An Ontology-Grounded Representation for Defeasible Professional Ethics Analysis

Christopher B. Rauch, J.D.[0000-0003-2061-3413]

Drexel University, Philadelphia PA 19104, USA
cr625@drexel.edu

Abstract. The evaluation of professional conduct requires the identification of applicable obligations when multiple provisions of the same professional code generate competing requirements. In prior work we defined a nine-component case representation for professional ethics analysis. This paper presents the ontological implementation of that representation, grounded in the Basic Formal Ontology (BFO) and classified through the Information Artifact Ontology (IAO), with type-level disjointness verified by the Pellet OWL-DL reasoner. Three object properties (`competesWith`, `prevailsOver`, `defeasibleUnder`) expose the competition structure between obligations as SPARQL-queryable relationships between typed individuals. OntServe, an ontology server, instantiates the representation as a knowledge graph of case ontologies extracted from a professional board of ethics archive.

Keywords: professional ethics · ontology · defeasible reasoning · case representation · OWL-DL · knowledge graph · BFO

1 Introduction

The evaluation of professional conduct often involves elements from multiple ethical traditions, including deontology, consequentialism, and virtue-based role ethics [16]. Automated case analysis must represent normative content from multiple ethical traditions simultaneously and track temporal sequences of actions and events. Few existing systems address both requirements. Professional ethics evaluation is bounded by established codes [7], defined roles, and case-based adjudication, where codes map to deontic structures, roles anchor obligation assignments, and prior cases introduce temporal ordering. This alignment makes a combined approach tractable.

These three dimensions (deontic structure, role-based obligations, and temporal ordering) require formal, queryable components. In prior work [14] we defined a nine-component case representation formalized as $D = (R, P, O, S, Rs, A, E, Ca, Cs)$. This paper presents the ontological implementation of that representation, grounded in the Basic Formal Ontology (BFO) [2], serialized in RDF/OWL, and served through OntServe,[1] a SPARQL-queryable ontology server. The implementation encodes temporal relations through Allen's [1] interval algebra via OWL-Time [5], and the Pellet OWL-DL reasoner [15] verifies type-level disjointness

[1] https://github.com/cr625/OntServe

across entity classifications. Three new object properties (`competesWith`, `prevailsOver`, `defeasibleUnder`) expose the competition structure between obligations as first-class graph relationships, making the defeasible patterns in professional ethics cases queryable and available to external reasoning engines.

2 Background

Two surveys map the computational ethics field from complementary angles. Tolmeijer et al. [16] classify existing implementations across ethical theories, non-technical design choices, and technical properties, finding that most systems apply a single ethical theory, few incorporate professional domain codes, and almost none provide formal temporal reasoning. Zhong et al. [17] reframe the field around the decision-making process itself, organizing approaches by their source of ethical guidance, their decision method, and their evaluation criteria. Viewed through this process lens, professional ethics evaluation presents a distinctive configuration. The source is an established domain code combined with precedent cases, the decision method requires both deontic dependency resolution and temporal sequence reasoning, and evaluation must reference expert-authored adjudications rather than abstract benchmarks.

The National Society of Professional Engineers (NSPE) Board of Ethical Review (BER) has published over 650 advisory opinions applying the NSPE Code of Ethics to specific fact patterns, each citing prior board decisions [11]. Individual components of this configuration appear in prior work. Ashley and McLaren [3] developed case-based knowledge representations for practical ethics, and McLaren's [10] SIROCCO demonstrates that precedent cases can serve as the computational source for professional ethical guidance. Berreby et al.'s [4] Event Calculus architecture provides a decision method that composes obligations, actions, and events with explicit temporal semantics.

3 Defeasible Normative Representation

The D-tuple $D = (R, P, O, S, Rs, A, E, Ca, Cs)$ defines nine components that represent the structural elements involved in obligation competition within professional ethics cases. Three functional dimensions organize them. Contextual Grounding components (Roles, States, Resources) establish who acts under what conditions with what knowledge. Normative Structure components (Principles, Obligations, Capabilities, Constraints) define what conduct is required, permitted, and prohibited. Temporal Dynamics components (Actions, Events) record what occurred and in what order.

The BFO/IAO classification [2] enforces foundational disjointness. The continuant/occurrent boundary prevents States from being confused with Actions at the type level. The directive/descriptive distinction within IAO separates normative content (Principles, Obligations, Constraints) from factual knowledge (Resources).

3.1 Defeasible Normative Structure

The dependency structure of the D-tuple represents the relationships along which obligations compete in professional ethics evaluation. Professional ethics is inherently defeasible. Obligations derived from one Principle can be overridden by stronger Obligations derived from another when specific States obtain, and an Obligation is unrealizable when the agent lacks the required Capability (the deontic principle that obligation presupposes capacity [8]). Defeasibility operates along these typed relationships. Representing the components as typed individuals with named edges between them makes the competition structure explicit and queryable.

The dependency chain $R \rightarrow P \rightarrow O$ establishes the normative structure. A Role generates applicable Principles, which derive concrete Obligations. This derivation follows Dennis et al. [6], who show that abstract principles require translation into context-dependent obligations. The D-tuple does not resolve the resulting competitions. It makes them explicit as named, typed relationships that a human evaluator or external reasoning engine can query and adjudicate.

For example, an engineer bound by client confidentiality who discovers a public safety risk faces two competing obligations derived from the same professional code [11]. The obligation to protect public safety overrides the obligation to maintain confidentiality when the State "public safety risk identified" obtains. The $R \rightarrow P \rightarrow O$ chain produces both obligations from the same Role, but the State determines which prevails. This pattern generalizes to any profession where a code of ethics generates obligations that can conflict under specific conditions. NSPE BER Case 72 illustrates this with a concrete scenario. The extraction client (Section 4) produces typed OWL individuals from the case narrative, and three defeasibility object properties connect them. Figure 1 shows three typed individuals extracted from Case 72 and the defeasibility edges (`competesWith`, `prevailsOver`, `defeasibleUnder`) introduced in the core ontology to expose this pattern as first-class graph structure.

3.2 Architectural Separation for Decidability

Representing defeasible normative reasoning in full generality exceeds the expressiveness of decidable description logics. OWL-DL is decidable but cannot natively express defeasibility or obligation override. The architecture addresses this through separation of concerns. The type-level ontology (component classifications, disjointness axioms, BFO alignments) remains within OWL-DL and benefits from sound and complete reasoning by Pellet [15]. The defeasible patterns described above are represented as structured relationships between typed individuals, queryable via SPARQL and available to external reasoning engines that support non-monotonic inference. OntServe (Section 4) provides the knowledge layer for such engines, exposing validated typed entities and their dependency relationships without requiring the ontology itself to encode defeasible inference rules.

```
# State that activates the competition (from case 72 facts)
case72:State_ConfirmedDischargeRisk
    a proeth:ConfirmedRiskWithoutAdequateSafeguardsState ;
    rdfs:label "Discharge below environmental standards (confirmed)" ;
    proeth:sourcetext "Doe concludes that the discharge from the plant
        will lower the quality of the receiving body of water below
        established standards" .

# Obligation A: derived from NSPE Sec. 1 (faithful-agent principle)
case72:Obl_FaithfulAgent
    a proeth:FaithfulAgentObligation ;
    rdfs:label "Faithful-agent obligation to XYZ" ;
    proeth:obligatedparty "Engineer Doe" ;
    proeth:sourcetext "will act ... as a faithful agent or trustee
        (NSPE Sec. 1)" .

# Obligation B: derived from NSPE Sec. 2(a) (paramount public duty)
case72:Obl_PublicSafetyReport
    a proeth:PostTermEnvRiskReportingObligation ;
    rdfs:label "Post-termination public-safety reporting" ;
    proeth:obligatedparty "Engineer Doe" ;
    proeth:sourcetext "duty to the public be paramount (NSPE Sec. 2a)" .

# Defeasibility edges (core ontology)
case72:Obl_FaithfulAgent
    proeth-core:competesWith case72:Obl_PublicSafetyReport ;
    proeth-core:defeasibleUnder case72:State_ConfirmedDischargeRisk .

case72:Obl_PublicSafetyReport
    proeth-core:prevailsOver case72:Obl_FaithfulAgent .
```

Fig. 1. NSPE BER Case 72 (Public Welfare, Knowledge of Information Damaging to Client's Interest). Three typed individuals extracted from the case narrative and the defeasibility edges that connect them. Two Obligations derived from different code sections compete. The State representing confirmed environmental risk renders the faithful-agent obligation defeasible, and the public-safety obligation prevails. Defeasibility edges are object properties in the core ontology, queryable via SPARQL. Type-level classification verified by the Pellet reasoner (§ 4). Local names abbreviated. Provenance triples omitted.

4 Implementation

The representation described in Section 3 requires an infrastructure that serves formally typed case ontologies, validates LLM-extracted classifications, and exposes the graph to SPARQL queries. OntServe implements this infrastructure as a three-layer ontology architecture.

The core layer declares the nine component classes with an `owl:AllDisjointClasses` assertion and the BFO/IAO alignments described in Section 3. A domain layer adds profession-specific subclasses. For engineering ethics, these include subclasses of Obligation for disclosure duties and public safety requirements, and subclasses of Resource for NSPE Code provisions and board precedents [11]. Each case then produces an individual layer: one OWL ontology per case, containing the typed individuals extracted from that narrative, linked to shared class definitions in the domain and core layers.

The repository currently holds 134 ontologies, including 119 case ontologies, 11 external ontologies (BFO, IAO, Relations Ontology, OWL-Time, and three professional codes of ethics) and four domain ontologies. The full graph is served through a SPARQL endpoint.

Pellet reasoning (Section 3.2) is implemented via owlready2 [9]. The reasoner operates on individual case ontologies with imports stripped, diffs the class hierarchy before and after inference, and can persist inferred relationships as a new ontology version.

OntServe exposes ontology definitions via the Model Context Protocol (MCP) for use by extraction clients. The current client is an LLM that proposes typed entities from case text without traversing the ontology or drawing inferences. Pellet verifies that the resulting classifications conform to the type structure described in Section 3. A State misclassified as an Action, for example, would violate the continuant/occurrent boundary and produce an inconsistency. The ontological framework defines what the LLM should extract. The reasoner verifies conformance.

OntServe is publicly accessible at https://ontserve.ontorealm.net.

5 Evaluation

The ontological infrastructure described in Sections 3 and 4 was evaluated by populating it with 119 case ontologies produced by the extraction client described in our prior work. Evaluation addresses three properties of the resulting knowledge graph: structural coverage, ontological consistency, and downstream retrieval performance.

The extraction client produced 119 case ontologies from NSPE Board of Ethical Review opinions spanning 1958 to 2025. Each case ontology contains between 170 and 310 typed individuals classified across the nine components, with coverage across all three functional dimensions in every case. Entity classes accumulate across cases, so the shared vocabulary grows without duplication.

Ontological consistency follows from the formal structure. The `owl:AllDisjointClasses` assertion over all nine components, combined with the BFO continuant/occurrent boundary, means that the SPARQL endpoint can validate classification integrity without additional logic. A query checking for entities with two incompatible component types will return empty because the ontology makes such a classification incoherent at the schema level. The Pellet reasoner confirms consistency for each case ontology submitted to inference.

The shared entity vocabulary also supports case retrieval. A companion paper [13] finds that component-aware embeddings over the same 119-case pool recover cited precedents at three to eight times the random baseline.

The evaluation is limited in two respects. The case base covers a single professional domain. Expert evaluation of whether the typed classifications and defeasibility relationships match domain practitioners' judgment has not yet been conducted.

6 Conclusion

The D-tuple provides a representation for professional ethics cases that unifies deontic dependency chains and formal ontological classification within a single deployable graph. BFO supplies the upper-level type structure. The Pellet OWL-DL reasoner validates consistency across case ontologies. Three defeasibility object properties expose the competition structure between obligations as queryable graph relationships.

OntServe instantiates this representation as a knowledge base with a SPARQL endpoint for cross-case queries and an MCP interface for extraction clients. A prior application of role-based ethics validation to military medical triage [12] demonstrates that the underlying evaluation model transfers across professional domains. Extension to any profession with established codes and precedent systems requires a domain ontology and corresponding code resources. Planned future work includes integrating a defeasible logic engine to formally resolve the obligation competitions that the current representation captures as structured data.

Acknowledgments. The NSPE Board of Ethical Review cases used in this study are publicly available through the NSPE Ethics Resources website.

Disclosure of Interests. The author has no competing interests to declare.

Declaration on Generative AI. During the preparation of this work, the author used Claude (Anthropic) to assist with manuscript editing. After using this tool, the author reviewed and edited the content as needed and takes full responsibility for the content of this publication.

References

1. Allen, J.F.: Maintaining knowledge about temporal intervals. Communications of the ACM **26**(11), 832–843 (1983). https://doi.org/10.1145/182.358434
2. Arp, R., Smith, B., Spear, A.D.: Building Ontologies with Basic Formal Ontology. MIT Press (2015)
3. Ashley, K.D., McLaren, B.M.: A CBR knowledge representation for practical ethics. In: Advances in Case-Based Reasoning. pp. 180–197. Springer (1995). https://doi.org/10.1007/3-540-60364-6_36
4. Berreby, F., Bourgne, G., Ganascia, J.G.: A declarative modular framework for representing and applying ethical principles. In: 16th Conference on Autonomous Agents and MultiAgent Systems. pp. 96–104 (2017)
5. Cox, S., Little, C.: OWL-Time: Time ontology in OWL. W3C Recommendation (2017), https://www.w3.org/TR/owl-time/
6. Dennis, L.A., Fisher, M., Slavkovik, M., Webster, M.: Formal verification of ethical choices in autonomous systems. Robotics and Autonomous Systems **77**, 1–14 (2016). https://doi.org/10.1016/j.robot.2015.11.012
7. Frankel, M.S.: Professional codes: Why, how, and with what impact? Journal of Business Ethics **8**(2–3), 109–115 (1989). https://doi.org/10.1007/BF00382575
8. Govindarajulu, N.S., Bringsjord, S.: On automating the doctrine of double effect. In: Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence. pp. 4722–4730 (2017). https://doi.org/10.24963/ijcai.2017/658
9. Lamy, J.B.: Owlready: Ontology-oriented programming in Python with automatic classification and high level constructs for biomedical ontologies. Artificial Intelligence in Medicine **80**, 11–28 (2017). https://doi.org/10.1016/j.artmed.2017.07.002
10. McLaren, B.M.: Extensionally defining principles and cases in ethics: An AI model. Artificial Intelligence **150**(1–2), 145–181 (2003). https://doi.org/10.1016/S0004-3702(03)00135-8
11. National Society of Professional Engineers: Board of ethical review case archive (2025), https://www.nspe.org/career-growth/ethics/board-ethical-review-cases
12. Rauch, C.B., Molineaux, M., Mainali, M., Sen, A., Floyd, M.W., Weber, R.O.: Role-based ethics for decision-maker alignment. In: 2025 IEEE Conference on Artificial Intelligence (CAI). pp. 1209–1212 (2025). https://doi.org/10.1109/CAI64502.2025.00209
13. Rauch, C.B., Weber, R.O.: Component-aware case retrieval for professional ethics with ontology-constrained LLM extraction. In: Case-Based Reasoning Research and Development: ICCBR 2026. Springer (2026), under review
14. Rauch, C.B., Weber, R.O.: ProEthica: A professional role based ethical analysis tool using LLM-orchestrated, ontology supported case based reasoning. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 40, pp. 41673–41675 (2026). https://doi.org/10.1609/aaai.v40i48.42377
15. Sirin, E., Parsia, B., Grau, B.C., Kalyanpur, A., Katz, Y.: Pellet: A practical OWL-DL reasoner. Journal of Web Semantics **5**(2), 51–53 (2007). https://doi.org/10.1016/j.websem.2007.03.004
16. Tolmeijer, S., Kneer, M., Sarasua, C., Christen, M., Bernstein, A.: Implementations in machine ethics: A survey. ACM Computing Surveys **53**(6), 132:1–132:38 (2021). https://doi.org/10.1145/3419633
17. Zhong, T., Song, Y., Limarga, R., Pagnucco, M.: Computational machine ethics: A survey. Journal of Artificial Intelligence Research **82**, 1581–1628 (2025). https://doi.org/10.1613/jair.1.16836