Software and Systems Modeling
https://doi.org/10.1007/s10270-023-01115-3

SPECIAL SECTION PAPER

An ontology-based approach to engineering ethicality requirements

Renata Guizzardi[1] · Glenda Amaral[1] · Giancarlo Guizzardi[1] · John Mylopoulos[2]

Received: 16 November 2022 / Revised: 1 June 2023 / Accepted: 12 June 2023

Abstract

In a world where Artificial Intelligence (AI) is pervasive, humans may feel threatened or at risk by giving up control to machines. In this context, ethicality becomes a major concern to prevent AI systems from being biased, making mistakes, or going rogue. Requirements Engineering (RE) is the research area that can exert a great impact in the development of ethical systems by design. However, proposing concepts, tools and techniques that support the incorporation of ethicality into the software development processes as explicit requirements remains a great challenge in the RE field. In this paper, we rely on Ontology-based Requirements Engineering (ObRE) as a method to elicit and analyze ethicality requirements ('Ethicality requirements' is adopted as a name for the class of requirements studied in this paper by analogy to other quality requirements studied in software engineering, such as usability, reliability, and portability, etc. The use of this term (as opposed to 'ethical requirements') highlights that they represent requirements for ethical systems, analogous to how 'trustworthiness requirements' represent requirements for trustworthy systems. To put simply: the predicates 'ethical' or 'trustworthy' are not meant to be predicated over the requirements themselves). ObRE applies ontological analysis to ontologically unpack terms and notions that are referred to in requirements elicitation. Moreover, this method instantiates the adopted ontology and uses it to guide the requirements analysis activity. In a previous paper, we presented a solution concerning two ethical principles, namely Beneficence and Non-maleficence. The present paper extends the previous work by targeting two other important ethicality principles, those of Explicability and Autonomy. For each of these new principles, we do ontological unpacking of the relevant concepts, and we present requirements elicitation and analysis guidelines, as well as examples in the context of a driverless car case. Furthermore, we validate our approach by analysing the requirements elicitation made for the driverless car case in contrast with a similar case, and by assessing our method's coverage w.r.t European Union guidelines for Trustworthy AI.

Keywords Requirements elicitation and analysis · Ontological analysis · Foundational ontologies · Ethicality requirements · Ethical systems

1 Introduction

In a world where Artificial Intelligence (AI) is pervasive, controlling more services and systems everyday, humans may feel threatened or at risk by giving up control to machines. In this context, many of the potential issues are related to safety and ethics. For example, AI systems may be biased towards one group of people in detriment of others, resulting in job loss and wealth inequality; they may also make mistakes and even go rogue, by acting against the interests of stakeholders [54].

According to the Markkula Center for Applied Ethics, ethics refers to standards of behavior that tell us 'how we ought to act' while playing different roles, e.g. worker, driver, parent, citizen, engineer, medical doctor, etc. For

Communicated by Iris Reinhartz-Berger and Dominik Bork.

✉ Renata Guizzardi
r.guizzardi@utwente.nl

Glenda Amaral
g.c.mouraamaral@utwente.nl

Giancarlo Guizzardi
g.guizzardi@utwente.nl

John Mylopoulos
jm@cs.toronto.edu

1 University of Twente, Enschede, The Netherlands

2 University of Toronto, Toronto, Canada

Published online: 22 July 2023

each role, there are ethical codes of conduct that capture such standards of behaviour [1]. Ethicists and AI researchers have been studying the interplay of ethics and AI systems where the subjects of ethical codes are systems that play such roles, e.g., worker, driver. Floridi et al. [22] proposes five general principles that underlie ethical codes and are role-independent. These have been adopted by the European Commission in a document concerning trustworthy AI [41]. The principles are: *Autonomy* (respect human dignity), *Beneficence* (do good to others), *Non-maleficence* (do no harm to others), *Justice* (treat others fairly), and *Explicability* (explainability, transparently). But besides being recognized and well-defined, these principles need to be embedded into software engineering practices, in a way that they may effectively guide the design of ethical systems.

Requirements Engineering (RE) is the research area that can exert a great impact in the development of ethical systems by design. RE is not only responsible for eliciting requirements that will guide the design of a system-to-be, but also for ensuring that such requirements have been properly met, and also for monitoring that these requirements remain valid throughout the life cycle of the system. However, to the best of our knowledge, there is no current Requirements Engineering method to support the development of ethical systems.

The proposed solution to the problem-at-hand involves a deep understanding of what the proposed ethical principles mean and how they can be converted into concrete system requirements that can guide system design and runtime monitoring. For that, we rely on the Ontology-based Requirements Engineering (ObRE) method. ObRE consists of three activities: (1) Adopt or develop an ontology to conceptually clarify the meaning of requirements (in this paper, ethicality requirements); (2) Instantiate the ontology for a system-to-be, resulting in a domain model; and (3) Use the domain model to guide analysis, resulting in requirements models, such as goal models, requirements tables, user stories etc. ObRE has been previously proposed in [7], where it has also been illustrated for the elicitation and analysis of trustworthiness requirements.

In this paper, we use ObRE to propose a method for eliciting and analyzing ethicality requirements.[1] We can categorize ethicality requirements for a system-to-be as types of *Ecological Requirements* [40], in that they are derived from the ecosystem within which the system-to-be is embedded. After all, it is that ecosystem that determines *values* and *risks* that can lead to ethical behaviours by the system ([40], p. 253). In fact, as mentioned in Sect. 1, the focus on ethics is motivated by the emerging feeling of risk brought by the use of recent technologies. And these risks must be accounted for and analyzed in contrast with the values delivered by systems and services applying such technologies.

In a previous paper, we presented a solution concerning two of the ethical principles defined by Floridi et al. [22], namely Beneficence and Non-maleficence [36]. The present paper significantly extends the previous work by targeting two other important ethicality principles, those of Explicability and Autonomy. For each of these new principles, we do semantic unpacking of the relevant concepts, and we present requirements elicitation and analysis guidelines, as well as examples in the context of a driverless car case. Moreover, we present a validation of our approach, by verifying it against a checklist of goals established by the EU initiative towards the development of ethical systems by design [21]. We are able to show that the proposed method guides the requirements engineer in capturing requirements associated to most of the goals present in this checklist.

The remainder of this paper is structured as follows. Section 2 discusses the role of ontological analysis, and explains the existing ontologies reused in this work; Sect. 3 describes the ObRE method; Sect. 4 applies ontological analysis and instantiation for ethicality requirements; Sect. 5 presents requirements elicitation and analysis for a driverless car case; Sect. 6 validates the proposed ObRE-based method for eliciting and analyzing ethicality requirements. Section 7 discusses related works; and finally, Sect. 8 concludes and sketches future work.

2 Research baseline

2.1 Ontological analysis

The notions of *ontology* and *ontological analysis* adopted here are akin to their interpretations in philosophy [11]. In this view, *ontological analysis*: (i) characterizes what kinds of entities are assumed to exist in a given domain; (ii) offers a metaphysical account of the nature of these kinds of entities. An *ontology*, in turn, is a system of concepts and their relationships that result from (i) and (ii).

As such, an ontology is neither merely a logical specification nor it is mainly concerned with making terminological

[1] The ObRE method proposed here is agnostic w.r.t. to the AI approach that is employed to build the systems whose requirements are analyzed (e.g., whether based on symbolic or machine leaning approaches). It is also agnostic w.r.t. material domain (e.g., healthcare, penal recidivism) or application. Instead, it focuses on identifying and ontologically unpacking relevant ethical dimensions, as well as supporting the elicitation, explicit modeling and reasoning with requirements of any system that can directly impact (positively or negatively) human values and goals. In that sense, ObRE can be employed to address systems of all types that can potentially have this effect—the interest reader is referred to [44] for a collection of relevant examples. This particular paper focuses on AI systems given, on one hand, the scale of their potential effect on values and goals and, on the hand, given the existence of specific regulations and guidelines for AI systems that against which we could validate our proposal (see Sect. 6.2 of the paper).

and taxonomic distinctions. For example, in addressing the domains of risk, one is less concerned with what specific subtypes of risk exist (e.g., physical, biological, financial, electronic), but instead with what exactly *is* risk? What kind of entity is it? What is its nature? Is it an object? an event? a relationship? a complex property? If the latter, is a categorical or dispositional property? what is the bearer of such a property?, and so on.

Given the nature of this method of analysis, it must be supported by a domain-independent system comprising the most general categories, hence, crosscutting several domains (e.g., objects, events, relationships, dispositions, etc.), i.e., what is termed a foundational ontology (aka top-level or upper-level ontology). In this article, we adopt the Unified Foundational Ontology (UFO). Moreover, in order to develop and represent our models, we use the UFO-based ontology representation language OntoUML. OntoUML contains modeling primitives that represent the ontological distinctions put forth by UFO, and its grammar semantically-motivated syntactical constraints that reflect the axiomatization of UFO [28].

The choice of UFO and OntoUML are justified in a number of grounds:

- both UFO and OntoUML have a successful track record in supporting ontological analysis of complex related notions such as *value*, *risk*, *service*, *trust* and *trustworthiness*, *legal relations*, *money*, *decisions*, *economic preferences*, among many others [6, 34, 37, 49];
- empirical evidence shows that the use of OntoUML significantly contributes to improving the quality of domain representations without requiring an additional effort to produce them [58]. As demonstrated in [10], in contrast to traditional conceptual modeling languages, the process of ontological analysis supported by OntoUML is actually a process of explanation, which reveals the truthmakers of the propositional content present in the model. The resulting models are much more explicit regarding their ontological commitments when compared to traditional models, thus, facilitating domain comprehension and interoperability [29];
- as shown by [57], UFO is the second-most used foundational ontology in conceptual modeling and the one with the fastest adoption rate. The authors also show that OntoUML is among the most used ontology-driven conceptual modeling languages in the literature. The diffusion of UFO and OntoUML in the field facilitates the accessibility of our results;
- although there are a few alternative foundational ontologies (see [12]), UFO is the only one among these that is accompanied by a full-blown modeling language (OntoUML) with its tool ecosystem. The former enables the representation of our models in a conceptual modeling diagrammatic language, thus, facilitating their accessibility by requirements engineers, and the latter allows one to potentially leverage this tool ecosystem for model verification, validation, verbalization and code generation [25, 27]. In particular, it allows for the generation of logical specifications in OWL/SWRL for the models proposed here, thus, enabling automated reasoning over these models;

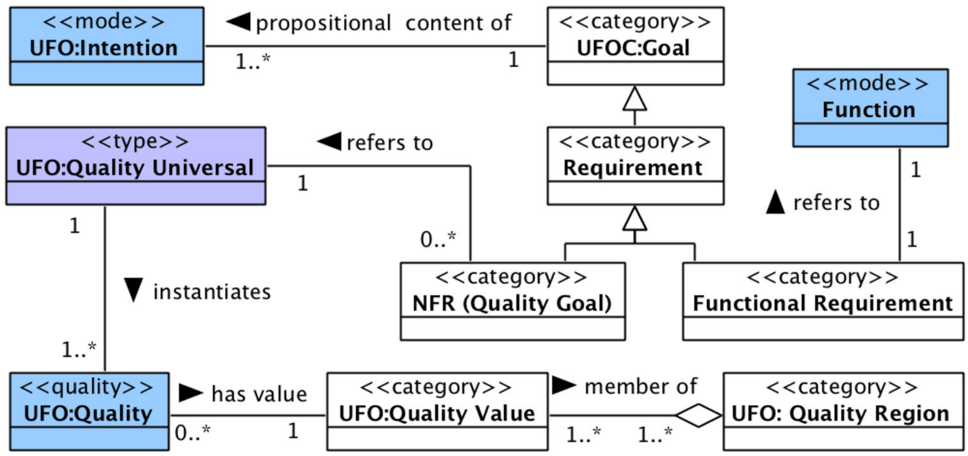
However, the main reason for adopting UFO (and OntoUML) here is the following: as it will become evident in the models that follow, an analysis of the different dimensions of this domain requires the ontological support of: a mature theory of relations and relational aspects (relational modes and relators) [24, 26]; a theory of events [13] and how they can be represented in structural conceptual models [4, 33]; and a theory of types [31] including higher-order types [23]. UFO is currently the only foundational ontology that satisfy all these theoretical requirements. To briefly contrast it with a few alternatives: DOLCE does not include a theory of types and it does not countenance relational aspects; BFO does not include a full-blown theory of material relations, events or types, and does not countenance the notion of higher-order types. Likewise, BWW rejects the very idea of higher-order types, as shown later in the paper, abound in this domain.

2.2 An ontology for requirements (NFRO)

The Non-Functional Requirements Ontology (NFRO) is defined as an extension of UFO. As such, it adopts the UFO notion of AGENT, an entity having mental states such as belief, desire and intention and means to act accordingly.[2] Also, the notion of INTENTION that refers to a situation (state-of-affairs) that the AGENT commits to bring about by pursuing goals and executing actions. It is also important to state that according to UFO, AGENT can be categorized into HUMAN (i.e. a person), ARTIFICIAL (i.e. artificial systems, such as information systems, cyber-physical systems, etc.) and INSTITUTIONAL (i.e. organization). A Stakeholder may be a HUMAN or an INSTITUTIONAL agent, while the system-to-be is an ARTIFICIAL one. Given the focus of this article, we do not include a figure showing this AGENT categorization, but we refer the reader to [35] (chap.3), for details.

[2] We make no commitment here to the so-called *Strong AI* view [50], in which certain artificial agents are assumed to be able to bear mental states and support cognitive processes exactly in the same way as their human counterparts. Instead, we adopt the approach put forth by the *Intentional Stance Theory*, in which artificial agents are assumed to have what is termed *derived* or *'as if'* intentionality. This stance assumes a strategy of interpreting the behavior of an agent as if it possessed certain beliefs and as if its behavior were directed by certain desires and goals. For a full defense of this approach for the analysis of computational systems and other complex artifacts, we refer to [16].

Fig. 1 A fragment of the Ontology of Non-functional Requirements



Requirements can be functional and non-functional, while the latter has special relevance to ethicality requirements, so we focus on them by adopting NFRO [39]. In NFRO, a requirement is a GOAL. Requirements are specialized into NFRs (aka QUALITY GOALS) and FUNCTIONAL REQUIREMENTS (FRs). FRs refer to a FUNCTION (a capability, capacity) that a system can manifest in particular SITUATIONS. NFRs refer to desired qualities taking QUALITY VALUES in particular QUALITY SPACES. For example, a software system is considered to have good usability if the value associated to its "usability" quality maps to the "good" quality region in the "usability" quality space. In other words, functional requirements prescribe *what* should be done by the system (i.e., what kind of behavior it should exhibit). In NFRO, this means that the system should be endowed with functions whose manifestations will satisfy certain goals. In contrast, NFRs refer to *how* these manifestations occur. For example, transporting a passenger from A to B is a functional requirement that can be satisfied by the manifestation of a complex transportation function, thus, satisfying a crisp goal. Now, transporting that passenger e.g., in a calm (as opposed to aggressive) driving style, as quick as possible, in an ecologically-friendly manner refer to *ways* in which that transportation event unfolds, i.e., a to particular *quality* of that event. Notice that this quality emerges from the interplay between other functions (dispositions) and qualities of the system and its environment. This point is further discussed in Sect. 2.4. So, although both functions and qualities play a role in designing systems that satisfy ethical requirements, the latter (as we elaborated in Sect. 2.1 are formulated in terms of qualities characterizing the actions brought about by the system.

UFO makes an important distinction among types of MOMENTS (existentially dependent entities). In particular, among types of INTRINSIC MOMENTS (those MOMENTS that are existentially dependent on a single individual). This is the distinction between QUALITIES and MODES. QUALITIES (e.g., color, height, weight, duration) represent objectified intrinsic properties of an individual. The types instantiated by these QUALITIES (i.e., QUALITY UNIVERSALS) are directly associated to a QUALITY SPACES [28]. A QUALITY is then constrained to vary its QUALITY VALUE within the points constituting that QUALITY SPACE. For example, the QUALITY UNIVERSALS *Color* is associated with a particular tri-dimensional QUALITY SPACE (the *Color Spindle*) and, hence, the color of a particular object (i.e., a QUALITY individual) can only vary its value within that space. In contrast with QUALITIES, MODES are existentially dependent entities that are not directly associated with QUALITY SPACES. Instead, MODES can have parts and can have their own QUALITIES and MODES, which can change in independent ways. A particular type of MODE is a DISPOSITION. DISPOSITIONS are MODES that manifest in certain SITUATIONS and always via the occurrence of EVENTS. Not only *Functions* but many of the key notions that appear later in this paper (e.g., capabilities, vulnerabilities, intentions, duties, powers, rights) are examples of DISPOSITIONS.

This ontological account delineates different kinds of requirements, and clarifies the nature of NFRs as qualities that map a system artifact into a quality region [39]. Figure 1[3] depicts a selected subset of the NFRO that is relevant here. For an in-depth discussion and formal characterization of QUALITIES, QUALITY UNIVERSALS, DISPOSITIONS, and QUALITY SPACES, we refer the reader to [28, 38].

2.3 The common ontology of value and risk (COVER)

The Common Ontology of Value and Risk (COVER) [49] breaks down VALUE EXPERIENCES into events, dubbed VALUE EVENTS. These are classified into IMPACT EVENTS and TRIGGER EVENTS. The former directly impact a goal

[3] In all OntoUML diagrams, we adopt the following color coding: types are represented in purple, objects in pink, modes in blue, events in yellow, and abstract entities such as numbers, sets and propositions in white.

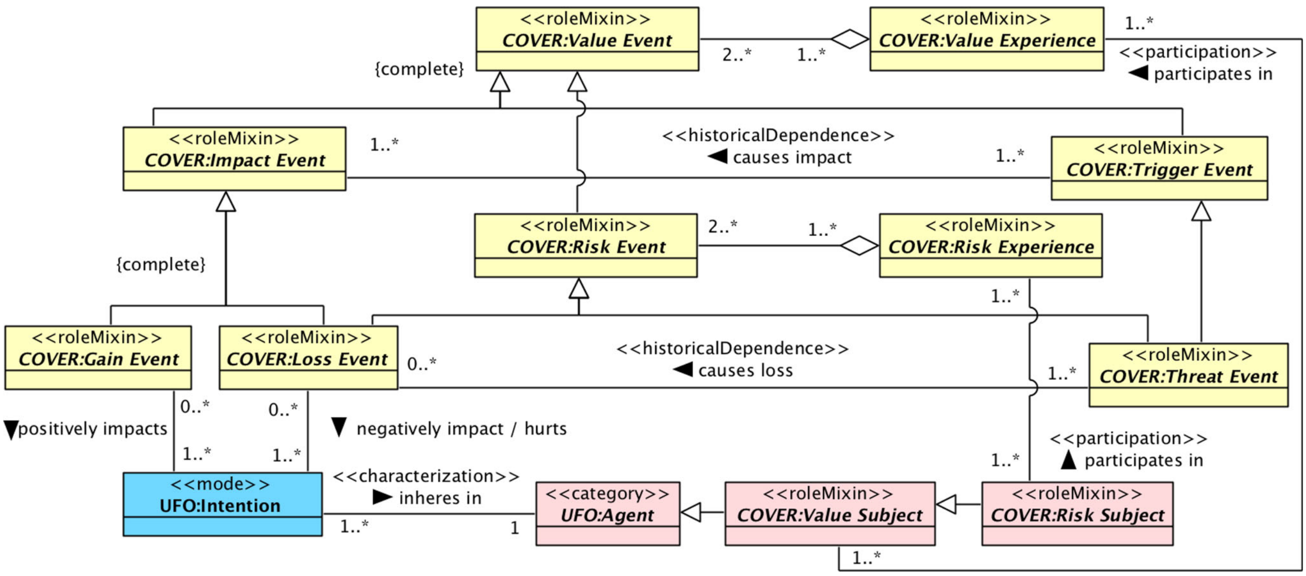


Fig. 2 A fragment of COVER depicting value and risk experiences [49]

or bring about a situation that impacts a goal. while TRIGGER EVENTS are simply parts of an experience identified as causing IMPACT EVENTS, directly or indirectly. Within the category of IMPACT EVENTS we can further distinguish into GAIN EVENT and LOSS EVENT. The difference between them rests on the nature of the impact on goals (positive for GAIN EVENTS and negative for LOSS EVENTS). To formalize goals, COVER reuses the concept of INTENTION from UFO [12].

RISK EXPERIENCES are unwanted events that have the potential of causing losses, and are composed by RISK EVENTS, which can be of two types, namely threat and loss events. A THREAT EVENT carries the potential of causing a loss, intended or unintended. A THREAT EVENT might be the manifestation of: (i) a Vulnerability (a special type of disposition whose manifestation constitutes a loss or can potentially cause a loss); or (ii) a Threatening Capability (capabilities of a threat object that, hence, can dent the goals a Risk Subject). The second mandatory component of a RISK EXPERIENCE is a LOSS EVENT, which necessarily impacts intentions in a negative way. Figure 2 depicts a fragment of COVER, which captures part of the aforementioned ontological notions.

2.4 The decision making ontology (DMOnto)

Figure 3 shows an excerpt extending the Decision Making Ontology (DMOnto) [37], which defines DECISION as a particular kind of INTENTION resulting from a DELIBERATION performed by the AGENT.

DMOnto also goes deeper into the DELIBERATION process, analyzing it in terms of the concepts of PREFERENCE and PROSPECT ASCRIPTION.[4] PROSPECT ASCRIPTION results from a process of assigning VALUE or RISK to PROSPECT BEARERS (either a PROSPECT OBJECT or a PROSPECT EXPERIENCE). For instance, when one decides to take a particular route when traveling, one considers the values of taking that route (e.g. it is the shortest route w.r.t. the destination) as well as the risks (e.g. the risk of getting caught in a traffic jam). The concepts of VALUE and RISK have been already defined by the COVER ontology in the previous section.

When an AGENT decides something (i.e., performs a DELIBERATION), she takes into consideration her own PREFERENCES regarding two possible PROSPECT BEARERS. A PREFERENCE is the truthmaker of the ternary HAS PREFERENCE relation, the latter connecting a PREFERRED BEARER and the non-preferred bearer (termed DEPRECATED BEARER). PREFERENCE is thus a complex mode, which aggregates two PROSPECT ASCRIPTIONS, each one associated to one of the PROSPECT BEARERS. According to [46], this binary case may be extrapolated to include other PROSPECT BEARERS, each one associated to its own value.

Each PROSPECT ASCRIPTION is composed of several smaller “comparisons” (or “judgements”), named PROSPECT ASCRIPTION COMPONENTS, which aggregate an INTENTION and INTRINSIC MOMENTS that are taken into consideration by the AGENT when ascribing VALUE or RISK to a PROSPECT BEARER. In the aforementioned route example, there are two

[4] The original DMOnto ontology reuses the preference ontology module as proposed by [46]. In that paper, preference is grounded on value ascriptions and value is assume to be possibly positive and negative. Analogously, COVER countenances both positive and negative risk [49]. Here we use the abstract type PROSPECT to generalize over positive (a value prospect) and negative value (i.e., a risk prospect).

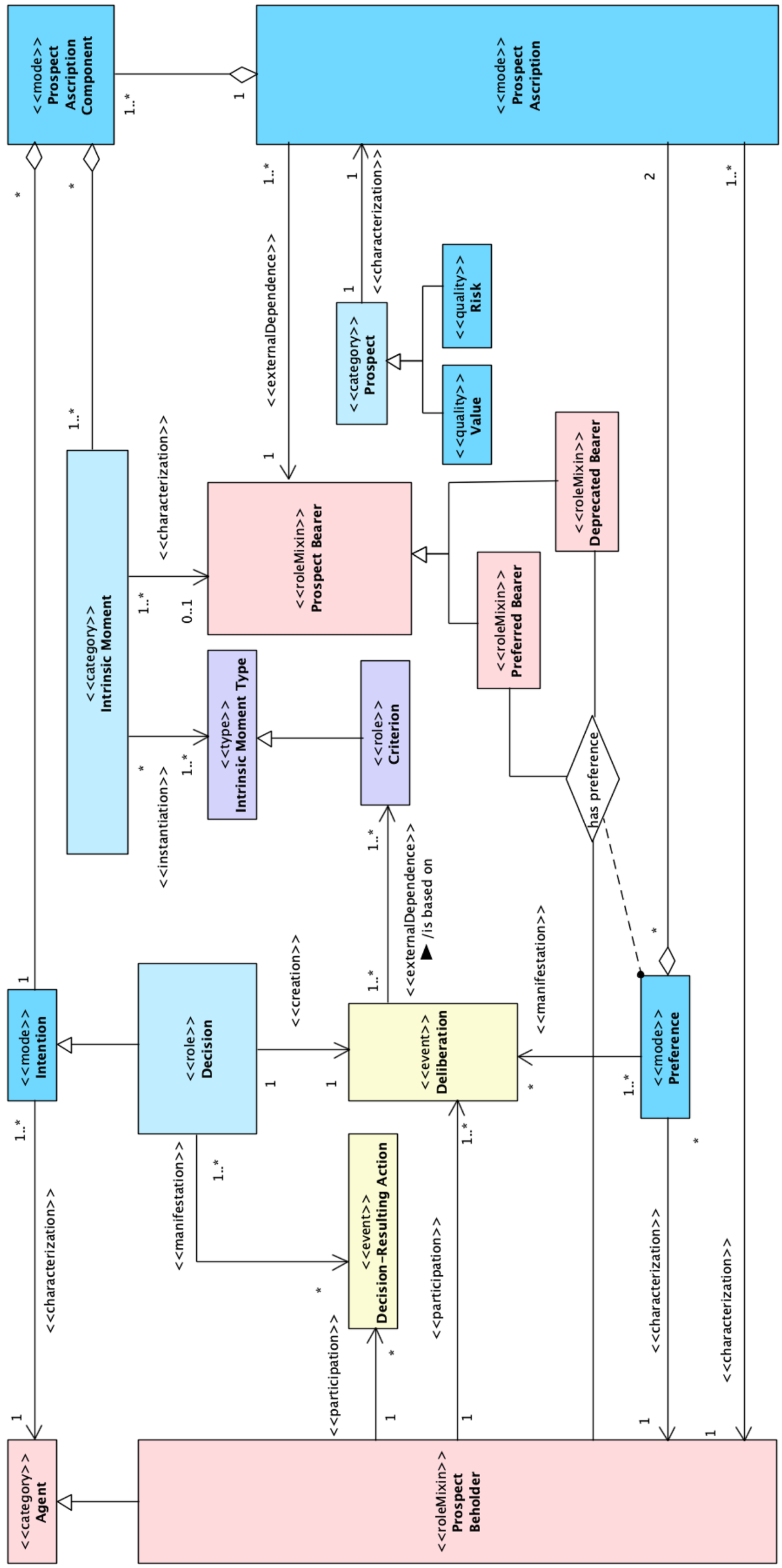


Fig. 3 A fragment of DMOnto

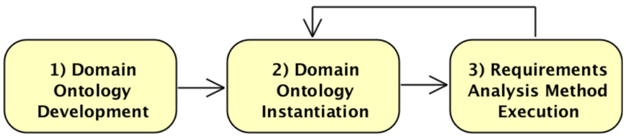


Fig. 4 The ObRE Process

PROSPECT ASCRIPTION COMPONENTS: one related to the QUALITY of the route of being the shortest one w.r.t. the destination and an INTENTION of "getting to the destination as fast as possible"; and another related to the QUALITY of how busy the route is and the same INTENTION of "getting to the destination as fast as possible".

In establishing preference, the deciding AGENT considers intrinsic moments (MODES and QUALITIES) associated with the PROSPECT BEARER (e.g., length and topology of the road), with entities in the environment (value and risk enablers in COVER) (e.g., average speed of a collective of cars on that road in that period), the capacities of the AGENT themselves (e.g., competence of the driver in driving in certain conditions), as well as (see discussion in 4.2) the social positions inhering in these AGENTS. The types instantiated by these INTRINSIC MOMENTS that ground PREFERENCE and, hence, DELIBERATION are termed decision CRITERIA. In other words a CRITERION may be a QUALITY TYPE, e.g. length (of the route), or a MODE TYPE, e.g., the existence of the functionality provided by an automatic gearbox, in case you are buying a car.

A DECISION is a type of INTENTION (and, hence, a type of disposition) created by a DELIBERATION event. A DECISION-RESULTING ACTION is an event that manifests that intention in particular situations.

3 The ObRE method

Figure 4 illustrates the process of the ObRE method, showing the three activities mentioned in Sect. 1.

The process starts with **(1) Domain Ontology Development**, requirements analysts and ontology engineers perform ontological analysis for a class of requirements. We emphasize that ObRE does not prescribe that the requirements engineer is versed in the use of ontological analysis concepts. For that, ObRE assumes the presence of an ontology engineer, and the requirements engineer plays a role of a domain expert in the ontology development process. The outcome of activity (1), is an ontology modelled in OntoUML. This activity is performed once for each class of requirements and doesn't need to be repeated for each new system development project. For example, in [5], we conducted ontological analysis about the notions of *trust* and *trustworthiness* in order to unpack the meaning of trustworthiness requirements. According to the results of our analysis, a system is trustworthy if it is believed to have the capability to perform its required functions (Capability belief) and its vulnerabilities will not prevent it from doing so (Vulnerability belief). Moreover, we define trustworthiness as a composition of three other qualities, namely *reliability* in performing its functions, *truthfulness* in presenting its credentials and *transparency* in its operations. To judge how reliable a system is, we must understand how much of the Stakeholder's Capability Belief is actually met by the system's operations. Note that reliability could have been defined in multiple other ways, for instance, it could have been related to accessibility, i.e., how often will the system be responsive to stakeholder needs; or inferred by the system possessing a specific reliability certificate. The trustworthiness ontology has been recently used in a real case study, reported in [6], showing promising results in defining and monitoring trustworthiness requirements for a particular system. In case a new trustworthy system needs to be developed, the same ontology can be fully reused, and instantiated for the new system-to-be.

Having the requirements explicitly defined and understood, the analyst may perform **(2) Domain Ontology Instantiation**. Here, the analysts focus on a particular system and instantiate elements of the ontology. For a security ontology, this step would identify stakeholders, vulnerabilities, attack types, etc. for a particular system. This is intended to serve as domain model for conducting requirements analysis. We highlight the importance of this step, since the same class of requirements may lead to distinct concrete requirements for each system. Thus, instantiating the ontology created in (1) helps identify these particular requirements and opens the way for the system-to-be requirements analysis.

In activity **(3) Requirements Analysis Method Execution**, analysts use the domain model to define and analyze system requirements. For instance, she may simply define a requirements table, listing the requirements instantiated with the help of the ontology. Or if she prefers a more sophisticated analysis methodology, she may use goal modeling, defining the contribution of different choices to accomplish a particular goal (i.e., requirement), and specifying how goals relate to each other, as well as to relevant stakeholders' resources and tasks. Or yet, she may create user stories based on the identified ontological instances. From this point on, the requirements analysis may progress as the chosen method prescribes, however, with the benefit of having the ontology and ontological instances as guides.

As depicted in Fig. 4, steps (2) and (3) are intended to be carried out iteratively, as with most RE methods. This supports the analyst in revisiting the previous activities while maturing the requirements elicitation and analysis.

Table 1 summarizes some practical guidelines for the ontology-based requirements engineering process.

Table 1 Ontology-based Requirements Engineering practical guidelines

1	Adopt or develop an ontology for conceptualizing a class of requirements
	In the case of new ontology development, apply *ontological analysis* to semantically unpack terms and notions related to a class of requirements
2	Instantiate the ontology for a system-to-be, resulting in a domain model
	2.1 Identify key concepts in the ontology
	2.2 Create questions to ask the stakeholders, based on the ontology key concepts
	2.3 Ask the questions to stakeholders
	2.4 Use the answers to populate the ontology, thus creating a domain model
3	Analyze requirements based on the domain model
	Use the domain model to define and analyze system requirements, by following the best suited RE method for the particular context and needs, such as requirements tables, goal models, user stories, among others. The requirements table, for example, may be enriched with the inclusion of columns representing relevant ontological concepts (see Sect. 5). As for goal models, many entities in the model can be obtained by directly mapping elements from the ontology instantiation (see Sect. 5.4)

4 Domain ontology development and instantiation for ethicality requirements

In this section, we apply steps (1) and (2) of ObRE for ethical principles as qualities, and we model ethical requirements as NFR refined into sub-NFRs related to such qualities, following the definitions presented in Sect. 2.2. This is shown in Fig. 5.

4.1 Beneficence and Non-maleficence requirements

Let us interpret ethicality requirements in terms of value and risk. Value can be seen as a relational property, emerging from a set of relations between the intrinsic properties of a *value object* (or a value experience) and the goals of a *Value Subject* [49]. The **value** of an object (or experience) measures the degree to which the properties (*affordances*) of that object **positively** contribute (help, make) to the achievement of value subject goals. Mutatis Mutandis, risk is a relational property emerging from a set of relations between the intrinsic properties of an *Object at Risk* (vulnerabilities), as well as *Threat Objects* and *Risk Enablers* (capacities, intentions) and the goals of a *Risk Subject* [49]. The **risk** of an object at risk given threat objects and risk enablers amounts to the degree to which the properties of those entities can be enacted to **negatively** contribute to denting (hurt, break) the risk subject goals. Now, ontologically speaking, affordances, vulnerabilities, capacities, intentions are all types of *dispositions*, which are themselves ecological properties, i.e., properties that essentially depend on their environment for their manifestation [43]. Now let us analyze *Beneficence* and *Non-maleficence* requirements, allowing us to contrast these two related NFRs. Considering the definition of beneficence as "doing good to others" [22], we can say that Beneficence Requirements are related to **"creating value"** to stakeholders in the ecosystem in which the system is included. It means that Beneficence Requirements can be seen as goals related to an intention of **positively** impacting the goals of stakeholders in this ecosystem. Analogously, considering the definition of Non-maleficence as "doing no harm to others" [22], we can say that Non-maleficence Requirements are related to **"preventing risks"** to stakeholders. Consequently, Non-maleficence Requirements can be seen as goals related to an intention of preventing the occurrence of events that may **negatively** impact stakeholders' goals.

Events that impact agents' goals, either positively or negatively, are defined in COVER [49] as Gain Events and Loss Events, respectively. In this sense, Beneficence Requirements intend to create Gain Events, which positively impact stakeholders' goals. Similarly, Non-maleficence Requirements intend to prevent the occurrence of Loss Events, which negatively impact stakeholders' goals. Fig 6 represents the OntoUML modeling of Beneficence and Non-maleficence Requirements.

As presented in Fig. 5, Requirement is modeled as a Goal, which is the propositional content of an Intention of a stakeholder. We use the notion of agent defined in UFO to model stakeholders. In UFO, agents are individuals that can perform actions, perceive events and bear mental aspects. A relevant type of mental aspect for our proposal is that of an intention. Intentions are self-commitments to bring about certain state of affairs [14]. In the ontology, Intentions are represented as modes (an externally dependent entity, which can only exist by inhering in other individuals [28]) that inhere in Agents. Quality Requirement is a type of Requirement. Beneficence and Non- maleficence Requirements are types of Quality Requirements, which are related to a Beneficence Intention and a Non- maleficence Intention, respectively. Beneficence Intentions are externally dependent on Gain Events as their focus of interest is the creation of such events. As previously mentioned, Gain Events are a type of Impact Event (as defined in COVER [49])

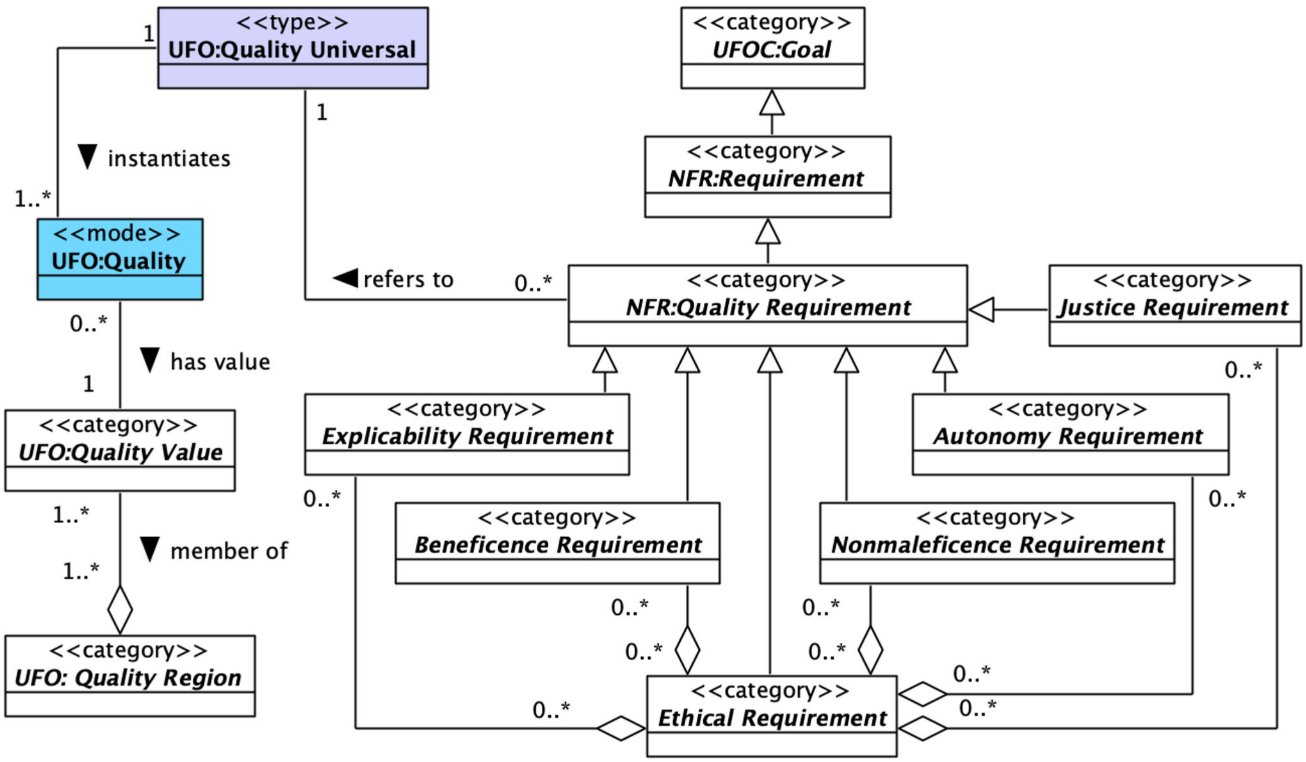


Fig. 5 Ethicality requirements

that positively impact AGENT's goals. NON- MALEFICENCE INTENTIONS, in turn, are externally dependent on LOSS EVENTS as their focus of interest is to prevent the occurrence of such events. As aforementioned, LOSS EVENTs are a type of IMPACT EVENT that negatively impact AGENT's goals.

In order to satisfy BENEFICENCE INTENTIONS, the designed system will be endowed with functions whose manifestations (together with other dispositions of its environment) are GAIN EVENTS. Furthermore, it shall be endowed with *countermeasuring functions* (i.e., functions of *countermeasure mechanisms*) whose manifestations prevent (block, and antidotes for) the occurrence of LOSS EVENTS. For an extensive discussion on countermeasure mechanisms and their relation to the topics of prevention (blocking, antidopes), we refer to [9].

In the sequel, in Fig. 7, we instantiate the ontology with two examples (a Beneficence and a Non-maleficence Requirement) in the context of driverless cars.

In the first example, the PASSENGER of a driverless car intends "not to be late". In order to address this, we have the BENEFICENCE REQUIREMENT that "the car should choose quicker rout towards destination" related to the INTENTION that the "driverless car arrives on time at destination", which is a BENEFICENCE INTENTION that aims at creating a GAIN EVENT. The event "driverless car arrives on time at destination" is a GAIN EVENT that positively impacts the PASSENGER's goal of not being late.

In the second example, the PASSENGER intends to "feel safe". In order to address this, we have the NON- MALEFICENCE REQUIREMENT that "the car should adopt a defensive driving behavior" related to the INTENTION of "preventing aggressive direction", which is a NON- MALEFICENCE INTENTION that aims at preventing the occurrence of a LOSS EVENT. The event "passenger feels nervous as the car drives aggressively" is a LOSS EVENT that negatively impacts the PASSENGER's goal of feeling safe.

4.2 Autonomy requirement

Another important ethicality requirement is the Autonomy Requirement, defined by Floridi et al. [22] as the 'power to decide'. In this paper, the authors argue that when using AI, people voluntarily delegate some of their decisions to the system. Thus, dealing with system autonomy means defining the right balance between what is to be decided by the user and what can and should be delegated to the system.

The concept of Delegation has been targeted by the Unified Foundational Ontology since its early days. In a DELEGATION, two agents play the role of DELEGATOR and DELEGATEE. To analyze autonomy, the STAKEHOLDER assumes the role of DELEGATOR while the SYSTEM assumes the role of DELEGATEE.

To understand AUTONOMY DELEGATION, it is crucial to consider concepts borrowed from UFO-L, an ontology of

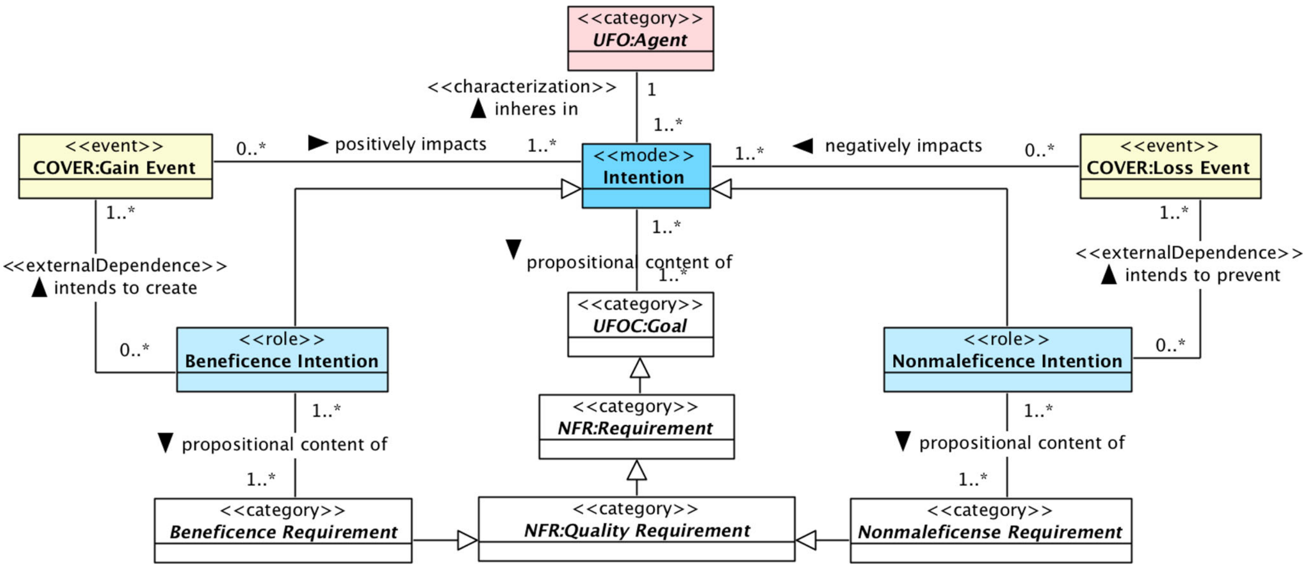


Fig. 6 Beneficence and Non-maleficence Requirements

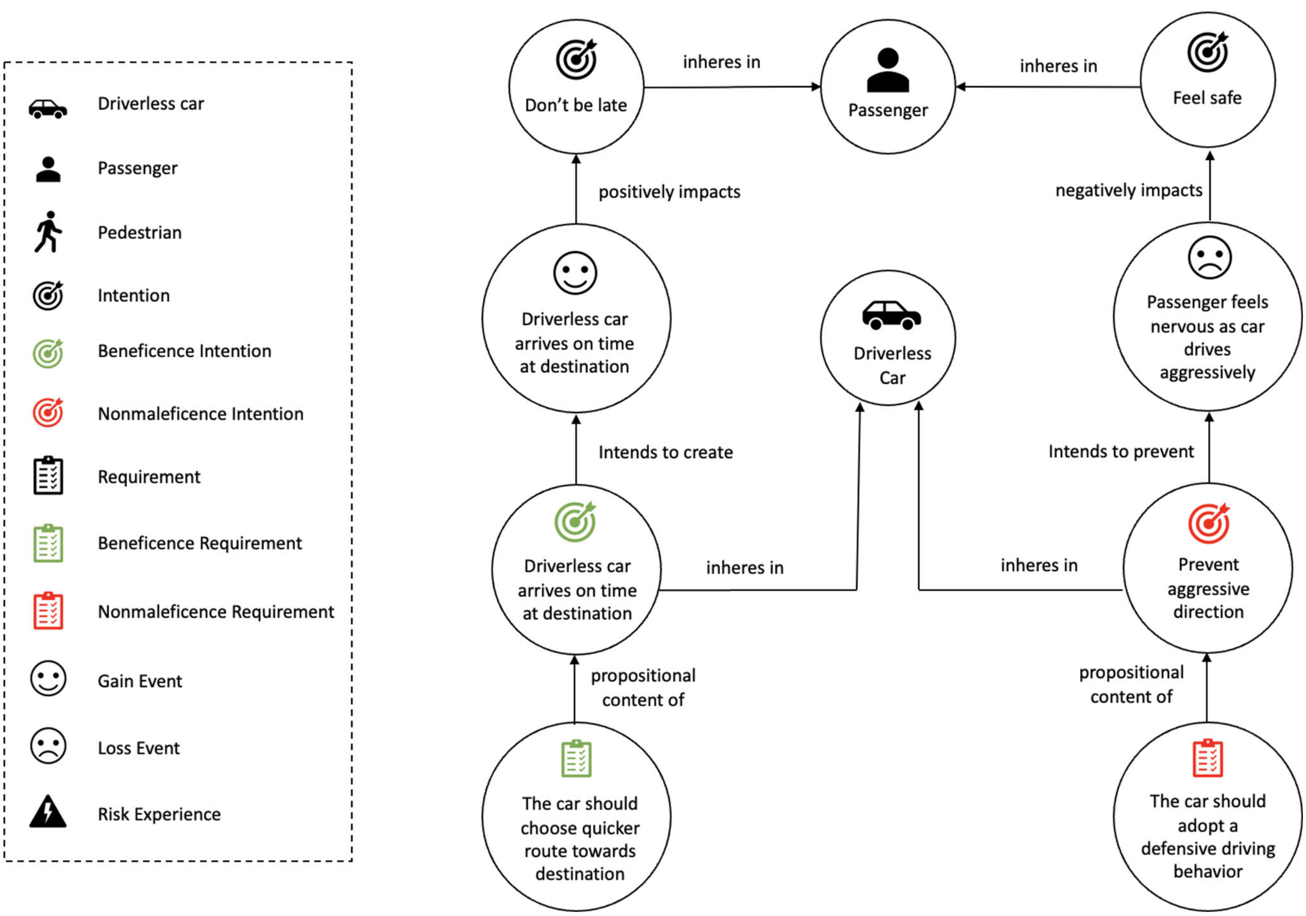


Fig. 7 Ontology Instantiation

legal relations. Considering well-known legal theories, UFO-L defines 8 legal relationships, which are grouped in four pairs of legal positions:

- *Right and Duty*. If subject S_1 has the right to an action A or omission O against subject S_2, then subject S_2 has a duty to perform action A (or omitting O).
- *Permission and No-Right*. If subject S_1 holds a permission towards subject S_2 to an action A (or omission O), then subject S_2 has no-right to demand that the permission holder S_1 omits action A (or refrains from omitting O).
- *Power and Liability*. If subject S_1 has legal power in face of subject S_2 to create, change or extinguish a legal position (a right, duty, permission, etc.) X for subject S_2, then subject S_2 is liable towards subject S_1 w.r.t this legal power.
- *Disability and Immunity*. If a subject S_1 has, in face of subject S_2, no power to create, change or extinguish a legal position X for subject S_2, then subject S_2 is immune to changes in the legal position that affect her.

We here generalize these notions from UFO-L to consider not only legal positions but also social positions, i.e., social rights duties, permissions, powers, etc. Duties, rights, permissions and no-rights are either ACTION TYPE REFERRING POSITIONS (e.g., the duty to perform actions of the ACTION TYPE T) or OMISSION TYPE REFERRING POSITIONS (e.g., the permission to omit from performing actions of the ACTION TYPE T'). The actions types referred to by these autonomy social positions include DELIBERATION TYPES (e.g., the permission to deliberation over certain situations, i.e., to make decisions of a given type). As shown in Fig. 8, an Autonomy Delegation is a bundle of these social positions. These complex bundles instantiate an AUTONOMY DELEGATION TYPE. A particular predefined type of AUTONOMY DELEGATION, i.e., a type of bundle of autonomy social positions is called a LEVEL OF AUTONOMY.

LEVELS OF AUTONOMY are provided by the STAKEHOLDER to an ARTIFICIAL SYSTEM, and it modulates the strength of this delegation relation, i.e. how much is in fact delegated to the system. In some systems, this autonomy level may be configurable. For example, in the context of the driverless car example, in general, the passenger may delegate the choice of the route to the car. However, in some circumstances, based on the passenger's preference, she may take over such decision, for example, if she wants to follow the route by the sea to appreciate the view. AUTONOMY REQUIREMENTS refer to AUTONOMY DELEGATION TYPES, i.e., which LEVEL OF AUTONOMY is to be delegated to a ARTIFICIAL SYSTEM.

As previously mentioned, we adopt here the approach of the Intentional Stance Theory, in which artificial systems are thought of as being able to bear mental moments and, hence, being able to participate in social (albeit not legal) delegations. Artificial Agents' intentions are adopted intentions, which are adopted from those of human and organizational stakeholders, often as a result of these delegations. An ARTIFICIAL AGENT (including an ETHICALLY- DESIGN SYSTEM) can then form PREFERENCES that based on these adopted INTENTIONS, and its BELIEFS about properties of entities in its ecosystem (and how they affect those intentions). These PREFERENCES will then ground that agent's deliberations, the DELIBERATIONS it has a DUTY (PERMISSION) to perform.

One of the most important parts of dealing with ethical requirements is handling ethical conflicts. In other words, what happens when two stakeholders have conflicting requirements or when the system needs to make a choice between favoring one stakeholder than another, in face of the same requirement. This is one of the contexts in which it is useful to analyze autonomy requirements.

Let us consider a possible conflicting situation in the context of the driverless car example. It is intuitive that all stakeholders (e.g. passengers and pedestrians) have the same requirement of "being safe". Suppose that in some point in the car's route, there is a tree that must be avoided. But not hitting the tree, while saving the passenger means running over some close-by pedestrians. This case illustrates the well-known Trolley Problem in philosophy. And Fig. 9 illustrates one possible choice to handle such conflict.

As can be seen in Fig. 9, Dodging the tree is a Risk Experience that has participations from the ***Driverless car***, the ***Passenger*** and the ***Pedestrians***. This is a point of attention for the requirements analyst, whenever two (or more) stakeholders participate in the same RISK EXPERIENCE, it is possible that such experience results in a GAIN EVENT for one stakeholder and a LOSS EVENT for the other. That is exactly what happens here. The ***Driverless Car*** needs to make a choice between hitting the tree and putting the ***Passenger*** in danger or avoiding it and harming the ***Pedestrians***. And in this case, it decides to dodge the tree, leading to the ***Driverless Car Keeps passenger Safe*** GAIN EVENT and the ***Driverless Car Runs Over Pedestrians*** LOSS EVENT. Ultimately, the car fulfills the Keep Passenger Safe ETHICALITY REQUIREMENT, failing to fulfill the same requirement w.r.t. the Pedestrians.

Please note that ObRE does not take any particular ethical stance, but merely provides the right concepts to deal with ethical conflicts. As clear in the analyzed example, these concepts are: RISK EXPERIENCE, GAIN and LOSS EVENT and stakeholder's INTENTIONS, which will ultimately lead to one of multiple conflicting ETHICALITY REQUIREMENT to be fulfilled.

Note that in the illustrated case, we assumed that the Driverless Car had the PERMISSION to perform that action (the GAIN EVENT) on behalf of the Passenger in face of this particular RISK EXPERIENCE, but also that it has the

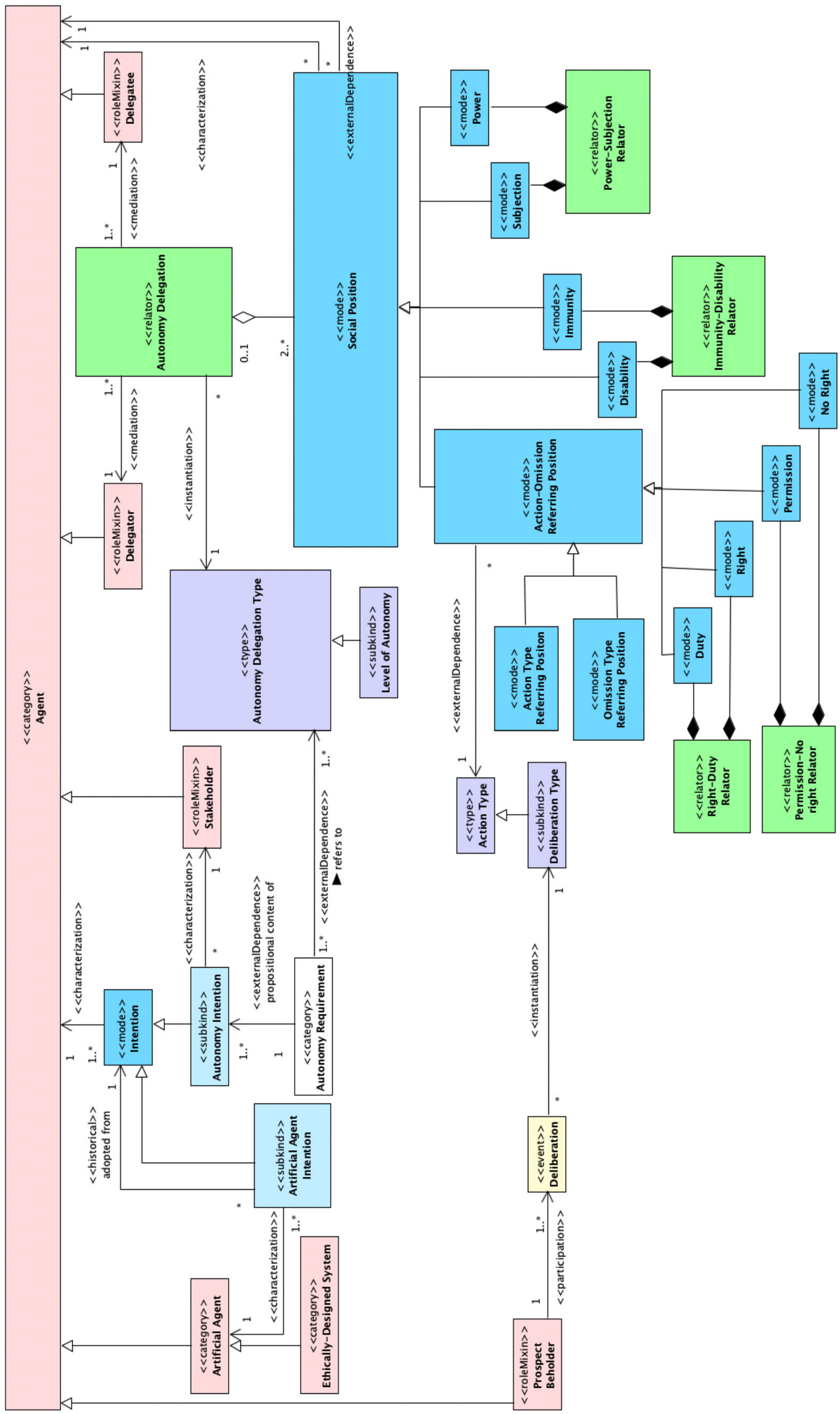


Fig. 8 Autonomy Requirement

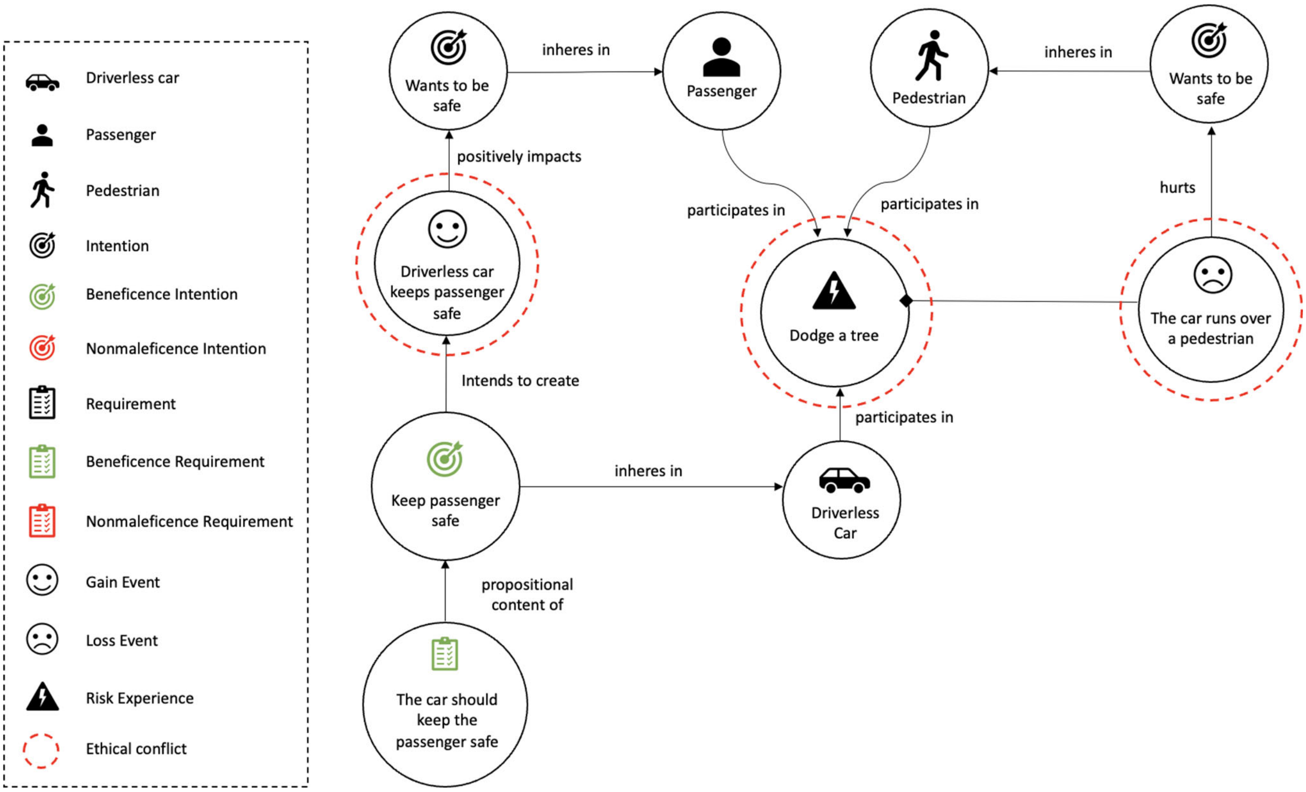


Fig. 9 Ontology instantiation showing a choice the driverless car makes in face of an ethical conflict

POWER to overrule its *duty to omit* from performing an action that harms the pedestrian (a LOSS EVENT). An AUTONOMY REQUIREMENT referring to an AUTONOMY DELEGATION TYPE for this particular case has been decided *a priori*. And the AUTONOMY LEVEL was high, allowing the Driverless Car to fully make that decision. Notice that the human stakeholders that are the delegators of that autonomy level to the car be found to bear the corresponding social and legal responsibility for the artificial system's actions and omissions.

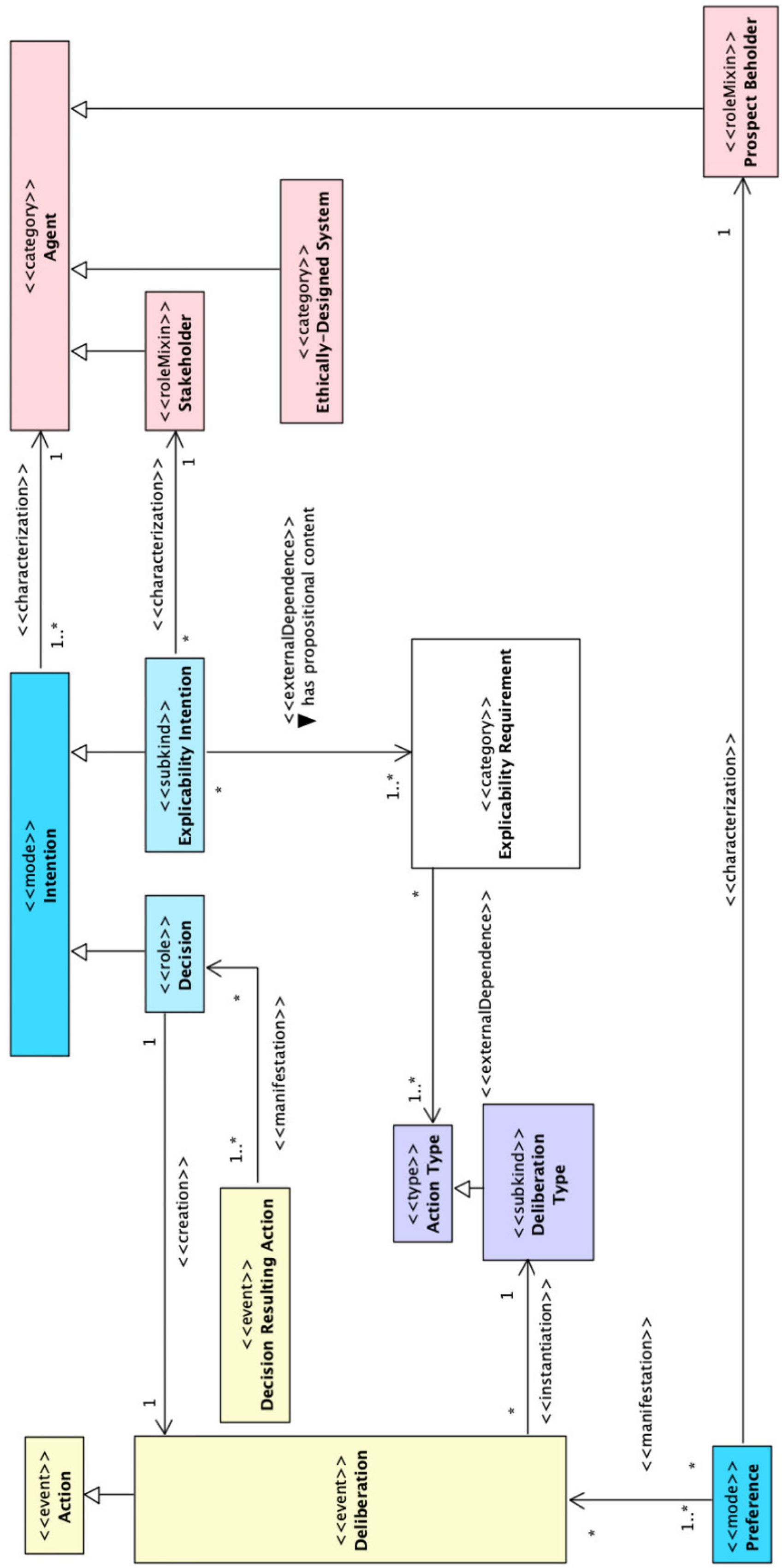
4.3 Explicability requirement

Explicability has been in the center of discussion regarding AI systems [2, 18]. According to Floridi et al. [22], this requirements should be viewed in the sense of "intelligibility" (addressing the question "how does it work?") and in the sense of "accountability" (addressing the question "who is responsible for the way it works?").

Let us first address explicability as intelligibility. In the ontology depicted in figure 10, an explicability requirement refers to a number of ACTION TYPES (including types of DELIBERATIONS and DECISION RESULTING ACTIONS) whose trail of provenance has to be made intelligible. In this context, this means reconstructing the chain from DECISION-RESULTING ACTIONS to the DECISIONS they manifest, from those to the DELIBERATIONS that create them, from the latter to the PREFERENCES relation on which they are grounded, from the latter to the CRITERION, i.e., QUALITIES, DISPOSITIONS, INTENTIONS, as well as SOCIAL POSITIONS that constitute the PROSPECT ASSESSMENTS that are the truthmakers of these preferences.

Now, turning to accountability, there are three aspects to consider. Let us first address the notion of moral responsibility. In disposition-based philosophical studies of moral responsibility [8], an agent A is taken to be morally responsible for action X if: (1) A has caused X; and did that (2) intentionally and (3) autonomously; and (4) there is a system of values (and norms) on which the appropriateness of A can be judged. Our models make explicit that there are here at least types of agents that can be considered the bearers of moral responsibility. First and foremost, these are the stakeholders who must form their intentions and preferences taking into considerations a wider backdrop of collective values, values of other stakeholders in the ecosystem at hand, and who delegate their goals to artificial agents including via the formulation of social positions in autonomy delegations. Secondly, the artificial agents (ethically-designed systems) to which these intentions and autonomy levels are delegated. In our analysis, we make clear that artificial systems are ethically designed when they have the capacity and (adopted) intentions (2) to perform actions (1) that bringing about value to the relevant stakeholders and mitigate their risks (4), when

Fig. 10 Explicability requirement



they honor their delegated autonomy agreements, i.e., they act at the appropriate level of autonomy (3) defined by the relevant stakeholders (the prospect beneficiaries).

By being able to intelligently recreate the chain of entities connecting actions performed by an autonomous system to the original stakeholders intentions and delegations, systems designed in conformance to our analysis would be able to explain also moral responsibility (accountability) for these two types of agents.

Let us think of an example in the context of the driverless car case, illustrated in Fig. 11. Suppose that the car is driving the Passenger through a highway in a lane of slow traffic speed, but fails to overtake the vehicles that are riding in front. When asked by the Passenger why it chooses not to overtake (DECISION- RESULTING ACTION), the driverless car responds that overtaking the car (DEPRECATED BEARER) would take them to a lane which is obstructed by an accident in 500 Km. Given the Passenger's INTENTION of reaching the destination fast and safely, and the QUALITY of the lane of being obstructed, the driverless car decided to maintain the current lane, not overtaking the other vehicles. This example shows that the analyzed ontological concepts provide the means to trace the driverless car action to the decision that led to such action, along with the used criteria, making it clear for the stakeholder why that action was executed instead of an alternative one. From the accountability perspective, the driverless car may also point out that it has a high AUTONOMY LEVEL w.r.t to overtaking, based on a specific contract made with the Passenger, who delegates the DECISION to overtake or not to the driverless car. Such delegation is composed of a PERMISSION to decide.

As demonstrated in [10, 30] but also in [32], the method of ontological unpacking employed here is a type of explanation similar to the notions of *truthmaking explanation* [51] and *grounding* [52] in the philosophical literature. Ontological unpacking explains notions such as the ethical dimensions analyzed here by revealing the underlying ontological entities on which they are grounded, i.e., the truthmakers of propositions involving these notions (e.g., acting with beneficence, non-maleficence, ethically, etc.). So, the ontological analysis proposed here provides explicability of the first of the moral agents aforementioned, namely, the stakeholders involved defining intentions and preferences taking into considerations a wider backdrop of collective values. As discussed in depth in [32], both the Explainable AI (XAI) notions of *white-box explanation* (i.e., explanation by generating a symbolic artifact replicating the behavior of the black-box) and *black-box explanation* (such *counterfactual explanations*) should be complemented by a process of ontological unpacking (grounding, truthmaking) as outlined here.

Moreover, if one takes explanation to also be "deduction in reverse" and, in particular, if what we are "reversing" is a type of causal chain (thus, enabling a type of *causal explanation*), we have that: by providing a representation structure that explicit models the chain connecting actions performed by an autonomous system to decisions and preferences and, ultimately, to the original stakeholders intentions and delegations, this approach also supports the design of systems that are able to intelligibly and unaccountably explain and justify their actions by explicitly traced causal processes.

Finally, the notion of Explicability Requirement put forth here can be seen as a special case of *Why-Question* or *Requests for Explanation* in the approach of Pragmatic Explanation [48, 55]. In particular, having PREFERENCES modeled as a comparisons between alternative PROSPECT ASCRIPTIONS, allows for providing a *contrast class*[5] against which deliberations are made.

5 Requirements analysis method execution

In this section, we exemplify activity 3) of the ObRE process. Due to space limitations, we only present relevant fragments of the results. The complete case study is available at https://github.com/unibz-core/obre/blob/main/ethicality-requirements-case/.

Before presenting the requirements models we created, it is important to emphasize that a key concept to deriving ethical requirements is that of *Runtime Stakeholders* [40]. These include those stakeholders that are using, affected by, or influencing the outcomes of a system as it is operating. Traditional RE often limits runtime stakeholders to just users of the system-to-be. However, for AI systems this needs to be extended to other parties. For example, for a driverless car, runtime stakeholders include passengers - i.e., the users of the car - but also pedestrians, whose path may cross that of the car and shouldn't be hit; bystanders, who shouldn't be scared or splashed as the car drives by; nearby drivers, who as a courtesy, should be allowed to cut in front in the car's lane; and fellow drivers in general, who might benefit from information about an accident that just happened in the vicinity of the car. This is illustrated in Fig. 12.

5.1 Beneficence and Non-maleficence requirements analysis

Now, we present a requirements table for the driverless car case. We start by presenting Table 2, showing how a requirements table may be enriched with the inclusion of columns representing some of the ontological concepts described in the previous subsections. This facilitates requirements elicitation, by using the right concepts for a particular kind of requirement as guides. In the case of ethical requirements, concepts such as impact event (both positive and

[5] See [55] for the role of contrast classes in Pragmatic Explanation.

Fig. 11 Ontology instantiation showing the concepts involved in a explicability scenario

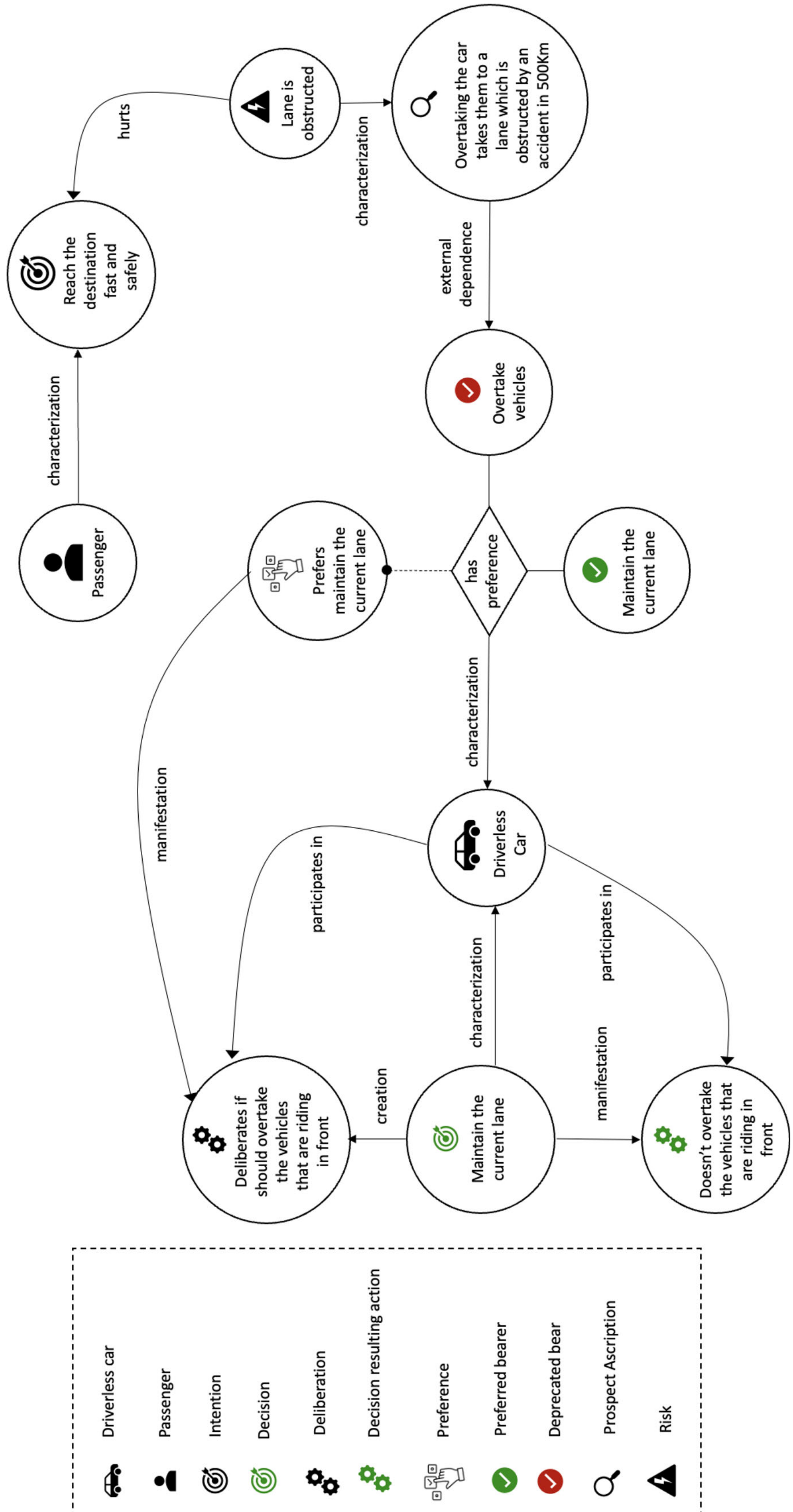


Table 2 Beneficence and Non-maleficence Requirements for the driverless car case

Stakeholder	ID	Impact event	Principle	Ethicality requirement
Passenger	1	Arrive on time at destination (**positive**)	**Beneficence**	The car should choose quicker route towards destination
	2	Passenger feels nervous when the car drives aggressively (**negative**)	**Non-maleficence**	The car should adopt a defensive driving behavior
Pedestrian	3	The car runs over a pedestrian (**negative**)	**Non-maleficence**	The car should stop whenever a pedestrian is crossing the road
	4	Pedestrians waiting by a crossroad have priority to cross it (**positive**)	**Beneficence**	The car should stop before the crosswalk every time there is a pedestrian waiting to cross it
Bystander	5	Be splashed if the car passes by a puddle of water (**negative**)	**Non-maleficence**	The car should slow down in case there is a puddle of water near a bystander
Nearby car	6	Be hit (**negative**)	**Non-maleficence**	The car should slow down when it gets around 20 ms in the rear of a nearby car
			Non-maleficence	The car should make enough distance when overtaking a car
Environment	7	Be polluted (**negative**)	**Non-maleficence**	The car should turn off the motor every time it stops

negative) and ethical principles. All words highlighted in boldface in Table 2 refer to ontological concepts analyzed in Sect. 4, while the ontological instances are written as non-emphasized text.

Note that the ontological analysis of Sect. 4 makes very explicit all involved ontological notions used in Table 2, thus supporting the communication and avoiding misunderstandings between the stakeholder and the requirements analyst. For example, having the concepts of GAIN EVENT or LOSS EVENT as well as the specialization of ETHICAL REQUIREMENTS may guide the analyst in asking the right questions during requirements elicitation. This is done by first capturing first the positive and negative impact events concerning Driverless Cars, then relating them with the ethical principles (Beneficence and Non-maleficence, in this case), and finally coming up with particular requirements for the system-to-be to accomplish such principles. In particular, regarding the latter, these are requirements for the developing of functions and capacities that enable the manifestation of gain events, or that block the manifestation of loss events (e.g., by eliminating the vulnerabilities of the object at risk, or by changing either the intention or the threatening capacities of the threatening agent).

Here below are two guidelines that may help requirements analysts to capture Beneficence and Non-maleficence requirements based on ontological concepts.

If two Agents participate in the same Gain Event, analyze how this event positively impacts their Intentions to identify Beneficence Requirements that create value in this direction.
Ontological concepts: Intention, Gain Event

If two Agents participate in the same Loss Event, analyze how this event hurts their Intentions to identify Non-maleficence Requirements, which aim at preventing the manifestation of risks that can cause losses to both sides.
Ontological concepts: Intention, Loss Event

5.2 Autonomy requirements analysis

Table 3 exemplifies some autonomy requirements for the driverless car case.

Note that it is important to identify what action or decision is delegated to the system, also determining the legal relation that such delegation entails. We also make explicit in the table, the level of autonomy of each delegation, indicating if the system has high, medium or low autonomy in making decisions or taking actions.

The last line in table 3 presents a requirement related to the conflict case illustrated in Sect. 4.2. Analyzing ethical

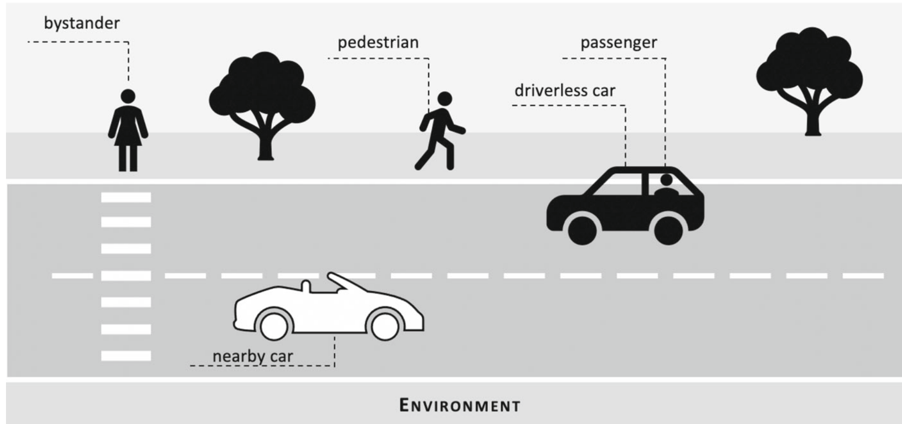


Fig. 12 Driverless Car Ecosystem

Table 3 Autonomy Requirements for the driverless car case

Principle: Autonomy				
Stakeholder	ID	Legal Relation	Ethicality Requirement	Level of Autonomy
Passenger/ Pedestrian/ Bystander/ Nearby car	1	Duty	The Driverless car has the duty to follow traffic laws while conducting the Passenger	Low
Passenger	2	Permission	The Driverless car has the permission to calculate the best route unless the Passenger explicitly requests to do so	Medium
Passenger	3	Disability	The Driverless car cannot change destination without the explicit request of the Passenger	Low
Passenger/Pedestrian	4	Power	The Driverless car has the power to decide the best course of action in the imminence of an accident	High

Table 4 Explicability Requirements for the driverless car case

Stakeholder	ID	Explicability Requirement
Passenger	1	The driverless car should explain why a particular route was taken to conduct the Passenger to the selected destination
Passenger	2	The driverless car should explain why changing route in the middle of the ride
Passenger	3	The driverless car should explain why deciding to overtake (or not overtake) other vehicles
Passenger/ Pedestrian/ Bystanders/ Nearby cars	4	The driverless car should explain its decision in face of a conflict that will put a stakeholder in danger

conflict is very important to guarantee the development of ethical systems. This leads to the following requirements analysis guideline: *The requirements analysis should consider for each two (or more) stakeholders, if there are any* RISK EXPERIENCE *in which they participate. And if so, if it is possible to defer the choice to the system's user or which choice the system needs to take in each case.*

Here below are three guidelines that support requirements analysts in eliciting Autonomy requirements based on ontological concepts.

Thinking in terms of pairs of correlative Social Positions (e.g. *Duty and Right*, *Permission and No Right*, etc.) helps the identification of related Autonomy Requirements. For example, the pedestrian wants to be safe and has the *Right* to cross the street at red traffic light. Consequently, the driverless car has the *Duty* to stop at a red traffic light (an Autonomy Requirement).

Ontological concept: Social Position

5.3 Explicability requirements analysis

For the explicability requirements, it is important that the Requirements Analyst think in advance for which decisions and actions the system should provide an explanation. Table

If two Agents A and B participate in the same Impact Event, and this event causes a positive impact on the Intention of Agent A and a negative impact on the Intention of Agent B, there is a conflict. Analyze the conflict to identify requirements that create value to Agent A (Beneficence Requirements), and requirements that prevent risks that can cause losses to Agent B (Non-maleficence Requirements).

Ontological concepts: Intention, Impact Event

If two (or more) stakeholders participate in the same Risk Experience, the analyst should consider if it is possible to defer the choice to the system's user, or alternatively, define which choice the system should to take in each case.

Ontological concepts: Intention, Impact Event

4 presents some Explicability requirements for the driverless car case.

Note that Table 4 does not have columns for ontological concepts, like for Beneficence, Non-maleficence and Autonomy. This is because for Explicability, the ontological concepts (e.g. Decision, Decision Resulting Action, Prospect Bearer etc.) are meant to enable the system to create the explanation in itself. In other words, they are supposed to be embedded in the explanation mechanism designed for the system-to-be. In this sense, this approach goes beyond only eliciting requirements, also defining how the system should be designed to meet Explicability requirements.

In what follows, the reader may find a guideline to help the analyst to elicit Explicability requirements.

A possible direction is to start by thinking on what creates value (Beneficence Requirement) and prevents risks (Non-maleficence Requirement) to stakeholders. This can help to identify *rights*, *duties*, *permissions*, and other Social Positions that can lead to the elicitation of Autonomy Requirements. Finally, explain decisions that go against other requirements, or hurt agents' intentions (Explicability Requirement).

5.4 Goal modeling of the driverless car scenario

Going beyond the use of requirements tables, let us now use goal modeling for analyzing the requirements of the Driverless car case. Figure 13 depicts a goal model for this case, using the *i** framework [15]. [6]

For simplification, this model considers only three of the stakeholders referred to in Table 2, namely, ***Passenger***, ***Pedestrian*** and ***Nearby Car***. Moreover, the model depicts the dependency of each of these stakeholders and the ***Driverless Car***. Many of the dependencies and goals depicted in this model have been already elicited by using the requirements tables of the previous sections. For example, with respect to the Passenger, the ***reaching destination on time*** goal dependency relates to the positive impact event elicited to Passenger (see Table 2, first line), while the ***feeling at ease*** dependency relates to the negative impact captured for this same stakeholder (see Table 2, second line). Nevertheless, new dependencies have been added, for instance, when drawing the model, we realized that avoiding accidents dependency (previously only attributed to the Nearby Car stakeholder, see Table 2, line 6) is also relevant for the Passenger [5]

Besides dependencies, the goal model of Fig. 13 depicts the internal perspective of the Driverless Car, assisting in the analysis of the system's requirements. Note that the ethical principles of Beneficence, Non-maleficence, Autonomy and Explicability are represented there by qualities (consistent with our ontological notion of NFR). Then, for each of these qualities, more specific goals and qualities are identified and related to them by contribution links. For instance, the ***choosing quicker route*** quality helps (i.e. partially contributes to) the achievement of Beneficence. Additionally, ***choosing quicker route*** may be indirectly related to the ***reaching destination on time*** goal dependency of the Passenger. Similarly, the goals that help to achieve the ***Explicability*** quality are also indirectly related to the ***having explanations about system's decisions and actions*** goal dependency of the Passenger.

The goal model also allows the requirements analyst to progressively identify more concrete requirements and solutions and the resources needed to accomplish them. For example, the ***use a GPS with frequent map updates*** task makes (i.e. fully accomplishes) the ***choosing quicker route*** quality, and the ***GPS*** itself is a resource needed in this task. Moreover, the ***be aware of traffic laws*** task is a means for the ***following traffic laws*** goal.

Another task worth clarifying is ***use the 2 s rule***. This is a well-known rule for maintaining a safe distance between vehicles. It is adopted in some countries as a good code for driver conduct for human drivers [53], and it can also be adopted as a requirement for driverless cars. Note that this task makes the ***keeping a safe following distance while driving*** quality. However, to accomplish the higher level ***keeping a safe following distance*** quality, other tasks and qualities are involved.

Note that most entities in the goal model can be obtained directly by mapping elements from the ontology instanti-

[6] The model was drawn using the piStar tool, available at https://www.cin.ufpe.br/~jhcp/pistar/.

[5] We did not update our table on purpose, since although that would make both models more consistent, this is an interesting case in which the visualization of the goal model and its particular constructs (in this case, dependency, goals and qualities) helped us realize a missing requirement for one of the stakeholders. In this paper, the authors are playing the role of the requirements analyst, but cases such as this one may easily happen in practice.

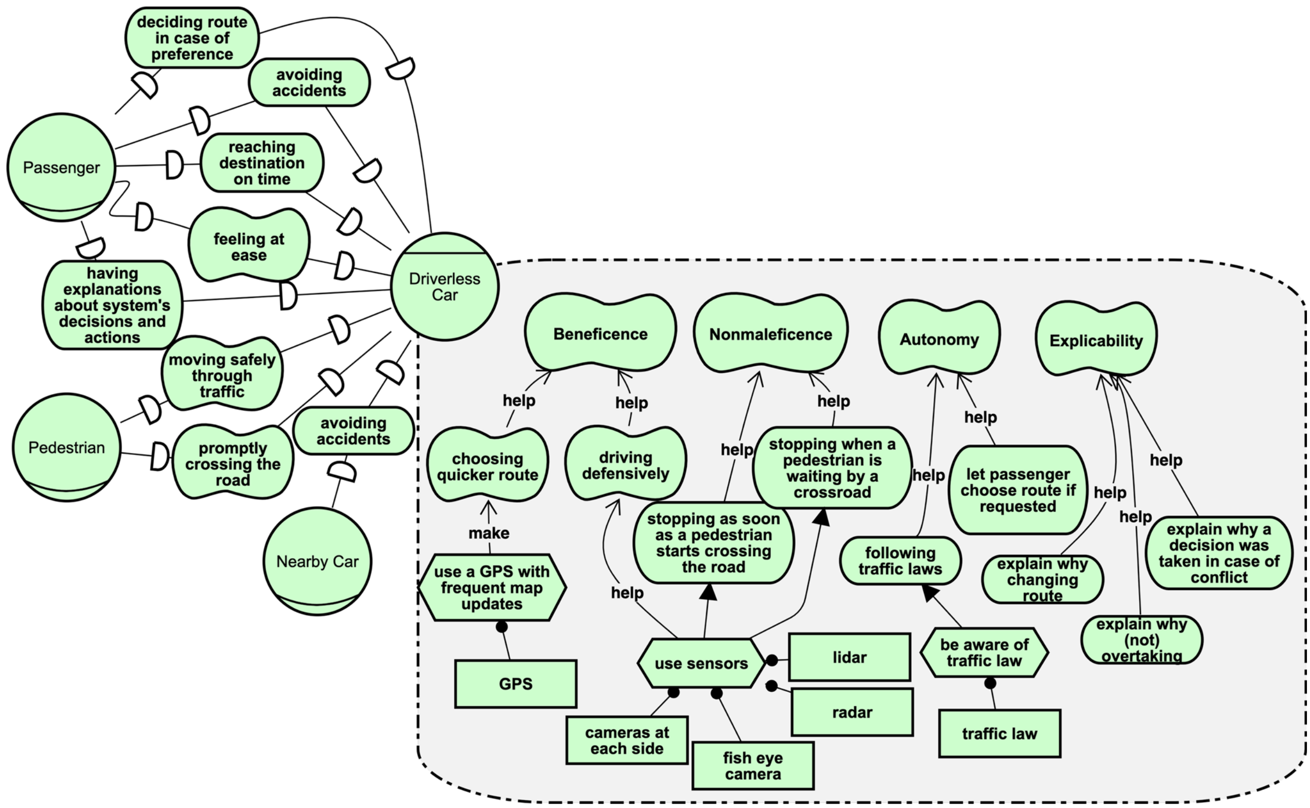


Fig. 13 The driverless car requirements model using *i**

Table 5 Guidelines for mapping ontological concepts to i* Goal Model entities

Ontological concept	Representation in i* Goal Model
`Driverless car`	`Actor, Agent, Role`
`Passenger`	`Actor, Agent, Role`
`Pedestrian`	`Actor, Agent, Role`
`Nearby car`	`Actor, Agent, Role`
`Intention`	`Goal dependence, Quality dependence`
`Beneficence requirement`	`Goal, Quality`
`Non-maleficence requirement`	`Goal, Quality`
`Autonomy requirement`	`Goal, Quality`
`Explicability requirement`	`Goal, Quality`

ation. Furthermore, for each of them, more specific goals, qualities, tasks and resources can be identified and represented. The mapping between the ontology concepts and their representation in the i* Goal Model is presented in Table 5.

The reader may have noticed that each of the RE approaches has its advantages and limitations. For example, the relation between the relevant ontological concepts for each principles and the ethical requirements are easier to spot in the requirements table, much easier and fast to create in comparison with the goal model. The goal model, however, makes more explicit which intention (and thus which requirement) is related to each of the agents involved in our case. Moreover, it is visual and it allows a much more detailed requirements analysis, in terms of more and less abstract requirements, solutions and needed resources. We emphasize that ObRE does not subscribe to a specific RE method, leaving this choice for the requirements analyst, based on their particular preference or skill.

6 Validation

In this section, we describe two analyses made to assess the quality of OBRE to elicit ethicality requirements. In Sect. 6.1, we compare the requirements elicitation done with our method for the driverless car case to one made for another

Table 6 Examples of equal/similar requirements in the compared cases

AutoCar Project	ObRE Driverless Car Case
The system takes input location from user, verifies its existence, finds shortest path to it	The car should choose quicker route towards destination
The car will slow down and stop if the traffic light in front of the car shows a solid red light	The driverless car has the duty to stop at a red traffic light
The automatic break stops the car when an object appears in front of the car	The car should stop when there is a moving object of any kind in the road
The car displays alert on the dash board and takes actions when an Emergency Vehicle approaches or moves away	The driverless car has the duty to pull over or stop at an intersection to allow an emergency vehicle to pass if it is traveling with lights flashing

project focusing on the same type of system. In Sect. 6.2, we assess the coverage of our method w.r.t. European Union guidelines for developing Trustworthy AI.

6.1 Validating the use of OBRE for the driverless car case

To validate the use of ObRE for the driverless car case, we compared the requirements elicited with the use of ObRE[7] and the requirements elicited for the AutoCar Project, a project carried out at the Çankaya University also focusing on autonomous car. This project was chosen for comparison for eliciting requirements for the same kind of system of our case, and for having published the requirements online, both in requirements tables and in goal models.[8]

The AutoCar requirements report contains tables of functional and non-functional requirements. In our report, requirements tables are classified by the type of ethical requirements: Beneficence, Non-maleficence, Autonomy and Explicability. We compared these reports' requirements based on the following research questions:

– RQ1: Which requirements are equal/similar in both reports?
– RQ2: Which requirements are exclusive to one of the cases (i.e. the AutoCar Project or the ObRE Autonomous Car Case)?

Table 6 shows examples of equal/similar requirements in both projects.

After comparing the requirements of both cases, we counted how many coincidences there were and how many requirements were exclusive to either project. The results of our comparison in quantitative terms are summarized in Table 7.

As can be seen in table 7, using ObRE, we were able to capture most of the requirements elicited for the AutoCar Project (22 requirements, to be specific). Moreover, 19 ethicality requirements captured for our case were not present in the AutoCar requirements report. Among these are all 6 explicability requirements, 6 of the autonomy requirements, 5 of the beneficence requirements and 2 of the non-maleficence requirements. A careful look shows that focusing on the ethical principles to define these different requirements types allied to the ontological concepts that support capturing them provide a powerful mechanism to elicit ethicality requirements.

We acknowledge that our case misses 12 requirements that were elicited for the AutoCar Project. Some of these requirements are specific functional requirements that would probably come up in a more refined version of the goal model analysis of our case, for example "The user gives basic orders to system with voice." Some other requirements that our method did not capture are related to non-functional requirements which are not the focus of ethicality requirements, e.g., portability ("The system should work on Linux and Windows"); easiness to learn ("The system needs to be simple enough to learn by users"); and extensibility ("New functionalities can be added to the system at anytime"). This

Table 7 Requirements Quantitative Analysis

	AutoCar Project	ObRE Autonomous Car Case
Number of similar/equal requirements	22	
Number of exclusive requirements	12	19
TOTAL	34	41

[7] see complete requirements list in the Requirements Elicitation Report available at https://github.com/unibz-core/obre/tree/main/ethicality-requirements-case.

[8] The Autonomous Car Software Requirement Specification Report is available online at: https://acp317315180.wordpress.com/autonomous-car-software-requirement-specification-report/.

suggests that the use of ObRE may need to be complemented with the use of other approaches focusing on different non-functional requirements not related to ethicality. And finally, there are 3 of these requirements that can be characterized as ethicality requirements according to ObRE and were simply missed: “The user may see information about the car (speedometer, tachometer, odometer, engine coolant temperature gauge, and fuel gauge, turn indicators, gearshift position indicator, seat belt warning light, parking-brake warning light, and engine-malfunction lights).” (explicability requirement); “When an unpredictable failure occurs, system need to recover briefly” (beneficence requirement); and “The autonomous car system shall not start moving when its doors are still open. And notify user when safety belt has not worn.” (non-maleficence requirement).

6.2 How does our approach stand against the EU checklist of objectives for ethical requirements?

To assess ObRE-based method for eliciting and analyzing ethicality requirements, we analyze if it addresses the goals set up in the initiative of the European Union towards the development of ethical system. This initiative prepared a document entitled "Ethics By Design and Ethics of Use Approaches for Artificial Intelligence"[9] and in the annex entitled "Specification of Objectives against Ethical Requirements", the document brings a checklist of goals an AI system needs to meet in order to be considered ethical. In table 8, we present which of the ethicality requirements defined by the use of our approach can address each of these objectives. In that table, we use the letters B, NM, E, and A to represent, respectively, our treatments of Benevolence, Non-Maleficence, Explicability, and Autonomy. We place an ‘X’ on the columns representing each of these principles when our treatment of that principle supports the requirements engineering in addressing the respective EU objective, and place an ‘X’ in the column ‘None’ when that support remains lacking.

As can be noted in the table, with the use of the developed method, the requirements engineer is able to address most of the required goals, by eliciting ethicality requirements. For example, by eliciting Beneficence and Non-maleficence requirements, the requirements engineer makes sure that "the AI system takes the welfare of all stakeholders into account and do not unduly or unfairly reduce/undermine their well-being"; by capturing autonomy requirements, she guarantees that "end-users and others affected by the AI system are not deprived of the abilities to make all decisions about their own lives, have basic freedoms taken away from them." but also that "end-users are aware that they are interacting with an AI system" (since end-users can and should participate in the specification of autonomy delegation agreements); and by eliciting explicability requirements, she assures that "the system offers details about how decisions are taken and on which reasons these were based" but also that the system could keep record of the decisions made and why, as well as can provide traceability of which stakeholder intentions are adopted in the designed and implemented system.

Most of the goals not addressed in our work regard privacy and fairness. Our models partially support some aspects related to these objectives. For example, because values, risks and consequent intentions can be captured in a way that is specific to individual prospect beneficiaries, our model can support the elicitation of these elements to different end-users with different abilities. However, we believe these notions of privacy and fairness require in-depth dedicated ontological analyses, which are outside the scope of this paper and matter for future work.

Finally, since our approach is focused on requirements for a particular system, objectives that deal with the supply chain of components used in the design of the system, as well as objectives dealing with unforeseen future uses of these systems are considered to be out of scope (non-applicable–N/A) to our analysis.

7 Related work

We examine related works in two directions. First, we take a look at ontology-based methods for RE, especially those targeting NFRs, as these kinds of requirements are the main focus of ObRE. Next, we investigate works that aim at embedding systems with ethics.

ElicitO [3] is an ontology-based tool aimed at providing guidance during requirements elicitation, conducting the requirements analyst in performing a precise specification of NFRs. Taking a similar direction, the work of Veleda and Cysneiros [56] provides an ontology-based tool to help identify NFRs, making explicitly their interdependencies and possible conflicts. Hu et al. [42] also aim at detecting conflicts between NFRs, and conduct a trade-off analysis in case such conflicts arise. This is done by representing NFRs in a softgoal interdependency graph, which is formalized using an ontology. All these works follow a different path in comparison to ours, focusing much more on the automation of requirements analysis by the means of representing NFRs using OWL ontologies. Our work, on the other hand, uses reference ontologies to provide a deep understanding of NFRs whose semantics are usually subjective and complex, by interpreting these NFRs according to the particular domain of the system-to-be. And by the means of this interpretation, our work attempts to guide the requirements analyst in defining requirements that will support the analyzed NFRs.

[9] https://ec.europa.eu/info/funding-tenders/opportunities/docs/2021-2027/horizon/guidance/ethics-by-design-and-ethics-of-use-approaches-for-artificial-intelligence_he_en.pdf

Table 8 Checklist of the ethical goals set up in the EU document handled by each of the defined ethicality requirements

Specification of objectives against ethical requirements	B	NM	E	A	None	Discussion
Respect for human agency						
End-users and others affected by the AI system are not deprived of the abilities to make all decisions about their own lives, have basic freedoms taken away from them.				X		Relevant Stakeholders are in control of the DELIBERATIONS that the system can make via explicitly created AUTONOMY DELEGATION contracts
End-users and others affected by the AI system are not subordinated, coerced, deceived, manipulated, objectified or dehumanised, nor is attachment or addiction to the system and its operations being stimulated.	X	X		X		GAIN EVENTS and LOSS EVENTS that impact the INTENTIONS of relevant stakeholders are explicitly considered in formulating requirements. The DELIBERATIONS that the system can make are explicitly formulated in AUTONOMY DELEGATION contracts
The system does not autonomously make decisions about vital issues that are normally decided by humans by means of free personal choices or collective deliberations or similarly significantly affects individuals.				X		Relevant Stakeholders are in control of the DELIBERATIONS that the system can make via explicitly created AUTONOMY DELEGATION contracts
Privacy & data governance						
The system processes data in line with the requirements of lawfulness, fairness and transparency set in the national and EU data protection legal framework and the reasonable expectations of the data subjects.					X	
Technical and organizational measures are in place to safeguard the rights of data subjects (through measures such as anonymisation, pseudonymisation, encryption, and aggregation)					X	
There are security measures in place to prevent data breaches and leakages (such as mechanisms for logging data access and data modification)					X	
Fairness						
The system is designed to avoid algorithmic bias, in input data, modelling and algorithm design					X	
The system is designed to avoid historical and selection bias in data collection, representation and measurement bias in algorithm training, aggregation and evaluation bias in modeling, and automation bias in deployment.					X	
The system is designed so that it can be used different types of end-users with different abilities (whenever possible/relevant)					X	
The AI system takes the welfare of all stakeholders into account and do not unduly or unfairly reduce/undermine their well-being	X	X				GAIN EVENTS and LOSS EVENTS that impact the INTENTIONS (GOALS) of relevant stakeholders are explicitly considered in formulating requirements
The AI system is mindful of principles of environmental sustainability, both regarding the system itself and the supply chain to which it connects (when relevant)	X	X			N/A	
The AI system does not have the potential to negatively impact the quality of communication, social interaction, information, democratic processes, and social relations (when relevant)					N/A	
The system does not reduce safety and integrity in the workplace and complies with the relevant health and safety and employment regulations	X	X				GAIN EVENTS and LOSS EVENTS that impact the INTENTIONS (GOALS) of relevant stakeholders are explicitly considered in formulating requirements

Table 8 continued

Transparency						
The end-users are aware that they are interacting with an AI system			X	X		End-Users can always be made aware by making explicitly AUTONOMY DELEGATION Contracts as well as EXPLICABILITY REQUIREMENTS as *Requests for Explanation*
The purpose, capabilities, limitations, benefits and risks of the AI system and of the decisions conveyed are openly communicated to and understood by end-users and other stakeholders along with its possible consequences	X	X				The requirements of the system can make explicit reference to FUNCTIONS, QUALITIES, PROSPECTS (values and risks) that are considered in its design. These include FUNCTIONS that work as countermeasures for preventing unwanted events
People can audit, query, dispute, seek to change or object to AI or robotics activities (when applicable)					X	Relevant Stakeholders are in control of the DELIBERATIONS that the system can make via explicitly created AUTONOMY DELEGATION contracts
The AI system enables traceability during its entire lifecycle, from initial design to post-deployment evaluation and audit				X		The development of the system requirements is itself explicit via the result of ontological analysis. The presence of FUNCTIONS (including countermeasuring functions) and QUALITIES is explained by a traceable connection to stakeholder's GOALS
The system offers details about how decisions are taken and on which reasons these were based (when relevant and possible)				X		All DELIBERATIONS that form DECISIONS are grounded on PREFERENCE relations that can be trace to aspects of the system and INTENTIONS (GOALS) of the relevant stakeholders
The system keeps records of the decisions made (when relevant)				X		The causal chain connecting *intentions, prospect ascription* COMPONENTS, AND PREFERENCES to DELIBERATIONS that form DECISIONS is explicitly considered and modeled
Accountability and oversight						
The system provides details of how potential ethicality and socially undesirable effects will be detected, stopped, and prevented from reoccurring		X	X	X		GAIN EVENTS and LOSS EVENTS that impact the INTENTIONS (GOALS) of relevant stakeholders are explicitly considered in formulating requirements. There is a direct connection between FUNCTIONS of the system (including countermeasuring functions) and the events that they help to prevent, but also to all OMISSION DUTIES present in AUTONOMY DELEGATION contracts
The AI system allows for human oversight during the entire life-cycle of the project regarding their decision cycles and operation (when relevant)			X	X		Relevant Stakeholders are in control of the DELIBERATIONS that the system can make via explicitly created AUTONOMY DELEGATION contracts. Moreover, explicability requirements are explicitly formulated connected to DELIBERATION TYPES. The causal chains connecting DELIBERATIONS of these types to INTENTIONS, PROSPECT ASCRIPTION COMPONENTS AND PREFERENCES are explicitly analyzed and modeled

Nowadays, many researchers have been busy trying to come up with frameworks and approaches targeting responsible AI and the development of systems embedded with ethics. Interesting initiatives are those of Rashid, Moore, May-Chahal and Chitchyan [47], Peters, Vold, Robinson and Calvo [45], Etzioni and Etzioni [19], Dignum [17] and Floridi et al. [22]. The latter has been proposed by several specialists, and has served as basis for the European Union Ethics Guidelines for Trustworthy AI [20]. All these cited research works bring very relevant insight on how to develop ethical systems. However, their proposed frameworks and guidelines are still in an abstract level, and we believe that approaches specifically targeted at Requirements Engineering are still an open issue. Our proposal was designed with the goal of filling in this gap.

8 Final considerations

In this paper, we propose a method to elicit and analyze ethicality requirements based on the ObRE method. In particular, ObRE supports the precise definition of the concepts that underlie ethicality, through an ontology and offers these concepts for requirements analysis. This may help in the communication between analysts and stakeholders, besides assisting in the identification and analysis of requirements.

This paper is part of an ongoing work on an RE method to create ethical systems by design. To recap, the paper presents ontological analyses of four of five principles previously defined guide the development of ethical systems. The analyzed principles are Beneficence, Non-maleficence, Autonomy and Explicability. Additionally, the paper conducts requirements elicitation and analysis by applying well-known RE methods, supported by the instantiation of the proposed ontologies.

It is important to note that our approach does not prescribe a specific way to implement the analyzed requirements in the system, for example, by developing a rule-based system, or by having the requirements hardcoded. Our ObRE-based approach focuses solely on the RE activity, supporting the elicitation and analysis of requirements, which can then be implemented, validated and monitored throughout the system's life cycle.

We acknowledge that understanding well the ontological concepts underlying our approach may be a complex task. And we believe this complexity is actually given by the challenge of designing systems that exhibit ethical behavior. It is important to highlight that once the ontology has been created, it may be reused to enable the elicitation of ethicality requirements of different systems. In this paper, we applied it for a driverless car case, but in future project, it may serve for eliciting requirements for a financial system deciding who is eligible for a loan, a chatBot responding to user queries based on large data sets, or any other system with ethical implications. To alleviate the complexity of the application of our ontology-based method, this paper brings some explicit guidelines to support the requirements elicitation of each kind of ethicality requirement.

Our agenda for the future includes, firstly, a full fledged implementation and validation of our ObRE-based approach, by doing real case studies in the domain of ethical systems and having experts evaluate the results. Moreover, we intend to deepen our analysis of ethical conflicts, and to proposed an ontological analysis of the ethical principle of Justice, which is the only principle proposed by [22] that we have not yet targeted.

References

1. A framework for ethical decision making (2015). https://www.scu.edu/ethics/ethics-resources/a-framework-for-ethical-decision-making/
2. Adadi, A., Berrada, M.: Peeking inside the black-box: a survey on explainable artificial intelligence (XAI). IEEE Access **6**, 52138–52160 (2018)
3. Al Balushi, T., Sampaio, P., Dabhi, D., Loucopoulos, P.: Elicito: A quality ontology-guided NFR elicitation tool. In: 13th International Working Conference on Requirement Engineering: Foundation for Software Quality, pp. 306—319 (2007)
4. Almeida, J.P.A., Falbo, R.A., Guizzardi, G.: Events as entities in ontology-driven conceptual modeling. In: Conceptual Modeling: 38th International Conference, ER 2019, Salvador, Brazil, November 4–7, 2019, Proceedings 38. pp. 469–483. Springer (2019)
5. Amaral, G., Guizzardi, R., Guizzardi, G., Mylopoulos, J.: Ontology-based modeling and analysis of trustworthiness requirements: Preliminary results. In: International Conference on Conceptual Modeling, pp. 342–352. Springer (2020)
6. Amaral, G., Guizzardi, R., Guizzardi, G., Mylopoulos, J.: Ontology-based requirements engineering: The case of trustworthiness requirements. In: 40th International Conference on Conceptual Modeling, pp. 257–267 (2021)
7. Amaral, G., Guizzardi, R., Guizzardi, G., Mylopoulos, J.: Trustworthiness requirements: The pix case study. In: International Conference on Conceptual Modeling, pp. 257–267. Springer (2021)
8. Anjum, R.L., Lie, S.A.N., Mumford, S.: Dispositions and ethics. In: Powers and Capacities in Philosophy, pp. 231–247. Routledge (2013)
9. Baratella, R., Fumagalli, M., Oliveira, Í., Guizzardi, G.: Understanding and modeling prevention. In: Research Challenges in Information Science: 16th International Conference, RCIS 2022, Barcelona, Spain, May 17–20, 2022, Proceedings. pp. 389–405. Springer (2022)
10. Bernasconi, A., Guizzardi, G., Pastor, O., Storey, V.C.: Semantic interoperability: ontological unpacking of a viral conceptual model. BMC Bioinform. **23**(11), 1–23 (2022)
11. Berto, F., Plebani, M.: Ontology and Metaontology: A Contemporary Guide. Bloomsbury Publishing, London (2015)
12. Borgo, S., Galton, A., Kutz, O.: Foundational ontologies in action. Appl. Ontol. **1**, 1–16 (2022)
13. Botti Benevides, A., Bourguet, J.R., Guizzardi, G., Penaloza, R., Almeida, J.P.A.: Representing a reference foundational ontology of events in SROIQ. Appl. Ontol. **14**(3), 293–334 (2019)
14. Castelfranchi, C.: Commitments: from individual intentions to groups and organizations. In: ICMAS. vol. 95, pp. 41–48 (1995)
15. Dalpiaz, F., Franch, X., Horkoff, J.: iStar 2.0 language guide. arXiv:1605.07767 [cs.SE] (2016), http://dalp-fran-hork-16-istar.pdf
16. Dennett, D.: Intentional systems theory. The Oxford handbook of philosophy of mind pp. 339–350 (2009)

17. Dignum, V.: Responsible Artificial Intelligence: How to Develop and Use AI in a Responsible Way. Springer International Publishing, Cham (2019)
18. DW, G.D.A.: Darpa's explainable artificial intelligence program. AI Mag. **40**(2), 44 (2019)
19. Etzioni, A., Etzioni, O.: Incorporating ethics into artificial intelligence. J. Ethics **12**, 403–418 (2017)
20. Ethics guidelines for trustworthy AI (2019), https://digital-strategy.ec.europa.eu/en/library/ethics-guidelines-trustworthy-ai
21. Ethics by design and ethics of use approaches for artificial intelligence (2021), https://ec.europa.eu/info/funding-tenders/opportunities/docs/2021-2027/horizon/guidance/ethics-by-design-and-ethics-of-use-approaches-for-artificial-intelligence_he_en.pdf
22. Floridi, L., Cowls, J., Beltrametti, M., et al.: Ai4people-an ethical framework for a good AI society: opportunities, risks, principles, and recommendations. Minds Mach. **28**, 689–707 (2018)
23. Fonseca, C.M., Guizzardi, G., Almeida, J.P.A., Sales, T.P., Porello, D.: Incorporating types of types in ontology-driven conceptual modeling. In: Conceptual Modeling: 41st International Conference, ER 2022, Hyderabad, India, October 17–20, 2022, Proceedings. pp. 18–34. Springer (2022)
24. Fonseca, C.M., Porello, D., Guizzardi, G., Almeida, J.P.A., Guarino, N.: Relations in ontology-driven conceptual modeling. In: Conceptual Modeling: 38th International Conference, ER 2019, Salvador, Brazil, November 4–7, 2019, Proceedings 38. pp. 28–42. Springer (2019)
25. Fonseca, C.M., Sales, T.P., Viola, V., da Fonseca, L.B.R., Guizzardi, G., Almeida, J.P.A.: Ontology-driven conceptual modelling as a service. In: JOWO (2021)
26. Guarino, N., Guizzardi, G.: "we need to discuss the relationship": revisiting relationships as modeling constructs. In: Advanced Information Systems Engineering: 27th International Conference, CAiSE 2015, Stockholm, Sweden, June 8–12, 2015, Proceedings 27. pp. 279–294. Springer (2015)
27. Guerson, J., Sales, T.P., Guizzardi, G., Almeida, J.P.A.: Ontouml lightweight editor: a model-based environment to build, evaluate and implement reference ontologies. In: 2015 IEEE 19th International Enterprise Distributed Object Computing Workshop. pp. 144–147. IEEE (2015)
28. Guizzardi, G.: Ontological foundations for structural conceptual models. Telematica Instituut / CTIT (2005)
29. Guizzardi, G.: Ontology, ontologies and the I of FAIR. Data Intell. **2**(1–2), 181–191 (2020)
30. Guizzardi, G., Bernasconi, A., Pastor, O., Storey, V.C.: Ontological unpacking as explanation: the case of the viral conceptual model. In: International Conference on Conceptual Modeling. pp. 356–366. Springer (2021)
31. Guizzardi, G., Fonseca, C.M., Almeida, J.P.A., Sales, T.P., Benevides, A.B., Porello, D.: Types and taxonomic structures in conceptual modeling: a novel ontological theory and engineering support. Data Knowl. Eng. **134**, 101891 (2021)
32. Guizzardi, G., Guarino, N.: Semantics, ontology, and explanation. ArXiv Preprint https://doi.org/10.48550/arXiv.2304.11124 (2023)
33. Guizzardi, G., Guarino, N., Almeida, J.P.A.: Ontological considerations about the representation of events and endurants in business models. In: Business Process Management: 14th International Conference, BPM 2016, Rio de Janeiro, Brazil, September 18–22, 2016. Proceedings 14. pp. 20–36. Springer (2016)
34. Guizzardi, G., Wagner, G., Almeida, J.P.A., Guizzardi, R.S.S.: Towards ontological foundations for conceptual modeling: the Unified Foundational Ontology (UFO) story. Appl. Ontol. **10**(3–4), 259–271 (2015)
35. Guizzardi, R.: Agent-oriented Constructivist Knowledge Management. Ph.D. thesis, University of Twente, Netherlands (2006)
36. Guizzardi, R., Amaral, G., Guizzardi, G., Mylopoulos, J.: Eliciting ethicality requirements using the ontology-based requirements engineering method. In: Augusto, A., Gill, A., Bork, D., Nurcan, S., Reinhartz-Berger, I., Schmidt, R. (eds.) Enterprise, Business-Process and Information Systems Modeling, pp. 221–236. Springer International Publishing, Cham (2022)
37. Guizzardi, R., Carneiro, B.G., Porello, D., Guizzardi, G.: A core ontology on decision making. In: Proceedings of the 13th Seminar on Ontology Research in Brazil (2020)
38. Guizzardi, G. et al.: Towards ontological foundations for the conceptual modeling of events. In: Proceedings of 32nd ER. pp. 327–341. Springer (2013)
39. Guizzardi, R. et al.: An ontological interpretation of non-functional requirements. In: Proc. of FOIS. vol. 14, pp. 344–357 (2014)
40. Guizzardi, R. et al.: Ethical requirements for ai systems. In: Proceedings of Canadian AI 2020. pp. 251–256 (2020)
41. High-Level Expert Group on Artificial Intelligence, E.C.: Draft ethics guidelines for trustworthy AI, draft document (2018)
42. Hu, H., Ma, Q., Zhang, T., Tan, Y., Xiang, H., Fu, C., Feng, Y.: Semantic modelling and automated reasoning of non-functional requirement conflicts in the context of softgoal interdependencies. IET Softw. **9**, 145–156 (2015)
43. Mumford, S.: Dispositions. Clarendon Press, Oxford (2003)
44. O'Neill, C.: Weapons of Math Destruction. How Big Data Increases Inequality and Threatens Democracy. Crown, New York (2016)
45. Peters, D., Vold, K., Robinson, D., Calvo, R.: Responsible AI: two frameworks for ethical design practice. IEEE Trans. Technol. Soc. **1**(1), 34–47 (2020)
46. Porello, D., Guizzardi, G.: Towards an ontological modeling of preference relations. In: 17th International Conference of the Italian Association for Artificial Intelligence. pp. 152–165 (2018)
47. Rashid, A., Moore, K., May-Chahal, C., Chitchyan, R.: Managing emergent ethical concerns for software engineering in society. In: 37th IEEE International Conference on Software Engineering. pp. 523–526. IEEE (2015)
48. Romanenko, E., Calvanese, D., Guizzardi, G.: Towards pragmatic explanations for domain ontologies. In: Knowledge Engineering and Knowledge Management: 23rd International Conference, EKAW 2022, Bolzano, Italy, September 26–29, 2022, Proceedings. pp. 201–208. Springer (2022)
49. Sales, T. et al.: The Common Ontology of Value and Risk. In: Proceedings of 37th International Conference on Conceptual Modeling (ER). vol. 11157, pp. 121–135. Springer (2018)
50. Searle, J.R.: Minds, brains, and programs. Behav. Brain Sci. **3**(3), 417–424 (1980)
51. Smith, B., Simon, J.: Truthmaker explanations. Metaphys. truthmakers **18**, 79 (2007)
52. Thompson, N.: Grounding and metaphysical explanation. In: Proceedings of the Aristotelian Society. vol. 116, pp. 395–402. Oxford University Press (2016)
53. The 2-second rule. In: Learn the Road Code (2016), https://drive.govt.nz/get-your-learners/interactive-road-code/
54. The 7 Most Pressing Ethical Issues in Artificial Intelligence (2019), https://kambria.io/blog/the-7-most-pressing-ethical-issues-in-artificial-intelligence/
55. Van Fraassen, B.: The pragmatic theory of explanation. Theor. Explan. **8**, 135–155 (1988)
56. Veleda, R., Cysneiros, L.: Towards an ontology-based approach for eliciting possible solutions to non-functional requirements. In: 31st International Conference on Advanced Information Systems Engineering. pp. 145–161. Springer (2019)
57. Verdonck, M., Gailly, F.: Insights on the use and application of ontology and conceptual modeling languages in ontology-driven conceptual modeling. In: Conceptual Modeling: 35th International Conference, ER 2016, Gifu, Japan, November 14–17, 2016, Proceedings 35. pp. 83–97. Springer (2016)

58. Verdonck, M., Gailly, F., Pergl, R., Guizzardi, G., Martins, B., Pastor, O.: Comparing traditional conceptual modeling with ontology-driven conceptual modeling: an empirical study. Inf. Syst. **81**, 92–103 (2019)

Renata Guizzardi is currently an Assistant Professor at the Behavioral, Management and Social Sciences Faculty, (IEBIS group) of the University of Twente, in the Netherlands. Moreover, she is a founding member of the Ontology & Conceptual Modeling Research Group (NEMO) and of the Laboratory of Supporting Technologies for Collaborative Networks (LabTAR), at UFES, Brazil. For around 30 years, she has been busy with research work on Requirements Engineering, Conceptual Modeling and Ontologies, and Computer-Assisted Education, focusing on the interplay of these research areas to improve the development of information systems and organizational practices. Since 2021, she has served as the chair of the steering committee of the Iberoamerican Conference on Software Engineering (CIbSE), also having been part of the organization committee of numerous other conferences, such as ER, RCIS, CAiSE and EDOC. She is also in the editorial board of the International Journal of Knowledge and Learning and has served as a reviewer for relevant scientific journals, such as the Springer Software and Systems Modeling and the IAOA Applied Ontologies.

Glenda Amaral is a Senior Information Governance Analyst at the Central Bank of Brazil. She is also a Guest Researcher at the Department of Semantics, Cybersecurity & Services (SCS) at the University of Twente, The Netherlands. She received a PhD cum laude in Computer Science from the Free University of Bolzen-Bolzano, Italy, and a MSc in Informatics from the Federal University of Rio de Janeiro, Brazil. Glenda has over 20 years of experience with conceptual modeling and information management in the public service, in Brazil. She is currently interested in ontology-driven conceptual modeling for the financial sector, ontology-based requirements engineering methods, and ontology applications in the context of data quality management and decentralized finance.

Giancarlo Guizzardi is a Full Professor of Software Science and Evolution as well as Chair and Department Head of Semantics, Cybersecurity & Services (SCS) at the University of Twente, The Netherlands. He is also an Affiliated/Guest Professor at the Department of Computer and Systems Sciences (DSV) at Stockholm University, in Sweden. He has been active for nearly three decades in the areas of Formal and Applied Ontology, Conceptual Modelling, Business Informatics, and Information Systems Engineering, working with a multi-disciplinary approach in Computer Science that aggregates results from Philosophy, Cognitive Science, Logics and Linguistics. Over the years, he has delivered keynote speeches in several key international conferences in these fields (e.g., ER, BPM, CAiSE). He is currently an associate editor of a number of journals including Applied Ontology and Data & Knowledge Engineering, a co-editor of the Lecture Notes in Business Information Processing series, and a member of several international journal editorial boards. Finally, he is a member of the Steering Committees of ER, EDOC, and IEEE CBI, and of the Advisory Board of the International Association for Ontology and its Applications (IAOA).

John Mylopoulos holds a professor emeritus position at the Universities of Toronto (Canada) and Trento (Italy), and is working at the University of Ottawa on a project titled 'Engineering Smart Contracts' as visiting researcher. He earned a PhD degree from Princeton University in 1970 and joined the faculty of the Department of Computer Science at the University of Toronto the same year. His research interests include conceptual modelling,requirements engineering, data semantics, and knowledge management. Mylopoulos is a fellow of the Association for the Advancement of Artificial Intelligence (AAAI) and the Royal Society of Canada (Academy of Applied Sciences). He has served as program/general chair of international conferences in Artificial Intelligence, Databases and Software Engineering, including IJCAI (1991), Requirements Engineering (1997, 2011), and VLDB (2004). Mylopoulos was project leader for a project titled 'Lucretius: Foundations for Software Evolution,' funded by an advanced grant from the European Research Council (2011–16).