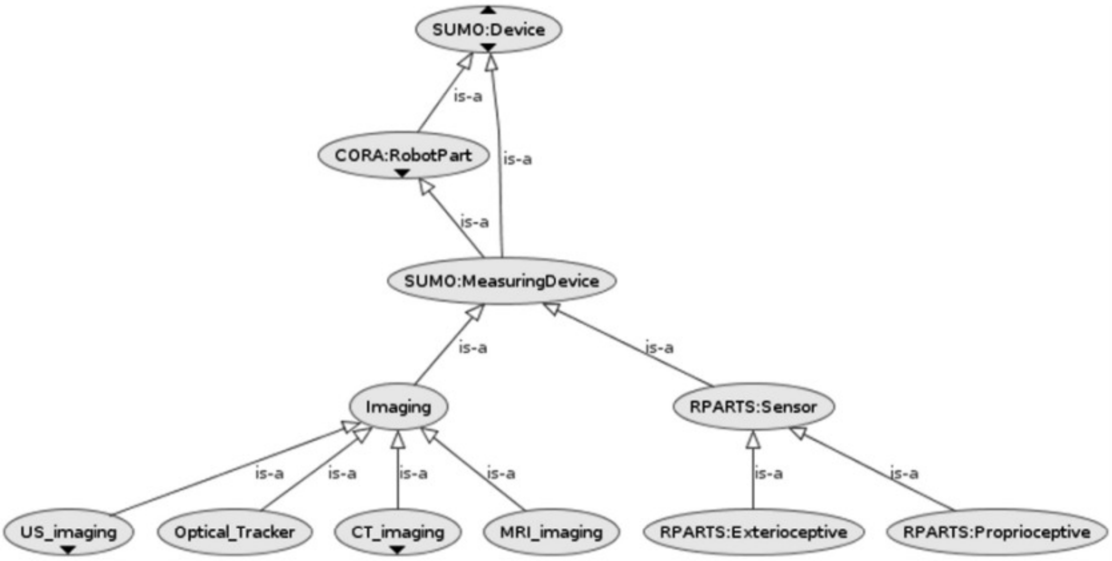
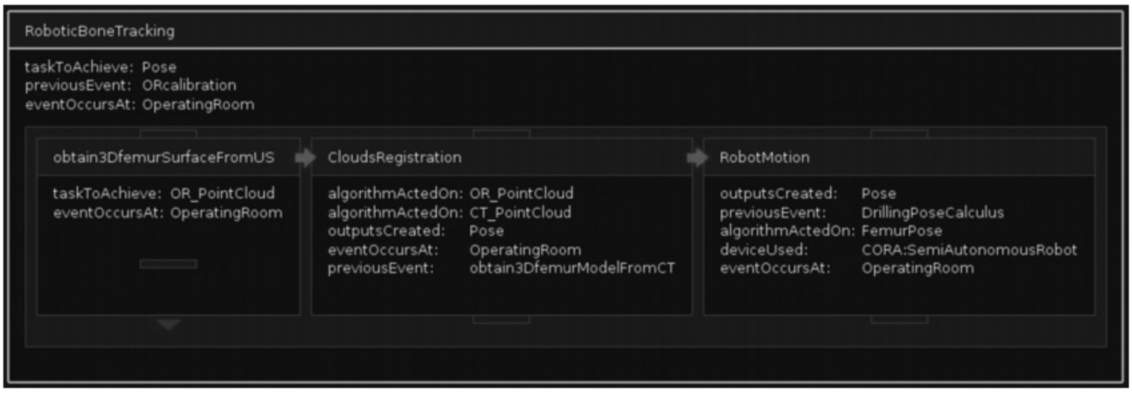


FIGURE 3.8 Some measuring devices defined in the ontology, relating SUMO, CORA and OROSU.



RoboticBoneTracking

taskToAchieve: Pose
previousEvent: ORcalibration
eventOccursAt: OperatingRoom

obtain3DfemurSurfaceFromUS

taskToAchieve: OR_PointCloud
eventOccursAt: OperatingRoom

CloudsRegistration

algorithmActedOn: OR_PointCloud
algorithmActedOn: CT_PointCloud
outputsCreated: Pose
eventOccursAt: OperatingRoom
previousEvent: obtain3DfemurModelFromCT

RobotMotion

outputsCreated: Pose
previousEvent: DrillingPoseCalculus
algorithmActedOn: FemurPose
deviceUsed: CORA:SemiAutonomousRobot
eventOccursAt: OperatingRoom

FIGURE 3.9 Example of a surgical procedure definition, relating CORA, OROSU and KNOWROB framework.

defined and used no matter what type of surgical robot (defined using CORA) is applied in different scenarios.

3.5.4 Other Applications

The ontologies in IEEE 1872-2015 have been used or cited in other contexts as well. For example, Banerjee et al. [BBK+15], have used CORAX and other ontologies as part of a stochastic integer linear program for policies for optimum high-level task partitioning between collaborative teams of human operators and mobile ground robots. They successfully demonstrated collaboration of one human–robot pair leading to shorter completion times of a kitting operation.

Using ontological representations for grounding physical layers of the various HMI technologies, Pirvu et al. [PZG16] attempted to engineer a synthetic hybrid system called ACPS (using SOA), a human centric CPS which integrates physical and human components.

In a paper on cloud robotics, [KPAG15] notes that research in robot ontologies (KnowRob and CORA) can help to address current challenges such as heterogeneity, by defining cross-platform data formats and in future helping to standardize representation of elementary concepts such as trajectories.

In his paper, Diaconescu [DW14] remarked that the Core Ontology for Web of Things (WoTCO) ontology proposed by him has many similarities at the top-level with CORA. We observe that the differences are due to CORA using SUMO and WoTCO using Unified Foundational Ontology (UFO). While detailed concepts in CORA is subsumed under R-Parts and is general, WoTCo claims to have more details of concepts specific to the Web of Things. This similarity at the top level is comforting and can be viewed as reinforcing coherence of ontological concepts at the top level.

In his paper using elements defined in CORAX, Kootbally developed a platform-independent, Industrial Robot Capability Model [Koo16] and a planning infrastructure for planning and execution of industrial robot tasks for kitting assembly. Using CORA to represent a kitting workstation, its environment and objects, Kootbally proposed a robot capability model that defined additional components and pointers to CORA and action ontology. To represent robot capabilities for manufacturing applications, using XML Schema Definition Language (XSDL), models for robot parts like end-effectors, sensors and constraints have been defined. Assembly Action Model extends the dataThing class in CORA and the Robot Capability Model outputs a process plan with the best available robot to accomplish a kitting assembly task.

3.6 FOLLOW-ON EFFORTS

3.6.1 The Autonomous Robotics Study Group

The IEEE Ontologies for Robotics and Automation Working Group (ORA) was divided into sub-groups that are in charge of studying and developing industrial robotics, service robotics and autonomous robotics ontologies. The Autonomous Robotics (AuR) Study Group has over 80 active group members from North America, South America, Europe, Asia and Africa, representing stakeholders in academia, industry and governments. The AuR Study Group's goal is to extend the core ontology for robotics and automation to represent more specific concepts and axioms commonly used in autonomous robotics. The ontology should allow one to specify the knowledge and reasoning needed to build autonomous systems that can operate in the air, on the ground and underwater.

The AuR Study Group studies various R&A domains to identify basic components including the hardware and software necessary to endow robots with autonomy. In order to develop the standard ontology for autonomous systems, the AuR Study Group has adopted the following approach:

- Development of standard vocabularies for architectural concepts in IEEE 1471/IEC 42010.
- Development of a functional ontology for robotics and automation.
- Validations of developed relationships and concepts.
- Use of developed vocabularies and ontology for conceptual design of sample robot application using extended 1471 concepts and extension of CORAX.

For vocabulary development, the AuR Study Group has defined: “behaviour”, “function”, “goal” and “task”. As part of the architectural concepts development, the AuR Study Group has proposed the Robot Architecture (ROA) Ontology which defines the main concepts and relations regarding robot architecture in the context of autonomous systems. Its goal is to serve as a conceptual framework so that people and robots can share information about robot architectures.

A standard ontology in autonomous robotics will provide the underlying semantics of the vocabulary used in problem-solving and communications for heterogeneous autonomous systems. The proposed standard ontology will make possible a smooth integration among these systems in order to exchange knowledge and cooperatively perform tasks. Creating a standard for knowledge representation and reasoning in autonomous robotics will have a significant impact on all robotics and automation domains, including knowledge transmission among autonomous robots and humans. As a result, the use of autonomous robots by humans will further benefit our society. To the best of the authors’ knowledge, the AuR Study Group is the first group to adopt a systematic approach in developing ontologies, consisting of specific concepts and axioms that are commonly used in autonomous robots.

3.6.2 Robot Task Representation Effort

Under the original ORA working group, a sub-group was formed to examine an application ontology focused on industrial automation. This group found that with the growing demand for product variations, limited skilled manpower, increased complexity of working environments and greater requirements for robot–robot and human–robot collaboration, there is a desire to seamlessly integrate processes and systems at the task level. In this context, *task* refers to the concrete decomposition from goal to sub-goals that enables the human/robot to accomplish outcomes at a specific instance in time. In order to accomplish

AU: “Parentheses (background, status, Stephen & Craig & NEED TO ADD Hiren Nakawala) deleted from heading.

this, there is a need for a standard providing an explicit knowledge representation for robot tasks. Unfortunately, it was also found that no internationally recognized standard exists for describing tasks and how to integrate them. Therefore, this sub-group decided to refocus its attention on a task ontology that will represent the knowledge necessary to fully describe autonomous tasking.

Actors performing autonomous tasks need to have sufficient knowledge of their tasks to not only perform them, but to also communicate their pending activities to others, and to recognize and correct errors without the need to interrupt the process. The tasks can be either informational (i.e. storing, representing or transferring information between the actors) or physical, where the actors actually manipulate materials (e.g. the robot picks and places an object) [JJ94]. The task could also be "collaborative" (e.g. human–robot manipulation in materials handling), where autonomous robots are expected to collaborate with other robots, as well as humans.

The availability of such a standard knowledge representation will:

- define the domain concepts as controlled vocabularies for robot task representation
- ensure a common understanding between different industrial groups and devices
- facilitate interoperability between robotic systems for efficient data transfer and information exchange
- increase manufacturing performance, i.e. flexibility (easier incorporation of new processes due to a common set of concepts)

The industrial sub-group of ORA morphed into the *IEEE Robot Task Representation* (RTR) Study Group, with the objective of developing a broad standard that provides a comprehensive ontology for robot task structure and reasoning across robotics domains. This work will be a supplement to the existing standard Core Ontology for Robotics and Automation (CORA) [IEE15]. This supplement will include the presentation of concepts in an initial set of application domains (e.g., in manufacturing) where robot task representation could be useful. The ontology provides a unified way to represent knowledge about robot tasks by sharing common knowledge and preserving semantic meaning. It can be utilized in manufacturing control applications, where the system needs to control multiple elements of the manufacturing process.

Our work plan for developing the standard has two aspects. The first is to develop the task ontology, extending CORA and capturing vocabularies for robot task representation by requirements analysis and surveying the literature. The final decision making on vocabularies will be achieved through consensus between different group members. The ontology will contain vocabularies for generic tasks and specialized tasks for industrial application. The second aspect is to develop a task repository, which will provide a set of instances that could be used for robotic implementation and validation of the Task Ontology.

Although the Task Ontology will be the official standard when completed, the Task Repository is necessary to help validate the standard and to provide an avenue that makes the standard more useful and applied in the industry. The Task Ontology formally defines what a task is, and specifies the properties of tasks, the properties of the hierarchy in which tasks are placed and the ways in which the performance of the capabilities required to accomplish the tasks are measured. The Task Repository enables the community to build up a shared catalogue of tasks and capabilities along with their relationships (based on elements within the Task Ontology). The purpose of the overall standard is to ensure common representations and frameworks when tasks are described, so the knowledge represented in the Task Ontology defines the structure and content of the tasks in the Task Repository.

The existence of such a repository allows for clear definitions and descriptions of tasks and the ability for a user to quickly and easily determine if a task description (and associated algorithm) exists that will accomplish their goal, even if that task may have been created for a different purpose in another domain (Figure 3.10).

To develop this ontology, we will perform the following steps:

- **Requirements Gathering**: It is important to gather the relevant information requirements from all of the target domains to ensure that the resulting knowledge representation is truly comprehensive. RTR will reach out to experts in various robotics fields, some of whom are part of the group already and to others who are outside the group, to gain a deep understanding of what is necessary to represent task information in their domains. In addition to compiling the terms and definitions, RTR will start the process of identifying cases in which the same term has different meanings in different domains, as well as the cases where the same definition is associated with different terms. The output of this process will be a glossary of terms and definition, sorted by robot domain, which will serve as the basis for subsequent steps in the work plan.
- **Surveying Similar Efforts**: There are a number of efforts that have attempted to capture aspects of task information in specific robotic domains. In this phase of the work plan, RTR will deeply analyze these efforts to see how well they capture the concepts identified in the step above and evaluate their potential use as sources of definitions for the main concepts identified. In addition, RTR will look at the way that these concepts are represented (i.e. the knowledge representation formalism that is used and the attributes and relationships that are represented) to leverage the existing representational approaches and concepts wherever possible.

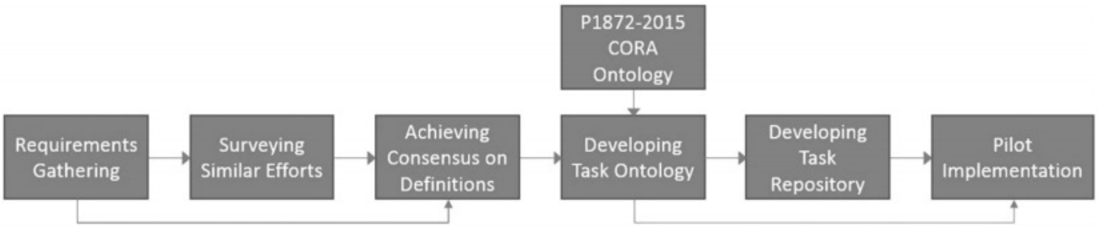


FIGURE 3.10 Robot Task Representation Study Group Work Plan.

- **Achieving Consensus on Definitions of Terms**: Coming to a consensus on terms and definitions is the most challenging, yet it is an essential part of the work. We have formed the study group so that there is representation from a wide array of robot domains. We are also closely collaborating with the autonomous robotics sub-area, which is also extending the P1872-2015 standard.

- **Developing the Task Ontology**: Building on the previous work of P1872-2015 (CORA), and using consensus definitions, RTR we will extend the CORA ontology to capture terms and definitions pertaining to robot tasks. While CORA does not specifically address robot tasks, it does define high level concepts of robot motion that can be leveraged. In addition, SUMO defines the concept of process, which can also be built upon as appropriate. RTR will use aspects of the METHONTOLOGY ontology development methodology [FLGPJ97], which includes the development of the ontology conceptual model; the formalization of the ontology, which involves transforming the conceptual model into a formal or semi-computable model (specified in first-order logic, for example); and the implementation of the ontology, which involves transforming the previously formalized ontology into a computable model, codified in an ontology representation language, such as OWL.

- **Modelling of Shared Tasks and Capabilities in a Task Repository**: While the Task Ontology provides the structure and definitions of concepts, the Task Repository provides task instances that can be applied towards robot applications. It uses the structure of the concepts in the Task Ontology and populates the values with instances specific to individual robot types and applications. To validate the Task Ontology, RTR will create a set of task instances in the Task Repository, focusing on tasks that are generally applicable across robot domains, such as pick and place tasks or robot mobility tasks. The hope is that as the Task Ontology and associated Task Repository get used more and more, the community will populate the Task Repository with additional task instances.

- **Pilot Implementation**: Using the task instances created in the previous step, we will create two control systems for robots in different domains. These control systems will use the knowledge represented in the Task Repository to control a robot performing operations in the domain of interest. We expect one of the domains to be in the manufacturing field due to the interest and availability of these types of robots among the study group members. The second domain is still to be determined. As mentioned above, we expect the tasks to focus on pick and place and mobility operations. This process will help to validate the knowledge represented in the Task Repository as well as the structure of the Task Ontology.

3.7 CONCLUSION AND DISCUSSION: TOWARDS AN ONTOLOGY FOR HRI

In this chapter, we showed an overview of the development and content of the IEEE 1872-2015 standard and the current efforts to expand it. The main contribution of IEEE 1872-2015 is to formalize core concepts of R&A in a set of ontologies. These ontologies, and in particular CORA, are intended to be used by other researchers and practitioners as a basis not only for the development of other ontologies in R&A, but also for system and database design. We also showcased some current applications of IEEE 1872-2015 and summarized the current standardization efforts in expanding the standard to robot task representation and autonomous robots.

While IEEE 1872-2015 covers some concepts related to HRI, it still needs to be extended in order to cover the domain to an adequate level. Designing partial or completely automated mechanisms for handling human–robot interaction requires a generic and consistent model for representing, sharing and reasoning about the semantics of the perceptions, actions and interactions of both humans and robots. This model must be sufficiently expressive and abstract to cover all the complexity and heterogeneity of the technologies and systems used to capture the interactions. CORA and the remaining ontologies in IEEE 1872-2015 provide some of these notions, but lack in enough specificity to be applied directly. A CORA-based ontology for HRI could help solve this issue. A basic requirement for an HRI ontology is that it should allow for consistent description of the spatial and temporal context of robot and user, integrating robot perception and human interpretation of the interaction situation, as well as instructions and queries. The HRI ontology should allow one to model the semantics of static entities, such as the description of the space and its constituents, but also the dynamic entities that describe the events, situations, circumstances and changes of the interactions. The ontology should be placed at the middle level of Figure 3.2, beside the task ontology, serving as a basis for the implementation of application-specific models in HRI systems.

HRI ontology requires a suitable representation language that could allow for models to be easily translated into formal models for AI planning or to the semantic web. The language must allow for the formal description of interactions between humans, robots and the other objects of the real world according to a human-like narrative approach. That means that the language should allow for the definition of conceptual n-ary relations between static and dynamic entities. For example, representing an interaction such as “the-man-gives-the-cup-to-the-robot” requires at least a quaternary predicate relating the man, the cup, the robot and the act of giving. The HRI ontology must allow one to represent complex dynamic entities according to a narrative approach inspired by the human way of reporting stories and events. To allow for that, HRI ontology must provide conceptual constructs corresponding to *verbs* describing the interactions of a robotic domain such as moving, sensing, grasping, behaving, changing state and so on. The NKRL language [ACA13, Zar13] could be a good candidate to specify dynamic and static entities of the CORA HRI ontology

The problem of connecting perceptions with the corresponding concepts describing static and dynamic entities in the HRI ontology requires the modelling of physical symbol grounding procedures. For example, from the perspective of an embodied robot that interacts with humans and objects in the environment, the grounding of its symbolic system needs to consider the complexity of its sensorimotor readings. In the context of multi-robot and Internet of Things environments, the problem becomes more complex as symbols are not only grounded, but commonly shared. Most of the state-of-the-art attempts have dealt with the grounding of static entities and new grounding techniques are needed for grounding dynamic entities. Therefore, the HRI ontology must propose a grounding methodology that will deal with the grounding of perceptions and situated verbal and non-verbal communication or dialogues between humans and robots. Symbolic grounding is called perceptual anchoring when specifically referring to the building of links between the robot's low-level processes (i.e. odometers, camera, radar, ultrasound) and the high-level semantic representations. Due to the complexity of grounding, it should be better to limit the perceptual anchoring to the concepts and rules that systematically creates and maintains in time the correspondences between symbols and sensor and actuation data that refers to the same physical object and the same human.

The driving force for using HRI ontology should be the use of constraint-based techniques to model and reason about the dependencies between the symbolic information returned by the perception systems, temporal objects and the dynamic entities describing interactions between humans and robots. For that, the ontology should include constructs to represent time relations, such as Allen's Interval Algebra. Uncertainty must be taken into account at the ontology level in order to allow a sound and tractable coupling of dynamic temporal objects and dynamic entities and avoid ambiguous reasoning. Thus, HRI ontology must include uncertainty measures with logical statements to describe simultaneous and nested dynamic entities. Considering the heterogeneity, redundancy and conflicts between perception and actuation systems, the use of belief theory at the symbolic level, for instance, would help to assign confidence values to assertions about dynamic entities and rules processed by reasoning engines.

ACKNOWLEDGMENT

T. Haidegger's research is supported by the Hungarian State and the European Union under the EFOP-3.6.1-16-2016-00010 project. T. Haidegger is a Bolyai Fellow of the Hungarian Academy of Sciences, and he is supported through the New National Excellence Program of the Ministry of Human Capacities.

REFERENCES

AU: Please give a location for "Ayari et al. (2013)".

[ACA13] N. Ayari, A. Chibani and Y. Amirat. Semantic management of human-robot interaction in ambient intelligence environments using n-ary ontologies. In *2013 IEEE International Conference on Robotics and Automation*, pp. 1172–1179, Institute of Electrical and Electronics Engineers (IEEE), May 2013.

[BBK+15] A.G. Banerjee, A. Barnes, K.N. Kaipa, J. Liu, S. Shriyam, N. Shah and S.K. Gupta. An ontology to enable optimized task partitioning in human-robot collaboration for warehouse kitting operations, volume 9494, SPIE, 2015.

[CDYM+07] A.F. Cutting-Decelle, R.I.M. Young, J.J. Michel, R. Grangel, J. Le Cardinal, and J.P. Bourey. ISO 15531 MANDATE: A product-process-resource based approach for managing modularity in production management. *Concurrent Engineering*, 15(2): 217–235, 2007.

[CFP+13] Joel Carbonera, Sandro Fiorini, Edson Prestes, Vitor A.M. Jorge, Mara Abel, Raj Madhavan, Angela Locoro, P.J.S. Gonçalves, Tamás Haidegger, Marcos E. Barreto, and Craig Schlenoff. Defining positioning in a core ontology for robotics. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1867–1872, Tokyo, Japan, November 2013.

AU: Please provide complete details for reference "Banerjee et al. (2015)".

AU: Please check authors for ref 5, York Sure is listed twice.

[DSV+05] John Davies, York Sure, Denny Vrandecic, Sofia Pinto, Christoph Tempich, and York Sure. The diligent knowledge processes. *Journal of Knowledge Management*, 9(5): 85–96, 2005.

[DW14] I.M. Diaconescu and G. Wagner. Towards a general framework for modeling, simulating and building sensor/actuator systems and robots for the web of things, volume 1319, pp. 30–41, CEUR-WS, 2014.

[FCG+15] Sandro Rama Fiorini, Joel Luis Carbonera, Paulo Gonçalves, Vitor AM Jorge, Vítor Fortes Rey, Tamás Haidegger, Mara Abel, Signe A. Redfield, Stephen Balakirsky, Veera Ragavan, et al. Extensions to the core ontology for robotics and automation. *Robotics and Computer-Integrated Manufacturing*, 33: 3–11, 2015.

[FLGPJ97] Mariano Fernández-López, Asunción Gómez-Pérez and Natalia Juristo. Methontology: From ontological art towards ontological engineering. 1997.

[Gon13] Paulo J.S. Gonçalves. Towards an ontology for orthopaedic surgery, application to hip resurfacing. In *Proceedings of the Hamlyn Symposium on Medical Robotics*, pp. 61–62, London, UK, June 2013.

[GT14] Paulo J.S. Gonçalves and Pedro M.B. Torres. A survey on biomedical knowledge representation for robotic orthopaedic surgery. *Robot Intelligence Technology and Applications*, 2: 259–268, 2014.

[GT15] Paulo J.S. Gonçalves and Pedro M.B. Torres. Knowledge representation applied to robotic orthopedic surgery. *Robotics and Computer-Integrated Manufacturing*, 33: 90–99, 2015.

[Gua98] Nicola Guarino. Formal ontology and information systems. In *Proceedings of FOIS*, volume 98, pp. 81–97, 1998.

[Gui05] Giancarlo Guizzardi. *Ontological Foundations for Structural Conceptual Models*. CTIT, Centre for Telematics and Information Technology, 2005.

[HMA03] Hui-Min Huang, Elena Messina and James Albus. Toward a generic model for autonomy levels for unmanned systems (ALFUS). In *Proceedings of Performance Metrics for Intelligent Systems 2003*, 2003.

[HMN+11] Mathias Haage, Jacek Malec, Anders Nilsson, Klas Nilsson, and Slawomir Nowaczyk. Declarative knowledge-based reconfiguration of automation systems using a blackboard architecture. In *Proceedings of the 11th Scandinavian Conference on Artificial Intelligence*, pp. 163–172, SCAI, 2011.

[IEE15] IEEE. IEEE Standard Ontologies for Robotics and Automation. *IEEE Std. 1872-2015*, 2015.

[JJ94] S. Joshi and S. Jeffrey. *Computer Control of Flexible Manufacturing Systems – Research and Development Edition*. Dordrecht, Springer Netherlands, 1994.

[JMN16] Ludwig Jacobsson, Jacek Malec and Klas Nilsson. Modularization of skill ontologies for industrial robots. In *Proceedings of International Symposium on Robotics*, Munich, Germany, June 2016.

[JRM+15] Vitor A.M. Jorge, Vitor F. Rey, Renan Maffei, Sandro Rama Fiorini, Joel Luis Carbonera, Flora Branchi, João P. Meireles, Guilherme S. Franco, Flávia Farina, Tatiana S. da Silva, Mariana Kolberg, Mara Abel and Edson Prestes. Exploring the {IEEE} ontology for robotics and automation for heterogeneous agent interaction. *Robotics and Computer-Integrated Manufacturing*, 33: 12–20, 2015.

[KJW+15] Darko Katić, Chantal Julliard, Anna-Laura Wekerle, Hannes Kenngott, Beat-Peter Müller-Stich, Rüdiger Dillmann, Stefanie Speidel, Pierre Jannin and Bernard Gibaud. Lapontospm: An ontology for laparoscopic surgeries and its application to surgical phase recognition. *International Journal of Computer Assisted Radiology and Surgery*, pp. 1–8, 2015.

[Koo16] Z. Kootbally. Industrial robot capability models for agile manufacturing. *Industrial Robot*, 43(5): 481–494, 2016.

[KPAG15] B. Kehoe, S. Patil, P. Abbeel and K. Goldberg. A survey of research on cloud robotics and automation. *IEEE Transactions on Automation Science and Engineering*, 12(2): 398–409, 2015.

[LAP+12] S. Lemaignan, R. Alami, A.K. Pandey, M. Warnier and J. Guitton. Bridges between the methodological and practical work of the robotics and cognitive systems communities-from sensors to concepts, chapter towards grounding human-robot interaction. *Intelligent Systems Reference Library*, 2012.

AU: Please give the publisher and location for reference 6.

AU: Please provide complete details for reference "Mariano Fernández-López et al. (1997)".

AU: Please provide location for references "Guarino (1998)", "Guizzardi (2005)" and "Haage (2011)".

AU: Please provide publisher details for reference "Hui-Min Huang et al. (2003)".

AU: Please provide volume number for reference "Darko Katic et al. (2015)".

AU: Please give the location for reference 23.

[Mad15] Raj Madhavan. The first-ever IEEE RAS standard published. *IEEE Robotics & Automation Magazine*, 22(2): 21, 2015.

[MJD+15] Andrej Machno, Pierre Jannin, Olivier Dameron, Werner Korb, Gerik Scheuermann and Jürgen Meixensberger. Ontology for assessment studies of human–computer-interaction in surgery. *Artificial Intelligence in Medicine*, 63(2): 73–84, 2015.

[NJS+09] T. Neumuth, P. Jannin, G. Strauß, J. Meixensberger, and O. Burgert. Validation of knowledge acquisition for surgical process models. *Journal of the American Medical Informatics Association*, 16: 72–80, 2009.

[NP01] Ian Niles and Adam Pease. Towards a standard upper ontology. In *Proceedings of the International Conference on Formal Ontology in Information Systems – Volume 2001*, FOIS'01, pp. 2–9, New York, NY, USA, ACM, 2001.

AU: Please list 10 authors before et al.

[NSW+09] N.F. Noy, N.H. Shah, P.L. Whetzel, B. Dai, et al. Bioportal: Ontologies and integrated data resources at the click of a mouse. *Nucleic Acids Research*, 37(s2): 170–173, 2009.

[OAH+07] Daniel Oberle, Anupriya Ankolekar, Pascal Hitzler, Philipp Cimiano, Michael Sintek, Malte Kiesel, Babak Mougouie, Stephan Baumann, Shankar Vembu, Massimo Romanelli, et al. DOLCE ergo SUMO: On foundational and domain models in the SmartWeb integrated ontology (SWIntO). *Web Semantics: Science, Services and Agents on the World Wide Web*, 5(3): 156–174, September 2007.

[Obr10] Leo Obrst. Ontological architectures. In *Theory and Applications of Ontology: Computer Applications*, pp. 27–66. Dordrecht, Springer, 2010.

AU: Please provide editor name and publisher location for reference "Leo Obrst (2010)".

[PCF+13] Edson Prestes, Joel Luis Carbonera, Sandro Rama Fiorini, Vitor A.M. Jorge, Mara Abel, Raj Madhavan, Angela Locoro, Paulo Goncalves, Marcos E. Barreto, Maki Habib, et al. Towards a core ontology for robotics and automation. *Robotics and Autonomous Systems*, 61(11): 1193–1204, 2013.

[PNDM+14] R. Perrone, F. Nessi, E. De Momi, F. Boriero, M. Capiluppi, P. Fiorini and G. Ferrigno. Ontology-based modular architecture for surgical autonomous robots. In *The Hamlyn Symposium on Medical Robotics*, p. 85, 2014.

AU: Please provide editor name and location for reference "Perrone et al. (2014)".

AU: Please give further details for '[pro14]'.

[pro14] NeOn project. Ontology design patterns (ODP), 2014.

[PZG16] B.-C.a Pirvu, C.-B.a b Zamfirescu and D.a Gorecky. Engineering insights from an anthropocentric cyber-physical system: A case study for an assembly station. *Mechatronics*, 34: 147–159, 2016.

AU: Please list 10 authors before et al.

[SAR+07] B. Smith, M. Ashburner, C. Rosse, J. Bard, W. Bug, W. Ceusters, et al. The OBO Foundry: Coordinated evolution of ontologies to support biomedical data integration. *National Biotechnology*, 37: 1251–1255, 2007.

[SBF98] Rudi Studer, V. Richard Benjamins and Dieter Fensel. Knowledge engineering: Principles and methods. *Data & Knowledge Engineering*, 25(1): 161–197, 1998.

[SFGPFL12] Mari Carmen Suárez-Figueroa, Asuncion Gomez-Perez and Mariano Fernandez-Lopez. The neon methodology for ontology engineering. In *Ontology Engineering in a Networked World*, pp. 9–34, Springer, 2012.

AU: Please provide editor name and location for reference 37.

[SM15] Maj Stenmark and Jacek Malec. Knowledge-based instruction of manipulation tasks for industrial robotics. *Robotics and Computer-Integrated Manufacturing*, 33: 56–67, 2015.

[SSS04] York Sure, Steffen Staab, and Rudi Studer. On-to-knowledge methodology (OTKM). In *Handbook on Ontologies*, pp. 117–132, Springer, 2004.

AU: Please provide editor name and location for reference "York Sure et al. (2004)".

[SSS09] York Sure, Steffen Staab and Rudi Studer. Ontology engineering methodology. In *Handbook on Ontologies*, pp. 135–152, Springer, 2009.

AU: Please provide editor name and location for reference "York Sure et al. (2009)" and "Guus Schreiber et al. (1995)".

[SWJ95] Guus Schreiber, Bob Wielinga and Wouter Jansweijer. The kactus view on the 'o' word. In *IJCAI Workshop on Basic Ontological Issues in Knowledge Sharing*, pp. 159–168, Citeseer, 1995.

[TB13] Moritz Tenorth and Michael Beetz. Knowrob: A knowledge processing infrastructure for cognition-enabled robots. *The International Journal of Robotics Research*, 32(5): 566–590, 2013.

[Zar13] Gian Piero Zarri. Advanced computational reasoning based on the {NKRL} conceptual model. *Expert Systems with Applications*, 40(8): 2872–2888, 2013.