Date : 10. November 2025

Modeling Civil Engineered Systems

Individual Assignment I: Ontological Modeling

Labkhand Fadaietamijani - TU Berlin - Student ID: 523630

1. Introduction and Background Research

Retaining walls are structural systems designed to hold back soil and create level ground in areas with different elevations. They are widely used in roadways, bridge abutments, basements, and landscaping. Their main purpose is to resist lateral earth pressure and maintain ground stability in places where the natural slope of the soil cannot be preserved (Das, 2019; Coduto, 2018). Retaining walls also form part of the interface between structural and geotechnical systems, providing stability for nearby infrastructure and safe working areas during construction.

There are several common types of retaining walls, including gravity walls, crib walls, sheet pile walls, gabion walls, and reinforced concrete cantilever walls. In this project, the focus is on the reinforced concrete cantilever retaining wall, one of the most frequently used systems in civil engineering due to its strength, stability, and cost efficiency for medium heights of about 6–8 meters (CIRIA, 2019). Ontology-based approaches have increasingly been used to integrate civil engineering knowledge and improve interoperability across structural and geotechnical domains (Zhang & El-Gohary, 2018).

A cantilever retaining wall consists of key components such as a vertical stem, heel slab, toe slab, base slab, and shear key. These components work together to resist various forces, including self-weight, lateral earth pressure, and surcharge loads. The wall interacts with the surrounding soil, and the selection of materials directly influences its performance and durability. Concrete and steel reinforcement are typically used to provide strength and serviceability (Pahl & Beitz, 2007).

To understand the engineering system, I reviewed standard design documents and literature, including Eurocode 7 for geotechnical design (EN 1997-1:2004+A1:2013), Eurocode 2 for concrete design (EN 1992-1-1:2004), and the CIRIA C760 Retaining Wall Design Guide (2019). These sources provided essential insights into the system's functions, physical behavior, and key design parameters. I also compared examples from JRC (2014) Eurocode 7 Worked Examples to identify typical wall configurations and load conditions.

This figure presents the main physical components and logical relationships of a reinforced concrete cantilever retaining wall. The wall consists of a vertical stem, a base slab, and two main extensions: the heel (under the retained soil) and the toe (on the exposed side). A shear key beneath the base slab improves sliding resistance. The wall interacts with the retained soil, which exerts lateral earth pressure represented by the isRetainedBy relation, and it is also subject to self-weight and surcharge loads, represented by isSubjectTo. Each labeled relationship, such as hasComponent or isSubjectTo, reflects how the physical structure is later formalized in the ontology. This figure bridges the engineering system and its logical decomposition, forming the foundation for the ontological model developed in Protégé.

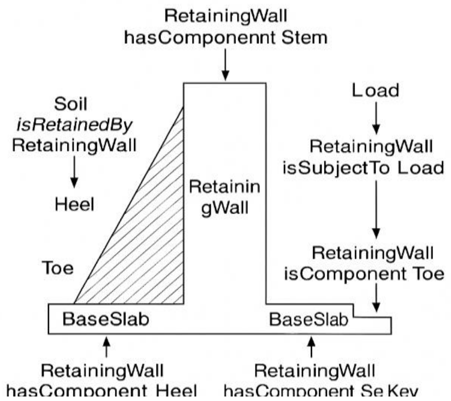

Figure 1. Rough System Sketch

Reinforced Concrete Cantilever Retaining Wall

2. Ontology Development Process

The ontology was developed following the general guidance of Noy and McGuinness (2001) and the description logic principles discussed by Krötzsch et al. (2012). The main objective was to create a clear and logically consistent representation of the structural, material, and loading aspects of a reinforced concrete cantilever retaining wall. The process began by defining the overall purpose, scope, and intended use of the ontology. The goal was to represent the essential elements and relationships that describe how the wall behaves as a system. The key classes identified were RetainingWall, Component, Material, Load, and Soil. Each of these was refined into more specific subclasses, such as BaseSlab, Stem, Heel, Toe, ShearKey, and LateralEarthPressure.

Following a top-down design approach (Noy & McGuinness, 2001), broader concepts were defined first, then specialized into detailed components. Relationships among these classes were represented through object properties such as hasComponent, isComponentOf, isSubjectTo, hasMaterial, and isRetainedBy. As suggested by Guarino et al. (2009), each relationship was defined to ensure both conceptual clarity and logical consistency, enabling machine-readable reasoning and knowledge reuse. This ensured that physical, functional, and material aspects of the retaining wall were logically connected.

To maintain logical rigor, I applied the description logic framework described by Krötzsch and colleagues, ensuring correct use of subclass hierarchies, disjointness axioms, inverse properties, and existential restrictions. Individuals such as RetainingWallOption1 and RetainingWallOption2 were created to represent specific design alternatives, each containing unique geometric and material data.

The ontology was implemented in Protégé 5.6.8, using the DL axioms provided in the TU Berlin Axioms Mapping – DL and Protégé (Version 2) tutorial (2025) as a structural reference. During the modeling process, relevant engineering standards and design documents, including Eurocode 7, Eurocode 2, JRC (2014), and CIRIA C760 (2019), were consulted to ensure that the modeled components, materials, and loads accurately reflected real-world engineering conditions.

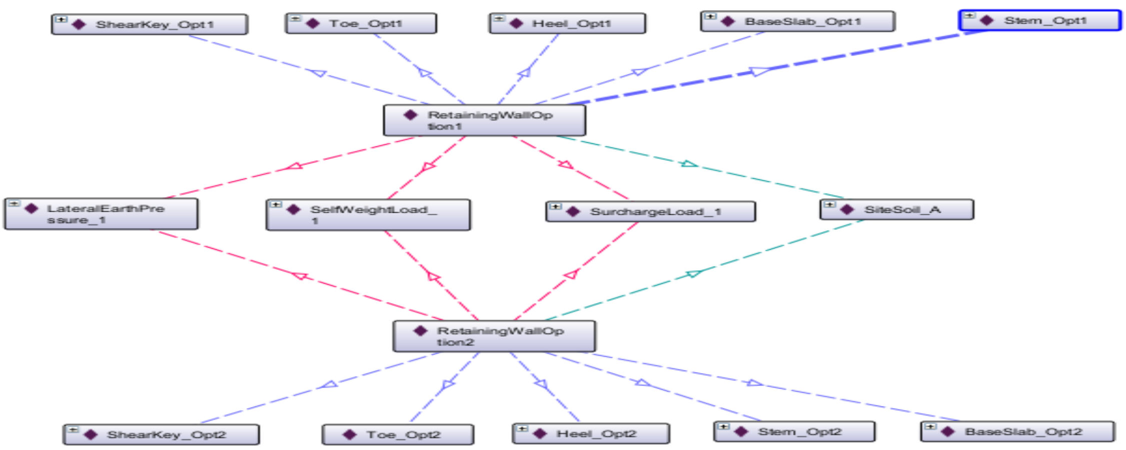

Figure 2. Object Property Network in Protégé

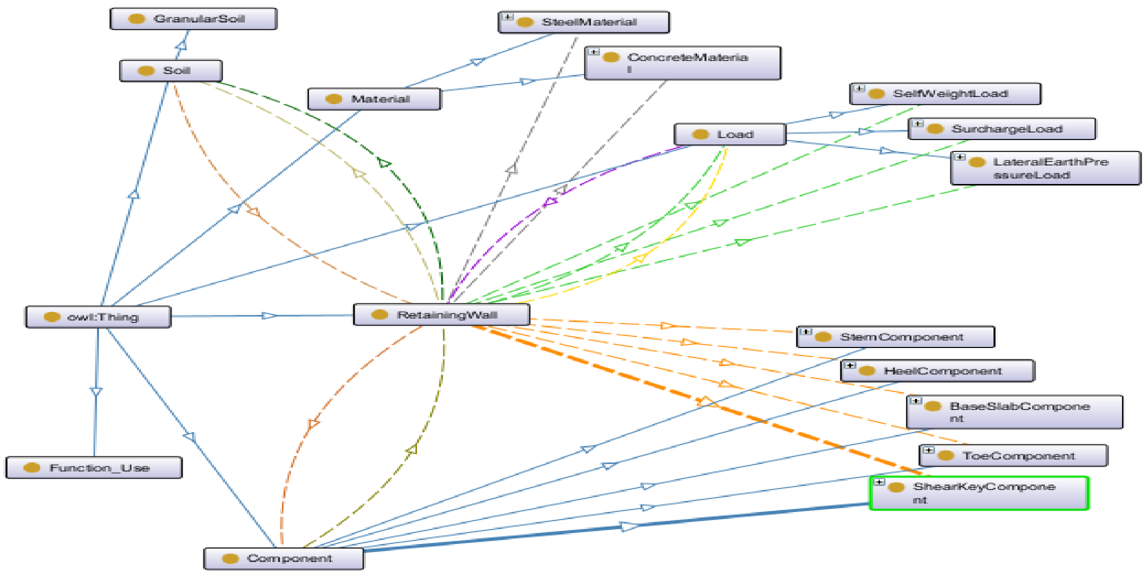

Figure 3. Class Hierarchy in Protégé

3. Purpose, Scope, Users, and Intended Use

The purpose of this ontology is to represent the structural, functional, and material relationships of a reinforced concrete cantilever retaining wall in a logically consistent way. The scope includes key components such as the stem, base slab, heel, toe, shear key, and their interactions with soil and loads. The intended users are structural and geotechnical engineers, as well as researchers working on data integration and automated reasoning in civil engineering. The ontology is intended to support design evaluation, consistency checking, and integration of information across disciplines.

4. Engineering Examples

Example 1 – Comparing Design Alternatives under Site Constraints

Scenario:
An engineer is designing two reinforced concrete cantilever retaining walls for a narrow site beside an existing roadway. Option 1 has a shorter base slab to fit within the property boundary, while Option 2 uses a wider base with a shear key for additional sliding resistance.

Use Case:
Within the ontology, each wall alternative (RetainingWallOption1 and RetainingWallOption2) is defined as an individual with its own components and materials. Through relationships such as hasComponent, hasMaterial, and isSubjectTo, engineers can query and compare the two options under similar soil and loading conditions. A reasoning tool can identify which option meets stability requirements with less material usage or lower cost. The resulting insights can then be exported to a parametric design model for further optimization and analysis.

Example 2 – Ensuring Consistency in Design Documentation

Scenario:
During the digital modeling of a retaining wall project, a design team discovers that certain wall entries are missing a stem component or have incorrect material assignments. Such inconsistencies can cause major issues during structural analysis and construction documentation.

Use Case:
The ontology supports automatic verification of model completeness. Because the class RetainingWall must include at least one BaseSlab and one Stem according to its DL axioms, the reasoning engine flags any wall instance that lacks these components. Similarly, if a Heel is linked to an incompatible material, such as steel instead of reinforced concrete, the ontology identifies the logical conflict. These automated checks ensure that the digital model remains consistent and technically valid before it is integrated into BIM or structural analysis software.

Example 3 – Adapting Design to Changing Soil Conditions

Scenario:
A retaining wall initially designed for dry sandy soil must be reassessed after a site investigation reveals clayey soil with higher lateral earth pressure.

Use Case:
In the ontology, the *Soil* and *Load* classes are connected to the RetainingWall through relationships such as isRetainedBy and isSubjectTo. When soil type information is updated, the reasoning engine can infer corresponding changes to lateral pressure and suggest alternative wall configurations or dimensions. This allows engineers to quickly adjust design parameters within a parametric modeling environment, ensuring structural safety and design efficiency under new geotechnical conditions.

5. Description Logic Axioms Table

The following table summarizes the main logical axioms used in the ontology developed in Protégé 5.6.8.
The full ontology contains a larger set of subclass, disjointness, restriction, and property axioms; however, due to page limitations, only the most representative examples are presented here.
Each row illustrates one of the key Description Logic patterns applied in the model—linking the engineering meaning (e.g., wall–component, wall–load, or wall–material relations) with its formal DL expression.

DL axiom (type)	**Example (DL notation)**	**Protégé entry (typical)**	**Application in my model (what I actually did)**
Subclass	BaseSlab ⊑ Component	SubClass Of	Declared all main parts (BaseSlab, Stem, Heel, Toe, ShearKey) as Components.
Disjoint Classes	Heel ⊓ Toe ⊓ Stem ⊓ ShearKey ⊑ ⊥	Disjoint Classes	Ensured structural parts are mutuall exclusive.
Inverse Properties	hasComponent ≡ isComponentOf^{-}	Inverse Of	Enables bidirectional reasoning between wall and components.
Domain & Range	Domain(hasComponent)=RetainingWall Range(hasComponent)=Component	Object Property → Domain/Range	Maintains logical typing between wal and parts.
Existential Restriction (components)	RetainingWall ⊑ hasComponent some BaseSlab	Subclass Of → Restriction	Every wall includes key components.
Existential Restriction (loads)	RetainingWall ⊑ isSubjectTo some LateralEarthPressure	Subclass Of → Restriction	Every wall is subject to major loads
Existential Restriction (soil)	RetainingWall ⊑ retains some Soil	Subclass Of → Restriction	Every wall retains soil mass.
Existential Restriction (materials)	RetainingWall ⊑ hasMaterial some ConcreteMaterial	Subclass Of → Restriction	Wall linked to main construction materials.
Object Assertions (ABox)	hasComponent(RetainingWallOption1, BaseSlab_1)	Individual Assertion	Connects wall instances to their component individuals
Data Property Assertions (ABox)	BaseSlab_1 hasLength "3.5"^^xsd:double	Data Property Assertion	Attaches geometry data to components.

Together, these axioms ensure that the ontology of the reinforced concrete retaining wall remains both logically consistent and practically applicable to civil engineering analysis.

6. Key Engineering and Ontology Sources Used in Protége

The following key engineering and ontology sources were used while developing the model in Protégé to ensure technical accuracy and logical consistency:

EN 1997-1:2004+A1:2013 – Eurocode 7: Geotechnical Design.

EN 1992-1-1:2004 – Eurocode 2: Concrete Design.

CIRIA C760 (2019) – Retaining Wall Design Guide.

JRC (2014) – Eurocode 7 Worked Examples.

Noy, N. F., & McGuinness, D. L. (2001) – Ontology Development 101.

Krötzsch, M., Simancik, F., & Horrocks, I. (2012) – A Description Logic Primer.

7. References

Das, B. M. (2019). Principles of Foundation Engineering. Cengage Learning.

Coduto, D. P. (2018). Foundation Design: Principles and Practices. Pearson.

Pahl, G., & Beitz, W. (2007). Engineering Design: A Systematic Approach. Springer-Verlag.

CIRIA C760. (2019). Retaining Wall Design Guide. Construction Industry Research and Information Association.

European Committee for Standardization. (2004). EN 1997-1: Eurocode 7 – Geotechnical Design.

European Committee for Standardization. (2004). EN 1992-1-1: Eurocode 2 – Design of Concrete Structures.

JRC. (2014). Eurocode 7 Worked Examples. European Commission.

Noy, N. F., & McGuinness, D. L. (2001). Ontology Development 101: A Guide to Creating Your First Ontology. Stanford University.

Krötzsch, M., Simancik, F., & Horrocks, I. (2012). A Description Logic Primer. arXiv:1201.4089.

Zhang, **L.**, & El-Gohary, N. M. (2018). Ontology-based reasoning for sustainable infrastructure design. Journal of Computing in Civil Engineering, 32(5), 04018037.

Guarino, N., Oberle, D., & Staab, S. (2009). *What Is an Ontology?* In S. Staab & R. Studer (Eds.), *Handbook on Ontologies* .(pp. 1–17). Springer.